



Universidade de Évora - Instituto de Investigação e Formação Avançada

Programa de Doutoramento em Informática

Tese de Doutoramento

**An Uncertainty Prediction Approach for Active Learning -
Application to Earth Observation**

Kashyap Damjibhai Raiyani

Orientador(es) | Luís Rato

Teresa Gonçalves

Évora 2023



Universidade de Évora - Instituto de Investigação e Formação Avançada

Programa de Doutoramento em Informática

Tese de Doutoramento

**An Uncertainty Prediction Approach for Active Learning -
Application to Earth Observation**

Kashyap Damjibhai Raiyani

Orientador(es) | Luís Rato
Teresa Gonçalves

Évora 2023



A tese de doutoramento foi objeto de apreciação e discussão pública pelo seguinte júri nomeado pelo Diretor do Instituto de Investigação e Formação Avançada:

Presidente | Salvador Abreu (Universidade de Évora)

Vogais | António Anjos (Universidade de Évora)
João Carlos Gomes Moura Pires (Universidade Nova de Lisboa - Faculdade de Ciências e Tecnologias)
Luís Filipe Barbosa de Almeida Alexandre (Universidade da Beira Interior)
Luís Rato (Universidade de Évora) (Orientador)
Nuno da Cruz Garcia (Universidade de Lisboa - Faculdade de Ciências)



•
• **UNIVERSIDADE DE ÉVORA**
•

• **Colégio Luis António Verney**
•

• Departamento de Informática
•
•
•
•
•
•
•
•
•
•
•
•
•
•

• **An Uncertainty Prediction Approach for**
• **Active Learning – Application to Earth**
• **Observation**
•

• **Kashyap Damjibhai Raiyani**
•
•
•

• Orientação *Teresa Cristina de Freitas Gonçalves*
• *Luís Miguel de Mendonça Rato*
•
•
•

• **Programa de Doutoramento em Informática**
•
•

• **Dissertação**
•
•
•
•

• 27 de Fevereiro de 2023
•
•

• *Esta dissertação não inclui as críticas e sugestões feitas pelo Júri*
•
•

*To my brother **Kunal**...*

Acknowledgement

We frequently meet obstacles and even major setbacks on our way to achieving any meaningful objective in life. High tenacity is necessary to climb the mountain that is a Doctor of Philosophy degree, but there is also much to say about those who assist us in reaching our goals, whose appreciation is sometimes overestimated or, worse, ignored totally. It is in this place that I acknowledge people who have helped me reach my goals.

Dr Teresa Cristina de Freitas Gonçalves's supervision and communicable vision pushed me to pursue my doctorate in the first place. You provided me with a research atmosphere that encouraged innovative thought, and initiative led me to realise I need to improve a lot. Our informative talks, constructive critiques, and brainstorming sessions all contributed to my personal growth, and for that, I could not thank you enough. I appreciate your excellent counsel and support during the overall supervision process; without your unwavering mentoring and relentless effort, I would not have achieved my objectives. I could not have asked for a superior doctorate research adviser.

Dr Luís Miguel de Mendonça Rato is the most cogitative and industrious person I have ever encountered. I'm appreciative of your endless attention, compassion, and enthusiasm as well as for introducing me to the realm of 'uncertainty'. Your prompt response to my work, together with your out-of-box comments, ideas, and revisions, helped me overcome all of the obstacles in my way and reach my milestones. It goes far beyond what I could have envisioned for you to be available after office hours and to always have my back. You constantly challenged me to think outside the box and achieve more, and I appreciate the chance to have a long-lasting, friendly relationship with you that goes above and beyond mentorship.

I want to express my gratitude to Dr Pedro Salgueiro, who has always supported me with my ICT-related difficulties. You always offered your professional judgment and pointed me on the appropriate route, no matter how minor. I truly wish to accumulate as much as domain knowledge you possess. Once again, I want to thank you for being there.

I want to thank Dr João Martins and Dr Pedro Pereira for helping me over the past year by making sure I never have too much project work on my plate and that I have enough time to work on my thesis. I couldn't have asked for a better working environment.

I appreciate the feedback Dr Mukesh Gordhanbhai Bhesaniya provided during the thesis editing process. Additionally, I am very grateful for everything you did for my family throughout the terrible Covid-19 period. I appreciate you from the bottom of my heart for what you did because, if I were in your position, I could not have done it or even come close to the amount of support you provided.

I want to thank all of my friends, especially Madhulika Agrawal, who encouraged me to come to Portugal in the first place, for all of her support throughout the early stages of my relocation. If I'm being really honest, Md Sajib Ahmed and his wife Sharmin Sultana Prite were the reason I was able to finish my thesis. You have supported me in every way necessary through thick and thin, ensuring I never go without food while I work. I also want to express my gratitude to Laura Plugge who helped me during proofreading, and to everyone who helped me in any manner but whom I was unable to name.

My brother, Kunal Raiyani, to whom I devote my thesis. We both understand that life has never been easy, and it takes a brother like you to protect me and provide a supportive and stress-free atmosphere. I couldn't have accomplished anything in my life without you. In the same way that a circle cannot exist without a center, neither could I exist without you, and for that, I am really grateful. Also, thank you Khushboo for taking care of Kunal.

Finally, to my parents, who lived their entire life selflessly providing for us and making us capable of achieving our ambitions. You endured a perilous Covid-19 period, but you persisted long enough to see your son's doctoral graduation, and for that, I earnestly thank you. I hope and pray for your well-being and will do everything in my power to ensure that.

Contents

Contents	vii
List of Figures	xi
List of Tables	xv
Acronyms	xvii
Abstract	xxi
Sumário	xxiii
1 Introduction	1
1.1 Context	1
1.1.1 Active Learning	1
1.1.2 Uncertainty Prediction	2
1.1.3 Earth Observation	3
1.2 Motivation	4
1.3 Research Questions	6
1.4 Proposed Approach	6
1.5 Main Contributions	7
1.6 Research Methodology	8
1.7 Document Structure	9
2 Earth Observation	13
2.1 Sentinel Mission	14
2.2 Land Usage and Land Cover	17
2.2.1 Monitoring	18
2.2.2 Case Studies	24
2.3 Image Scene Classification	32
2.3.1 Object-based classification	33
2.3.2 Pixel-based classification	33

2.3.3	Rule-based classification algorithm	36
2.4	Summary	37
3	Active Learning	39
3.1	Sampling Method	40
3.1.1	Membership-based sampling	40
3.1.2	Stream-based sampling	41
3.1.3	Pool-based sampling	43
3.2	Query Selection Methods	44
3.2.1	Committee	44
3.2.2	Expected Error Reduction	45
3.2.3	Uncertainty	46
3.3	Baseline Specification	47
3.3.1	Initial Training Sample Selection	47
3.3.2	Myopic vs. Batch Mode	48
3.3.3	Batch Size	49
3.3.4	Batch Diversity	50
3.3.5	Labeling Cost	51
3.4	Active Learning and Application	52
3.4.1	Remote Sensing	52
3.4.2	Abbreviating Training Cost	58
3.5	Summary	59
4	Evidence Function Model	61
4.1	Confidence Estimation and Notion of Evidence	62
4.2	Mahalanobis Distance	64
4.3	Evidence Function Model	64
4.4	Summary and Application of EFM	70
5	Experimental Datasets	71
5.1	Image Scene Dataset	71
5.2	Waterbody Dataset	76
5.3	Unlabeled Dataset	78
5.4	Summary	79
6	Image Scene Classification: Modeling and Results	81
6.1	Machine Learning Modeling	82
6.2	Experimental Setup	84

6.3	Classification Results	86
6.4	Discussion	92
6.5	Image Scene Classification: Bands or Spectral Indices	95
6.6	Atmospheric Disturbance Identification	98
6.7	Summary	100
6.7.1	Limitation	100
7	Misclassification Detection: Modeling and Results	101
7.1	EFM Modeling	102
7.2	Experimental Setup	107
7.3	EFM Results	108
7.3.1	Image scene dataset results	108
7.3.2	Waterbody dataset results	112
7.3.3	Unlabeled dataset results	114
7.4	Discussion	118
7.5	Summary	121
7.5.1	Limitation	121
8	Abbreviating Train Cost: Modeling and Results	123
8.1	Sampling Algorithm	124
8.2	Selection Methods	125
8.2.1	Entropy based method	125
8.2.2	Mahalanobis distance based method	125
8.3	Modeling Active Learning	126
8.4	Experimental Setup	126
8.5	Training Cost Reduction Results	127
8.6	Discussion	128
8.7	Summary	131
8.7.1	Limitation	131
9	Conclusions and Future Work	133
9.1	Conclusions	133
9.2	Future Work	135
A	Supporting Material	137
A.1	Spectral Indices	137
A.2	Lemma	140
A.3	Sensor	142
A.4	Image-wise Class Value Distribution	142

B Sentinel-2 Image Scene Classification Package	145
C EFM Waterbody dataset results	149
Bibliography	151

List of Figures

1.1	Adopted research method.	8
1.2	The structure of the thesis and the relationships between the chapters.	10
2.1	Copernicus 2.0 Sentinel expansion missions (European Space Agency, 2022a).	15
2.2	Top-of-Atmosphere and Bottom-of-Atmosphere (Mousivand et al., 2015).	17
2.3	Biophysical regions of Willamette Basin, Oregon, USA.	24
2.4	Willamette Basin Map Use - Crop Species vs Economic Returns (Polasky et al., 2008).	25
2.5	Land Use and Land Cover in Portugal (Meneses et al., 2018).	26
2.6	Main LUC dynamics in mainland Portugal (Meneses et al., 2018).	27
2.7	Spatial distribution of crops - number of crops (Patel and Oza, 2014).	28
2.8	Spatial distribution of crops - single crop (Patel and Oza, 2014).	29
2.9	Spatial distribution of crops - double crop (Patel and Oza, 2014).	29
2.10	The annual land cover map series of northeast China based on multi-temporal Landsat imagery (Zhao et al., 2019).	30
2.11	Mashhad basin in the northeast of Iran (Pareeth et al., 2019).	31
2.12	Sen2cor Cloud and Snow mask algorithm.	37
3.1	Membership-based sampling active learning life cycle (Settles, 2009).	41
3.2	Stream-based sampling active learning life cycle (Settles, 2009).	42
3.3	Pool-based sampling active learning life cycle (Settles, 2009).	43
4.1	Example of Mahalanobis distances. A distribution of points described by 2 attributes (the X and Y axis).	65
4.2	Generalized problem statement.	66
4.3	EFM: main modules and data-flow.	67
4.4	<i>Train set Engineering</i> : overall process flow, step 1 to step 4.	68
4.5	<i>Test set Engineering</i> : overall process flow, step 1 and step 2.	69
5.1	Global distribution of scenes.	72

5.2	Image Scene dataset generation process.	73
5.3	Class-wise band surface reflectance (ρ) value distribution.	75
5.4	Waterbody in Kazakhstan (Escobar, 2020). Left: true color image, Right: mask.	77
5.5	Waterbody dataset generation process.	77
5.6	Fiji RGB image between ($17^{\circ}11'04''$ S , $176^{\circ}59'59''$ E) and ($18^{\circ}10'26''$ S, $178^{\circ}02'16''$ E) coordinates.	78
5.7	Portugal RGB image between ($38^{\circ}50'56''$ N , $9^{\circ}00'00''$ W) and ($37^{\circ}51'10''$ N, $7^{\circ}45'07''$ W) coordinates.	78
6.1	Decision tree structure (Charbuty and Abdulazeez, 2021).	82
6.2	Proposed Convolutional Neural Network (CNN) architecture (Raiyani et al., 2021).	86
6.3	RGB image of Lautoka Area, Fiji between ($17^{\circ}42'58''$ E , $177^{\circ}35'46''$ S) and ($18^{\circ}03'24''$ E, $177^{\circ}54'01''$ S) coordinates.	89
6.4	Extra Tree classified image of Lautoka Area, Fiji. Color Labels: Cloud (White), Shadow (Brown), Other (Green).	90
6.5	Sen2Cor classified image of Lautoka Area, Fiji. Color Labels: Cloud (White), Shadow (Brown), Other (Green).	91
6.6	A Coastal Area Image (Lisbon, Portugal, between ($38^{\circ}29'28''$ N , $8^{\circ}55'$ W) and ($38^{\circ}26'11''$ N, $8^{\circ}49'18''$ W)) with brighter surface reflectance. Color Labels: Water (Blue), Cloud (White), Other (Green)	92
6.7	Ballyhaunis, Ireland, area between ($54^{\circ}04'02''$ N , $8^{\circ}50'03''$ W) and ($54^{\circ}01'$ N, $8^{\circ}44'44''$ W) coordinates.	93
6.8	Sukabumi, Indonesia, area between ($6^{\circ}37'13''$ S , $106^{\circ}53'43''$ E) and ($6^{\circ}38'22''$ S, $106^{\circ}55'55''$ E) coordinates.	93
6.9	Béja, Tunisia, area between ($36^{\circ}57'45''$ N , $9^{\circ}45'21''$ E) and ($36^{\circ}57'45''$ N, $9^{\circ}48'08''$ E) coordinates.	93
6.10	Ten corn parcels from Alentejo Region, Portugal between ($37^{\circ}56'29.13''$ N, $8^{\circ}22'21.95''$ W) and ($37^{\circ}55'32.44''$ N, $8^{\circ}21'02.23''$ W) coordinates.	98
6.11	Mean NDVI value for parcel-1 from 05-01-2017 to 03-08-2019.	99
6.12	Parcel-1: Mean NDVI and atmospheric disturbance identification by ML (over dates 14-06-2017 to 01-12-2017).	99
7.1	Problem-specific EFM modeling: main modules and data-flow.	102
7.2	Process illustration: generation of Evidence Function Model using Mahalanobis distances between train and test sets.	106
7.3	Misclassification Detection of KNN model.	109
7.4	Misclassification Detection of ET model.	110
7.5	Misclassification Detection of CNN model.	111

7.6	Classification represent the output from the ML classifier (KNN, ET and CNN), and Misclassification Detection shows the error detected over classified images. Color labels: Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan, Other as Green, Error as Red. . . .	115
7.7	Classification represent the output from the ML classifier (KNN, ET and CNN), and Misclassification Detection shows the error detected over classified images. Color labels: Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan, Other as Green, Error as Red. . . .	116
7.8	Six Mahalanobis distance based Parallel Coordinates visualization of test set from train set: True classification vs. Misclassification for KNN, ET, and CNN classifiers.	119
8.1	A graphical performance of Entropy-based method (M_{en}) based sampling strategy for various batch sizes E	128
8.2	A graphical performance of Mahalanobis distance method(M_{md}) based sampling strategy for various batch sizes E	129
8.3	Initial training samples selected (random vs grouped).	129
8.4	Image scene dataset: class-wise surface reflectance value distribution over 13 Bands (Figure 5.3 from Section 5.1).	130
B.1	Package Processing Steps: Classifying Sentinel-2 L1C Product.	146
B.2	(a) L1C product (b) RGB Scene classified image using developed package. Labels—Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan and Other as Green.	147
C.1	Water Body RGB image followed by Classified and Error Detected image for KNN, ET and, CNN. Color Labels—Other as Green, Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan and, Error as Red.	150

List of Tables

2.1	Sentinel Mission.	15
2.2	Sentinel-2 Spectral Bands.	16
2.3	Vegetation Index and their Applications in Agriculture.	20
2.4	Examples of Ecological Variables and Data Sources useful for Quantifying and Modeling Biodiversity.	23
2.5	Object-based LULC Classification Survey.	34
2.6	Pixel-based LULC Classification Survey.	34
2.7	Pixel-based LULC Change Detection Survey.	35
2.8	List of Sen2Cor Scene Classification Classes and Corresponding Colors (European Space Agency, 2020).	36
3.1	Baseline Specification Review Summary.	53
3.2	Literature Review of Active Learning and Remote Sensing Application.	57
4.1	Notation and their explanation.	66
5.1	Selected Products Geographical Distribution.	72
5.2	Surface Types and Overall Distribution of Classes.	72
5.3	Image Scene dataset - Class Mapping for Sen2Cor Assessment.	74
5.4	Number of points with Surface Reflectance (ρ) greater than 1.0.	76
6.1	Train and Test sets: Class-wise Point Distribution (%).	84
6.2	Fine-tune Parameter values for Random Forests (RF) and Extra Trees (ET) Algorithms.	85
6.3	Fine-tune Parameter values for KNN Algorithm.	85
6.4	Precision and Recall Results over the test set: Random Forest (RF), Extra Trees (ET), K-Nearest Neighbors (KNN), Convolutional Neural Network (CNN) and Sen2Cor (SCL).	87
6.5	Micro-F1 Results over the test set: Random Forest (RF), Extra Trees (ET), K-Nearest Neighbors (KNN), Convolutional Neural Network (CNN) and Sen2Cor (SCL).	87
6.6	Scene Biasness Test Results: $F1_{avg}$ values of ML algorithms and Sen2Cor.	94

6.7	Classes and Used Spectral Indices.	96
6.8	micro-F1 with Spectral Indices and 13 Bands.	97
6.9	micro-F1 using 10 Bands (not included bands - 6/7/8A).	97
7.1	Sentinel-2 image scene classification misclassified points (in %).	102
7.2	Evidence Function Model: Fine-tune Parameter values for Extra Trees (ET) Algorithm.	107
7.3	Experimental Setup and Time Specifications.	107
7.4	Misclassification vs. Misclassification Detection of KNN model.	108
7.5	Misclassification vs. Misclassification Detection of ET model.	109
7.6	Misclassification vs. Misclassification Detection of CNN model.	110
7.7	Precision and Recall of EFM in-detecting the misclassification of KNN, ET, and CNN models.	111
7.8	F1 of EFM in-detecting the misclassification of KNN, ET, and CNN models.	112
7.9	Misclassification vs. Misclassification Detection of KNN, ET, and CNN models.	113
7.10	Precision, Recall, and micro-F1 of EFM in-detecting the misclassification of KNN, ET, and CNN models.	113
7.11	Image Scene Classification - 13 Bands vs. Mahalanobis distances.	120
7.12	KNN, ET, and CNN: result comparison between model trained using 13 bands vs. 13 bands + 6 Mahalanobis distance over test set.	120
8.1	Experimental setup.	126
8.2	Extra Trees (ET) Algorithm Parameters.	127
8.3	Total number of labels added to reach micro-F1 for different batch sizes E	128
A.1	Sensor and Reference.	142
A.2	Train set (50) Products Wise Class Value Distribution.	143
A.3	Test set (10) Products Wise Class Value Distribution.	144

Acronyms

AI	Artificial Intelligence
AID	Aerial Image Dataset
AL	Active Learning
ANN	Artificial Neural Network
AOT	Aerosol Optical Thickness
AVHRR	Advanced Very High-Resolution Radiometer
BI	Brightness Index
BMAL	Batch Mode Active Learning
BOA	Bottom-of-Atmosphere
BSI	Bare Soil Index
CAP	Common Agriculture Policy
CATD	Canopy Air Temperature Difference
CCF	Canonical Correlation Forest
CNN	Convolutional Neural Network
CTF	Controlled Traffic Farming
CTV	Canopy Temperature Variability
CWSI	Crop Water Stress Index
DBN	Deep Belief Network
DGT	Directorate General of Training
DT	Decision Tree
EEA	European Environment Agency
EF	Evaporative Fraction
EFM	Evidence Function Model
ELAI	Effective Leaf Area Index
ESA	European Space Agency
ET	Extra Trees
EU	European Union
FFNN	Feed-forward Neural Network
Fmask	Function of Mask

GI	Greenness Index
GIS	Geographic Information Systems
GMES	Global Monitoring for Environment and Security
GNDVI	Green Normalised Difference Vegetation Index
GNSS	Global Navigation Satellite System
GPR	Gaussian Process Regression
GPS	Global Positioning System
GYURI	General Yield Unified Reference Index
ICIMOD	International Centre for Integrated Mountain Development
ISP	Instrument Source Packet
K-NN	k-Nearest Neighbors
KRR	Kernel Ridge Regression
L1C	Level-1C
L2A	Level-2A
LAI	Leaf Area Index
LIME	Local Interpretable Model-agnostic Explanations
LUC	Land Use Changes
LULC	Land Usage and Land Cover
MAJA	Maccs-Atcor Joint Algorithm
MAL	Myopic Active Learning
MDDC	Multi Date Direct Comparison
MESMA	Multiple Endmember Spectral Mixture Analysis
MG	Multivariate Gaussians
ML	Machine Learning
MLC	Maximum Likelihood
MSAVI2	Modified Soil Adjusted Vegetation Index
MSDF	Multi-Sensor Data Fusion
NDII	Normalised Difference Infrared Index
NDSI	Normalized Difference Snow Index
NDSII	Normalized Difference Snow Ice Index
NDVI	Normalized Difference Vegetation Index
NDWI	Normalised Difference Water Index
NIR	Near Infrared
NN	Neural Networks
NPCI	Normalised Pigment Chlorophyll Ratio Index
OBC	Object-based Classification

OOB Out of Bag
OSM OpenStreetMap
PAC Presumably Approximately Accurate
PBC Pixel-based Classification
PCA Principal Component Analysis
PVI Perpendicular Vegetation Index
QBB Query by Bagging
QBC Query by Committee
RBKF Radial Basis Function Kernel
RF Random Forest
RRI Relative Reflectance Index
RQ Research Questions
RVI Ratio Vegetation Index
SAVI Soil Adjusted Vegetation Index
SCL Scene Classification
SEI Shadow Enhancement Index
SI Stress Index
SIWSI Short wave Infrared Water Stress Index
ST Surface Temperature
SVDI Saturation Value Different Index
SVM Support Vector Machine
SWI Sentinel-2 Water Index
SWIR Specific Wavelength Range
TB Terabyte
TCI Temperature Crop Index
TCP Transductive Conformal Predictor
TGI Triangular Greenness Index
TM Thematic Mapper
TOA Top-of-Atmosphere
VCI Vegetation Condition Index
VGI Volunteered Geographic Information
VHGR Variational Heteroscedastic Gaussian Regression
VPMs Vegetation Phenology Metrics
VRA Variable Rate Application
VRE Vegetation Red Edge
WDI Water Deficit Index
WV Water Vapour

Abstract

Mapping land cover and land usage dynamics are crucial in remote sensing since farmers are encouraged to either intensify or extend crop use due to the ongoing rise in the world's population. A major issue in this area is interpreting and classifying a scene captured in high-resolution satellite imagery. Several methods have been put forth, including neural networks which generate data-dependent models (i.e. model is biased toward data) and static rule-based approaches with thresholds which are limited in terms of diversity (i.e. model lacks diversity in terms of rules). However, the problem of having a machine learning model that, given a large amount of training data, can classify multiple classes over different geographic Sentinel-2 imagery that out scales existing approaches remains open.

On the other hand, supervised machine learning has evolved into an essential part of many areas due to the increasing number of labeled datasets. Examples include creating classifiers for applications that recognize images and voices, anticipate traffic, propose products, act as a virtual personal assistant and detect online fraud, among many more. Since these classifiers are highly dependent from the training datasets, without human interaction or accurate labels, the performance of these generated classifiers with unseen observations is uncertain. Thus, researchers attempted to evaluate a number of independent models using a statistical distance. However, the problem of, given a train-test split and classifiers modeled over the train set, identifying a prediction error using the relation between train and test sets remains open.

Moreover, while some training data is essential for supervised machine learning, what happens if there is insufficient labeled data? After all, assigning labels to unlabeled datasets is a time-consuming process that may need significant expert human involvement. When there aren't enough expert manual labels accessible for the vast amount of openly available data, active learning becomes crucial. However, given a large amount of training and unlabeled datasets, having an active learning model that can reduce the training cost of the classifier and at the same time assist in labeling new data points remains an open problem.

From the experimental approaches and findings, the main research contributions, which concentrate on the issue of optical satellite image scene classification include: building labeled Sentinel-2 datasets with surface reflectance values; proposal of machine learning models for pixel-based image scene classification; proposal of a statistical distance based Evidence Function Model (EFM) to detect ML models misclassification; and proposal of a generalised sampling approach for active learning that, together with the EFM enables a way of determining the most informative examples.

Firstly, using a manually annotated Sentinel-2 dataset, Machine Learning (ML) models for scene classification were developed and their performance was compared to Sen2Cor

– the reference package from the European Space Agency – a micro-F1 value of 84% was attained by the ML model, which is a significant improvement over the corresponding Sen2Cor performance of 59%. Secondly, to quantify the misclassification of the ML models, the Mahalanobis distance-based EFM was devised. This model achieved, for the labeled Sentinel-2 dataset, a micro-F1 of 67.89% for misclassification detection. Lastly, EFM was engineered as a sampling strategy for active learning leading to an approach that attains the same level of accuracy with only 0.02% of the total training samples when compared to a classifier trained with the full training set.

With the help of the above-mentioned research contributions, we were able to provide an open-source Sentinel-2 image scene classification package which consists of ready-to-use Python scripts and a ML model that classifies Sentinel-2 L1C images generating a 20m-resolution RGB image with the six studied classes (Cloud, Cirrus, Shadow, Snow, Water, and Other) giving academics a straightforward method for rapidly and effectively classifying Sentinel-2 scene images. Additionally, an active learning approach that uses, as sampling strategy, the observed prediction uncertainty given by EFM, will allow labeling only the most informative points to be used as input to build classifiers.

Keywords: Sentinel-2; High-resolution Imagery; Scene Classification; Sen2Cor; Surface Reflectance; Artificial Intelligence; Machine Learning; Mahalanobis Distance; Classification Prediction Error; Active Learning; Training Data Reduction

Sumário

Uma Abordagem de Previsão de Incerteza para Aprendizagem Ativa – Aplicação à Observação da Terra

O mapeamento da cobertura do solo e a dinâmica da utilização do solo são cruciais na detecção remota uma vez que os agricultores são incentivados a intensificar ou estender as culturas devido ao aumento contínuo da população mundial. Uma questão importante nesta área é interpretar e classificar cenas capturadas em imagens de satélite de alta resolução. Várias aproximações têm sido propostas incluindo a utilização de redes neuronais que produzem modelos dependentes dos dados (ou seja, o modelo é tendencioso em relação aos dados) e aproximações baseadas em regras que apresentam restrições de diversidade (ou seja, o modelo carece de diversidade em termos de regras). No entanto, a criação de um modelo de aprendizagem automática que, dada uma grande quantidade de dados de treino, é capaz de classificar, com desempenho superior, as imagens do Sentinel-2 em diferentes áreas geográficas permanece um problema em aberto.

Por outro lado, têm sido utilizadas técnicas de aprendizagem supervisionada na resolução de problemas nas mais diversas áreas de devido à proliferação de conjuntos de dados etiquetados. Exemplos disto incluem classificadores para aplicações que reconhecem imagem e voz, antecipam tráfego, propõem produtos, atuam como assistentes pessoais virtuais e detetam fraudes online, entre muitos outros. Uma vez que estes classificadores são fortemente dependente do conjunto de dados de treino, sem interação humana ou etiquetas precisas, o seu desempenho sobre novos dados é incerta. Neste sentido existem propostas para avaliar modelos independentes usando uma distância estatística. No entanto, o problema de, dada uma divisão de treino-teste e um classificador, identificar o erro de previsão usando a relação entre aqueles conjuntos, permanece aberto.

Mais ainda, embora alguns dados de treino sejam essenciais para a aprendizagem supervisionada, o que acontece quando a quantidade de dados etiquetados é insuficiente? Afinal, atribuir etiquetas é um processo demorado e que exige perícia, o que se traduz num envolvimento humano significativo. Quando a quantidade de dados etiquetados manualmente por peritos é insuficiente a aprendizagem ativa torna-se crucial. No entanto, dada uma grande quantidade de dados de treino não etiquetados, ter um modelo de aprendizagem ativa que reduz o custo de treino do classificador e, ao mesmo tempo, auxilia a etiquetagem de novas observações permanece um problema em aberto.

A partir das abordagens e estudos experimentais, as principais contribuições deste trabalho, que se concentra na classificação de cenas de imagens de satélite óptico incluem: criação de conjuntos de dados Sentinel-2 etiquetados, com valores de refletância de superfície; proposta de modelos de aprendizagem automática baseados em pixels para classificação

de cenas de imagens de satélite; proposta de um Modelo de Função de Evidência (EFM) baseado numa distância estatística para detetar erros de classificação de modelos de aprendizagem; e proposta de uma abordagem de amostragem generalizada para aprendizagem ativa que, em conjunto com o EFM, possibilita uma forma de determinar os exemplos mais informativos.

Em primeiro lugar, usando um conjunto de dados Sentinel-2 etiquetado manualmente, foram desenvolvidos modelos de Aprendizagem Automática (AA) para classificação de cenas e seu desempenho foi comparado com o do Sen2Cor – o produto de referência da Agência Espacial Europeia – tendo sido alcançado um valor de micro-F1 de 84% pelo classificador, o que representa uma melhoria significativa em relação ao desempenho Sen2Cor correspondente, de 59%. Em segundo lugar, para quantificar o erro de classificação dos modelos de AA, foi concebido o Modelo de Função de Evidência baseado na distância de Mahalanobis. Este modelo conseguiu, para o conjunto de dados etiquetado do Sentinel-2 um micro-F1 de 67,89% na deteção de classificação incorreta. Por fim, o EFM foi utilizado como uma estratégia de amostragem para a aprendizagem ativa, uma abordagem que permitiu atingir o mesmo nível de desempenho com apenas 0,02% do total de exemplos de treino quando comparado com um classificador treinado com o conjunto de treino completo.

Com a ajuda das contribuições acima mencionadas, foi possível desenvolver um pacote de código aberto para classificação de cenas de imagens Sentinel-2 que, utilizando num conjunto de scripts Python, um modelo de classificação, e uma imagem Sentinel-2 L1C, gera a imagem RGB correspondente (com resolução de 20m) com as seis classes estudadas (Cloud, Cirrus, Shadow, Snow, Water e Other), disponibilizando à academia um método direto para a classificação de cenas de imagens do Sentinel-2 rápida e eficaz. Além disso, a abordagem de aprendizagem ativa que usa, como estratégia de amostragem, a deteção de classificação incorreta dada pelo EFM, permite etiquetar apenas os pontos mais informativos a serem usados como entrada na construção de classificadores.

Palavras chave: Sentinel-2; Imagens de alta resolução; Classificação de Cenas; Sen2Cor; Refletância de Superfície; Inteligência Artificial; Aprendizagem Automática; Distância Mahalanobis; Erro de Previsão de Classificação; Aprendizagem Ativa; Redução de dados de treino

Chapter 1

Introduction

“The greatest challenge to any thinker is stating the problem in a way that will allow a solution.”

— Bertrand Russell

Human civilization is gradually altering the Earth system. Identifying possible repercussions and preventing negative ones using Artificial Intelligence (AI) based solutions is becoming crucially influential (Dwivedi et al., 2021). At the same time, when using AI based solutions, quantifying errors in those solutions is critical for the success and dependability of AI applications (Tavazza et al., 2021). Furthermore, AI based systems require a large amount of training data; in this context, employing limited training data may result in faster and less expensive AI based solutions (Felderer and Ramler, 2021).

Thus, a discussion of the thesis title, “An Uncertainty Prediction Approach for Active Learning - Application to Earth Observation,” is presented next, where Active Learning, Uncertainty Prediction, and Earth Observation are the three domains.

1.1 Context

This section attempts to address three questions for each of the Active Learning, Uncertainty Prediction, and Earth Observation domains: *what*, *when*, and *where*.

1.1.1 Active Learning

What is Active Learning in Machine Learning?

In Machine Learning (ML), “Active Learning” (AL) allows a learning algorithm to engage with a user to classify incoming data points with the intended responses, when there is a large amount of unlabeled data, yet human labeling is expensive. Learning algorithms can actively ask the user for labels in this situation. Active learning is the term coined for this iterative supervised learning method (Settles, 2010).

When is Active Learning Valuable?

So, while having training data is required for machine learning, what happens if you don't have enough data? After all, adding labels to unlabeled datasets is a time-consuming procedure that may need substantial human work before you can even begin supervised ML training.

Here is where active learning comes into play. Through active learning, an algorithm may scan unlabeled training data and choose just the most significant data points for labeling. Because the algorithm only picks the data points necessary for training, the total number of data points required for analysis is far lower than in traditional supervised learning (Das et al., 2016).

Where is Active Learning Used?

Let the examples below help us understand where active learning can be used:

Active learning is a fitting choice for a variety of remote sensing applications, including the detection of local surface changes. Here, changes are infrequent, and their appearance is diverse and scattered, making it difficult to collect a representative training set ahead of time (Campbell and Wynne, 2011).

As an example, building natural language processing models necessitates training datasets that have been tagged to represent portions of speech, named things, and so on. It can be difficult to find datasets that have such mentioned labeling as well as enough unique data points. Active learning has proven particularly beneficial in this regard (Thompson et al., 1999).

Active learning has also proved beneficial in medical imaging and other situations where a human annotator identifies quantity of data necessary to aid the algorithm. Although the process might be lengthy at times since the model must continually modify and retrain based on incremental labeling updates, it can still save time when compared to traditional data gathering approaches (Komura and Ishikawa, 2019).

1.1.2 Uncertainty Prediction

What is Uncertainty Prediction in Machine Learning?

Everyday scenarios deal with a wide range of uncertainty, from investment possibilities and medical diagnoses to sports tournaments and weather predictions and, in all situations, judgments are based on obtained data. Machine learning based models are frequently used for all forms of inference and decision-making, and before these systems can be used in practice, it is becoming increasingly vital to assess their reliability and efficacy. Because models' predictions are sensitive to noise and model inference mistakes, they are prone to uncertainty in predictions (Abdar et al., 2021).

In other words, in a machine learning model, evaluating prediction uncertainty entails estimating the variability in model prediction owing to uncertainty in input values and determining the contribution of dominating inputs to the variance. Thus, Uncertainty

Prediction can be defined as a process of predicting or recognizing such variance or variability by dominating inputs.

When is Uncertainty Prediction Valuable?

Quantifying uncertainty in Artificial Intelligence (AI) based substance property predictions are critical for AI applications in materials science to succeed and be reliable. While confidence intervals for ML models are regularly published, prediction intervals, or the evaluation of the uncertainty on each prediction, are not as prevalent (Zhan and Kitchin, 2022).

The assessment of uncertainty may be a critical component, especially in areas where the dependability associated with a specific prediction is significant, such as the medical sector or weather forecast, where the accuracy of the model is at the greatest expectancy.

Where is Uncertainty Prediction Used?

Knowing prediction uncertainty could also help further research areas like ‘Active Learning’ (Settles, 2009) or ‘Disagreement based Active Learning’ (Hanneke, 2014). The reasoning is that given a large amount of freely available data where the expert’s lack of manual labels is noticeable, providing a feature of ‘measurement of uncertainty’ could help in generating labeled datasets where human input is only required for data with an excessive amount of uncertainty.

1.1.3 Earth Observation

What is Earth Observation?

Earth Observation is the collection of data about the physical, chemical, and biological processes of the planet Earth. It entails keeping track of and evaluating changes in natural and man-made environments. With the development of remote-sensing satellites and more high-tech “in-situ” sensors, Earth observation has grown increasingly sophisticated in recent years. Floating buoys for monitoring ocean currents, temperature, and salinity; land stations for recording air quality and rainwater trends; sonar and radar for estimating fish and bird populations; seismic and GPS stations; and over 60 high-tech environmental satellites that scan the Earth from space are among today’s Earth observation instruments. Because of the significant influence that contemporary human civilization is having on the global environment, Earth monitoring is now more crucial than ever (Barrett, 2013).

Space-based technologies provide repeatable and trustworthy datasets, which, when paired with suitable research and development, give a unique means of acquiring knowledge about the planet. Monitoring the status and evolution of our environment, whether on land, sea, or air, and the capacity to quickly analyze situations during emergencies such as extreme weather occurrences or times of human conflict are just a few examples.

When is Earth Observation Valuable?

The Earth system is becoming progressively influenced by human civilisation. Earth observations are crucial for assessing and preventing unwanted consequences. They can also be utilized to take advantage of new opportunities, such as sustainable natural resource management. The following are some examples showing when Earth observation is valuable:

Forecasting weather and monitoring developments in biodiversity and wildlife; land-use change measurement (such as deforestation); monitoring and responsiveness to catastrophic events such as fires, floods, earthquakes, and tsunamis; agriculture, energy sources, and freshwater resources are all under management; handling new illnesses and other threats to one's health; climate change prediction, adaptation, and mitigation.

Where is Earth Observation Used?

The United Nations relies on Earth observations to carry out its missions, resolutions, and operations (Guanter et al., 2015). Satellite sensors (government or commercial organizations), air fleet (planes or helicopters) or, more recently, unmanned aerial systems (also known as drones) can all provide remotely sensed data.

In Earth observation, classifying parts of the high-resolution optical satellite images into morphological categories (e.g., land, water, cloud, etc.) is known as scene classification (Mohajerani et al., 2018). Recently, the challenge of optical satellite image scene classification has been the focal point of many researchers. Scene classification plays a key role for example, in urban and regional planning (Hashem and Balakrishnan, 2015; Rahman et al., 2012), environmental vulnerability and impact assessment (Liou et al., 2017; Nguyen and Liou, 2019) and natural disasters and hazard monitoring (Dao and Liou, 2015). Further, given the current population growth and industrial expansion needs, assessment of land-use dynamics is certainly required for the well-being of individuals.

1.2 Motivation

According to Dubovik et al. (2021), satellite remote sensing has become one of the most effective technologies for surveying the Earth at local, regional, and global spatial scales over the last five decades. The non-destructive nature of these space-based studies enables quick monitoring of the ambient atmosphere, its underlying surface, and the ocean's mixed layer. Satellite instrumentation may also study poisonous or dangerous areas without endangering humans or equipment. Detailed (but scarce) field observations are supplemented by large-scale continuous satellite observations, which give measurements of unrivaled volume and content for theoretical modeling and data assimilation.

It seems there are a great number of critical applications that rely on data from satellites (Wielicki et al., 1996) such as weather forecast, pollution monitoring, climate variability, and other applications rely on atmospheric sensing. Similarly, for coastal areas changes like, meteorological parameters and salinity, maritime ecosystem and carbon biomass, sea-level rise, coastal traffic and fisheries, and mapping of water current and underlying topography in shallow areas are all monitored via remote sensing of ocean surfaces (Fu et al.,

2019). In addition, satellite-based remote sensing of the land helps in mineral discovery, flood and drought monitoring, soil moisture, vegetation, deforestation, forest fires, agricultural monitoring, and urban planning, among other things (Jeyaseelan, 2003; Lentile et al., 2006; Xie et al., 2008; Atzberger, 2013; Kadhim et al., 2016; Zhang et al., 2017; Babaeian et al., 2019; Gao et al., 2020).

Satellite remote sensing has also been demonstrated to be an excellent instrument to collect statistical information, including orbital geomorphology (Rees and Rees, 1999); tropospheric profiles of temperature, water vapour, carbon dioxide, and other trace gases (Van der Meer et al., 2012); geology and biological compositions of the surface and atmosphere; polar caps properties such as snow, sea ice, glaciers, and melting ponds (Bhardwaj et al., 2016); particle and electromagnetic properties of the thermosphere, ionosphere, and magnetosphere (Dubovik et al., 2021).

Copernicus (2018) is the world’s third-largest data supplier and the leading producer of Earth Observation data. Every day, Copernicus transmits 20 TB of geodata; this implies that a day’s worth of Copernicus data is the equivalent of a 1.5-year-long high-definition video; Moreover, a total of 11.8 million images have been delivered by Sentinel satellites; this corresponds roughly to the population of Madrid, Berlin, Rome, and Paris added together.

Also, given the vast expanse of farmland, optical satellite images of farms are used to create maps that depict land-use dynamics and behaviour depending on time, crops, and regions, as well as under varied environmental circumstances (Joshi et al., 2016). In this context, a major issue in this area is interpreting and classifying a scene captured in high-resolution satellite imagery (Zhong et al., 2015). Several methods have been put forth, including neural networks with data-dependent limits and static rule-based thresholds with diversity as restrictions. However, the problem of “is it possible to have a machine learning model that can classify multiple classes over different geographic Sentinel-2 imagery and can out scale existing approaches?” remains open given a large amount of available data.

As a result of the vast quantity of accessible data, supervised machine learning has grown into an integral aspect of any problem-solving process throughout time (Alzubi et al., 2018). However, in the absence of human involvement or correct labels, the effectiveness of these produced classifiers is heavily dependent on the training dataset, leaving the judgment of unseen observations unknown (Attenberg et al., 2015). Researchers attempted to assess the model’s goodness-of-fit by comparing it to a large number of independent models using a statistical distance (McDonald and Marsh, 1990; Hunter et al., 2008; Read and Cressie, 2012). However, the problem of “is it possible to identify a prediction error without human interaction?” remains open. The main motivation, however, remained the lack of a publicly accessible methodology for estimating prediction error based on the relationship between the train and test sets.

While some training data is required for supervised machine learning, supervised learning algorithms will be unable to make sense of it if it is not labeled (Carneiro et al., 2007). Labeling such a massive volume of data would need a great quantity of effort and would be expensive in terms of human expertise and time. When there aren’t enough expert manual labels available for the massive volume of publicly available data, active learning becomes critical (Budd et al., 2021). However, given the current active learning approaches, the problem of “is it conceivable to create an active learning model that can reduce classifier training costs while also assisting in labeling new data points?” remains open.

Given these problem contexts, the research questions are posed next.

1.3 Research Questions

The following Research Questions (RQ) are posed:

RQ1 Can we provide an ML model that can scene classify any new image, regardless of region, using Sentinel-2 images?

RQ2 Can we provide an AI model that can detect misclassification for any new data, regardless of the classification algorithm used, without knowledge about new data?

RQ3 Can we provide an AL model that can reduce the data required for training classifiers and assist in the generation of new labeled data?

1.4 Proposed Approach

This doctorate research focuses on three research topics, Active Learning, Uncertainty Prediction, and Earth Observation. Following the above-mentioned research questions, the proposed approaches are divided as:

A Machine Learning Model for Sentinel-2 Image Scene Classification

Several approaches have been proposed, either using static rule-based thresholds with limitation of diversity or neural networks with data-dependent restrictions (Raiyani et al., 2021). We proposed an inductive method of learning from surface reflectances approach and built a Machine Learning model for image scene classification, generating classified images that reflect land dynamics and their response over different time frames, and regions with varying environmental conditions.

Further, investigating the land activity and providing support information to farmers will help to have a better understanding of the land and its dynamics for quantitative and qualitative crop production, contributing to Sustainable Development Goal 2, Zero Hunger.

Prediction Uncertainty Identification for ML Classifiers

ML classification solutions aims at performing well on the test set, but there is no method to verify the performance of the ML model when used against real-world data without human input (Raiyani et al., 2022b). Could there be a system that supports humans in judging performance with minimum human involvement? In other words, the system will detect ML models' errors. We proposed a generalised approach for detecting classifiers' misclassifications using the Mahalanobis statistical distance between train and test sets. The approach also helps in selecting the most informative points within the datasets, introducing a novel sampling strategy for Active Learning.

This generalised technique was also used to detect misclassification in Sentinel-2 classified image scenes, as well as to identify the most informative points for classifier training.

An Active Learning Approach to Abbreviate Classification Training Cost

The absence of manual labeling is noteworthy, given the large amount of publically available data. There should be a supervised learning system that uses active learning to scan unlabeled training data and select only the most significant data points for analysis. This way, the algorithm only selects data points needed for training, and the total number of data points required for analysis is much lower than in classical supervised learning (Raiyani et al., 2022a). We proposed Active Learning based generalised sampling methods which reduces the number of labeled instances needed for training an ML classifier. The proposed approach is less data-intensive.

Further, we incorporated the uncertainty identification model (from the previous module) as one of the sampling methods for reducing the label cost for Sentinel-2 image scene classification.

1.5 Main Contributions

The main contributions in this research work can be stated as:

- Two Sentinel-2 datasets, image scene and waterbody. The image scene dataset has 60 images, 6.6 million points with six classes and a matching Sen2Cor class. The waterbody dataset has 49 images, 2.3 million with a single class. For both datasets, 13 raw bands are supplied for each point.
- ML model for pixel-based image scene classification. Empirically it was shown that the ML model outperforms Sen2Cor, a rule-based image scene classification technique.
- A statistical distance based Evidence Function Model proposal to detect misclassification caused by an ML model. Empirically detection of misclassification was shown over Sentinel-2 classified image scenes.
- Proposed a generalised sampling approach for active learning, together with the Evidence Function Model, as a way of determining the most informative points for pixel-based Sentinel-2 image scene classification, resulting in a reduction in training cost.

In addition, an exhaustive assessment of the literature on Earth observation was provided, as well as documented current gaps in domain specific applications, particularly image scene classification. Similarly, the working concept of active learning was described, along with several benchmark criteria for implementation and outlined existing gaps in remote sensing and reducing training costs.

Also, published an open-source application package Raiyani (2023). A simpler technique for classifying Sentinel-2 scene images more quickly and accurately; a learning-based strategy, enabling researchers to label more datasets using the prediction uncertainty identification model. Note: our purpose in discussing and demonstrating the use of these free resources is not to prescribe the best practice or the most accurate tools and procedures for Sentinel-2 image classification and identifying prediction uncertainty, nor is it to dismiss directly obtained data.

1.6 Research Methodology

The goal of using the Research Methodology in this doctoral study is to propose a generic answer to a topic rather than to prove a theory. Then, the generalized solutions are tested against a domain-specific application.

To formulate any significant problem with a solution, a proper research methodology structure is required. The Design Science Research (DSR) is an approach to discover and identify the problem statement, which includes the purpose of research, research questions, and procedures to obtain the solution, and analyse the significance and limitation of the solution.

DSR, which is centered on practical problem solving, comprises prescriptive or knowledge, which may be utilized to create answers to complicated and relevant domain issues using the results of scientific justification (inferring, interpreting, or describing phenomena). Its fundamental aim is to produce information that may be used by experts in the domain in question to build solutions to their field problem by presenting and analyzing possible courses of action in dealing with domain challenges (Hanid, 2014).

DSR was selected as the (philosophical) approach for the reason: at the moment, much academic research is based on the theory-driven technique of descriptive knowledge (explanatory science), with the core mission of developing conceptually-valid expertise by understanding the natural or social world, or more specifically - describing, explaining, possibly predicting, and producing shared understanding (Van Aken, 2005; Voordijk, 2009).

Figure 1.1 depicts the selected Research Methodology Flow.



Figure 1.1: Adopted research method.

Area of Research. The study field should, ideally, be related to a future professional path and have the potential to aid in the attainment of career goals.

Research Question. The most important stage in research is to define the scope of the project and follow the processes in a methodical manner until a conclusion is reached based on the studied data. According to the hypothesis test, this research topic must be validated or disproved.

Background Research. This stage entails looking at comparable works in order to figure out what the researcher intends to achieve. State-of-the-art study is often undertaken by conducting research and participating in discussion groups to determine if the work has been done before, to examine comparable techniques, and to design an approach with a suitable perspective to have an influence on the area of research.

Formulate Hypothesis. To test each study question, one or more scientific hypotheses might be defined. It usually gives a study issue more clarity, precision, and concentration. Hypotheses are recast if the outcomes from the more advanced phases are unsatisfactory.

Design Experiment. This stage establishes the feasibility of the research. Engineering research frequently entails the creation of prototypes or system designs in order to develop variables that can be changed and measured, as well as the need for quantifiable research findings. Overall, thesis validation must be accomplished using this step's design experiment.

Test Hypothesis. The implementation of the prototype, data collection, and execution tests according to the predefined validation technique are all part of the hypothesis testing process. This stage highlights the necessity to alter the prototype design and retesting to confirm the result.

Analyse Results. The results of the tests may be evaluated using quantitative and qualitative analysis. This stage necessitates an in-depth study to generate conversations regarding the relevance of the findings. If the conclusion fails the tests, it must be rejected or retested, and the process should restart from Formulate Hypothesis step.

Publish Findings. The research findings indicate a step forward for the community, and it is highly suggested that the conclusions are published and a critical analysis is provided from peers. Typically, conference proceedings are released with interim results to solicit input. The collected results are the focus of articles in referred journals.

The traditional phases of the scientific process was followed during this doctoral study. Common results were submitted to reputable conferences and journals for publication, and backward loops were used when necessary.

1.7 Document Structure

This dissertation is divided into six parts. Figure 1.2 displays the thesis structure and the links between the chapters and different parts.

The first part is an introduction. The second part, which includes Chapters 2 and 3, introduces background information, the state-of-the-art, and related work in Earth observation and active learning; the third part, Chapter 4 introduces proposals for achieving the thesis goal and answering the research questions; the fourth part describes the dataset built (Chapter 5); the fifth part presents the modeling and the results obtained (Chapter 6 to Chapter 8). Finally, the fifth section discusses the findings and future work (Chapter 9).

The chapters are breakdown as follows:

Earth Observation. Details what Earth observation is, how the Sentinel mission, particularly Sentinel-2, works, what land usage and land cover (LULC) is, how LULC may be monitored, what image scene classification is and the different approaches available, and how Sen2Cor does image scene classification. It also provides a comprehensive analysis of existing methodologies for LULC and image scene classification, as well as an assessment of their strengths and drawbacks in terms of practical applications.

Active Learning. Provides an in-depth introduction to the theory of active learning, dis-

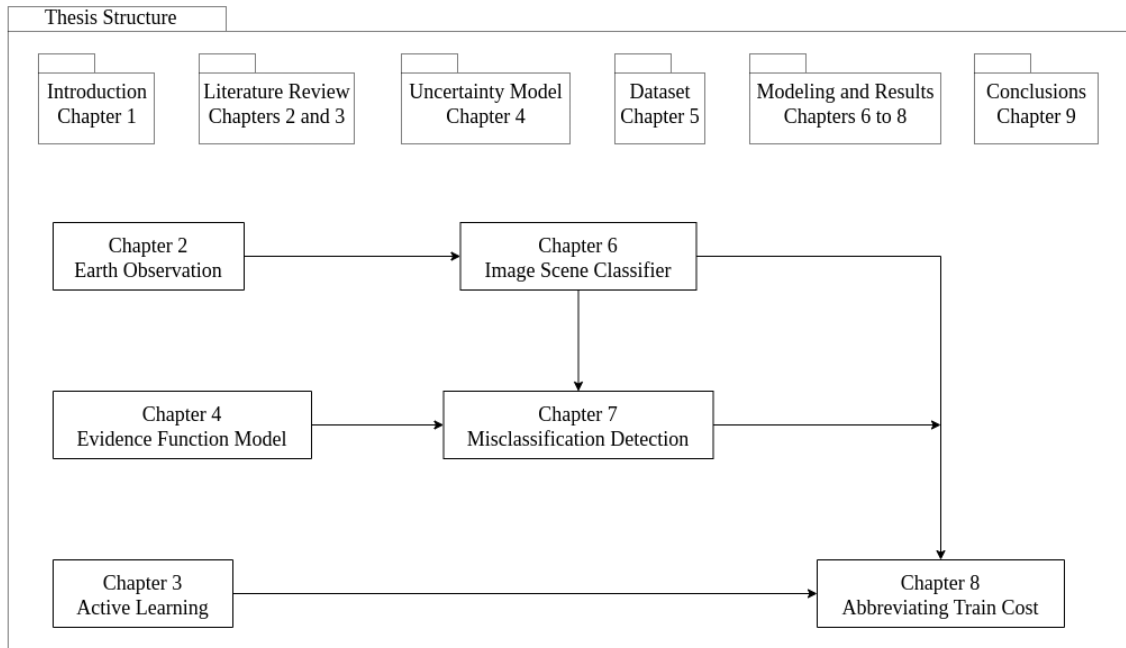


Figure 1.2: The structure of the thesis and the relationships between the chapters.

tinct sampling and query selection methods, baseline setting for active learning in practice, and the application of active learning to reduce classifier training costs. It also provides a thorough overview of current active learning approaches in remote sensing and analyzes their strengths and shortcomings in the context of operational applications.

Evidence Function Model (EFM). Discusses confidence estimates and the concept of evidence and presents a broad and statistical interpretation of it in relation to ML models; describes the Mahalanobis distance and how it is employed in the Evidence Function Model. It also specifies where the Evidence Function Model can be applied.

Experimental Datasets. Presents three separate datasets to validate the established methodologies. Image Scene dataset consists of 60 Sentinel-2 images classified into six classes; the Waterbody dataset consists of 49 Sentinel-2 images with one class and two unlabeled Sentinel-2 images.

Image Scene Classification: Modeling and Results. Describes the ML techniques used to construct classifier models, along with a comprehensive experimental setup. It also contrasts the usage of Sen2Cor vs. Sentinel-2 raw bands for image scene classification, and it last shows how the classifier model can be used in Atmospheric Disturbance Identification.

Misclassification Detection: Modeling and Results. Details EFM modeling in relation to an image scene classification problem and provides a broad grasp of how misclassification might be discovered for the same, as well as a complete experimental setup. It also evaluates the modeled EFM across a variety of experimental datasets.

Abbreviating Train Cost: Modeling and Results. Proposes and discusses a broad concept of the suggested sampling algorithm, which can be used with any query selection technique, and introduces and discusses EFM as a new query selection method and its application. It also shows how to reduce training costs in general, particularly for Sentinel-

2 image scene classifiers, and compares Entropy and EFM approaches.

Conclusions and Future Work. Provides a summary of the research's major results and discusses prospects and research implications for transferring technology to potential end users.

Chapter 2

Earth Observation

“Not everything that can be counted counts, and not everything that counts can be counted.”

— Albert Einstein

Agriculture in Europe has witness a substantial change after the creation of the Common Agriculture Policy (CAP) (Komissio, 2018) in 1962. Food security is now ensured in most parts of Europe but there is evidence that increased production has led to significant including, harmful environmental consequences in terms of water pollution, greenhouse gas emissions and damaged natural surroundings (Geiger et al., 2010; Marja et al., 2019). To face this, agricultural subsidies shifted recently from production support towards delivering public goods and services (environmental related) (Zarco-Tejada et al., 2014). However, an increase of production will be needed to sustain an estimated global population growth from the current level of about 7 billion to 9 billion by 2050 (UNESCO, 2013). Despite the apparently opposing pressures to preserve our environment and be careful with our resources (Tilman et al., 2011), the agriculture sector has to face this main challenge and produce more food. The way to address this challenge is to rely on science and technology for possible answers.

Over the last few decades many new technologies have been developed for, or adapted to, agricultural use. Examples of these include: low-cost positioning systems such as the Global Navigation Satellite System (GNSS), Geographic Information Systems (GIS), sensors mounted on agricultural machinery, geophysical sensors aimed at measuring soil properties, low-cost remote sensing techniques and reliable devices to store, process and exchange/share sensory information (Pierce and Nowak, 1999; Gibbons, 2000).

In remote sensing, Earth observation can be defined as gathering physical, chemical, and biological information about the planet surface using surveying techniques at a distance, without coming into direct physical contact (San, 2014). The principle behind remote sensing is the use of electromagnetic spectrum (visible, infrared and microwaves) for assessing Earth’s properties. The typical responses of the targets to these wavelength regions are different so that they are used for distinguishing the vegetation, bare soil, water, and other similar features (Shanmugapriya et al., 2019). It can also be used to crop growth monitoring, land use pattern and land cover changes, water resources map and water status

under field condition, monitor diseases and pest infestation, forecasting of harvest date and yield estimation, precision farming and weather forecast along with field observations. Together these new technologies have produced a large amount of affordable, high-resolution information and have led to the development of fine-scale or site-specific agricultural management that is often termed Precision Agriculture (PA). There are many aspects related to Precision Agriculture and this chapter aims at investigating the land scene classification and providing support information for a better understanding of the land and its dynamics.

Given the limitations of existing approaches, such as ground and air-based sensors in terms of time-consumed and usage, space-based satellite technologies are increasing relevance for obtaining spatio-temporal information to supplement them, making remote sensing widely used techniques for Earth observation.

In remote sensing there are two types of sensors: *passive* and *active*. Passive sensors measure radiation that reaches a detector without the sensor first transmitting a pulse of radiation; active sensors emit a pulse and later measure the energy returned or bounced back to the detector. Both passive and active sensors record the intensity of a signal within a wavelength interval, known as a *band* or *channel*, of specified width within the electromagnetic spectrum.

The remainder of the chapter is organized as follows: Section 2.1 talks about Sentinel Mission, the Copernicus Program, a part of the European Union’s Earth monitoring system; Section 2.2 details Land Usage and Land Cover (LULC), describing different monitoring parameters, along with LULC case studies; Section 2.3 shows Image Scene Classification, describing different type of classification methodology.

2.1 Sentinel Mission

The Copernicus Program, a part of the European Union’s Earth monitoring system, coordinates the Sentinel missions. The European Commission, in collaboration with European Space Agency (ESA), European Union (EU) member states, and EU agencies, funds all activities of Sentinel missions (Machado et al., 2005). The ESA and the EU launched the Global Monitoring for Environment and Security (GMES) program in 1998, which was renamed Copernicus Program in 2014 (Drusch et al., 2012). The Copernicus Programme has strategic plans for developing seven satellite missions (Sentinel-1, 2, 3, 4, 5P, 5, 6) (European Space Agency, 2022i). Table 2.1 details the various satellites under the Sentinel mission project.

Figure 2.1 shows the six high priority candidate ‘extension’ missions being researched by ESA in preparation for the second generation of Copernicus (Copernicus 2.0) to meet EU policy and gaps in Copernicus user demands, as well as to expand the current capabilities of the Copernicus Space Component (European Space Agency, 2022a). They are:

Table 2.1: Sentinel Mission.

Satellite	Service	Launch		
		Time:DD.MM.YY	Vehicle	Site
Sentinel-1	Day and night radar imaging	A:03.04.14 B:25.04.16	Syuz	Kourou
Sentinel-2	High-resolution optical imaging	A:23.06.15 B:07.03.17	Vega	Kourou
Sentinel-3	Sea surface topography	A:16.02.16 B:25.04.18	Rockot	Plesetsk Cosmodrome
Sentinel-4	Atmospheric composition	2023	—	—
Sentinel-5	Sciamachy atmospheric	13.10.17	Rockot	Plesetsk Cosmodrome
Sentinel-6	Sea surface radar imaging	A:21.11.20 B:2025	SpaceX Falcon 9	Vandenberg

1. Sentinel-7: Anthropogenic CO₂ emissions monitoring (CO₂M).
2. Sentinel-8: High Spatio-temporal Land Surface Temperature (LSTM).
3. Sentinel-9: Copernicus Polar Ice and Snow Topography Altimeter (CRISTAL).
4. Sentinel-10: Copernicus Hyperspectral Imaging Mission for the Environment (CHIME).
5. Sentinel-11: Polar Imaging Microwave Radiometer (PIMR).
6. Sentinel-12: Radar Observing System for Europe - L-band SAR (ROSE-L).

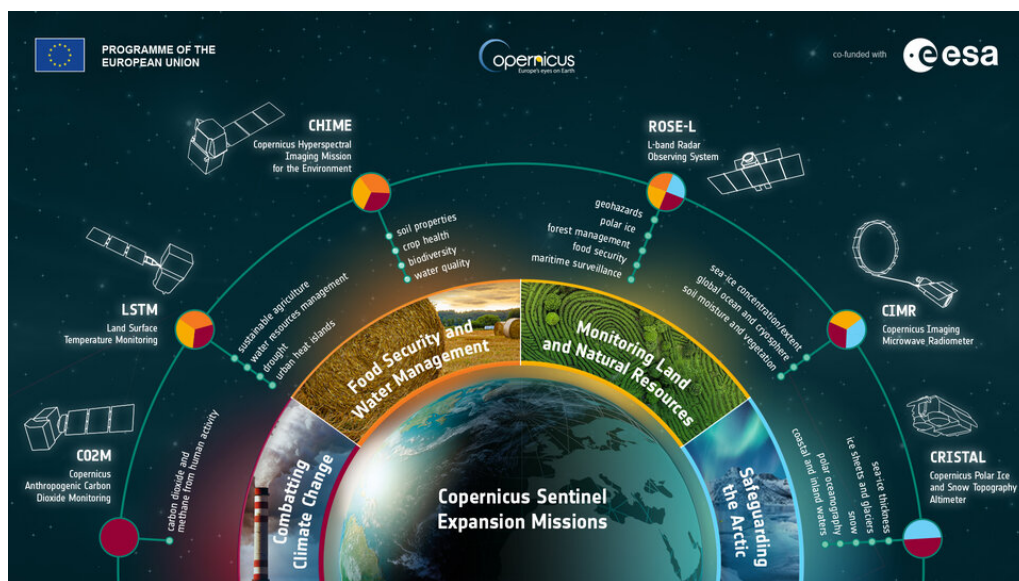


Figure 2.1: Copernicus 2.0 Sentinel expansion missions (European Space Agency, 2022a).

Sentinel-2. Sentinel-2 systematically captures high-resolution optical images across land and coastal waters. Which can be used for a wide variety of services and applications, including agricultural and forest monitoring, emergency management, land cover categorization, and water quality monitoring. Images also include floods, volcanic eruptions, and landslides to enhance disaster mapping and humanitarian relief efforts (Gutierrez et al., 2011; Vuolo et al., 2018).

The Sentinel-2 mission has the following key characteristics (European Space Agency, 2022d):

- Multi-spectral data with 13 bands in the visible, near-infrared, and short wave infrared part of the spectrum,
- Revisitation every 5 days under the same viewing angles,
- Spatial resolution of 10m, 20m, and 60m,

Sentinel-2 data represents an increment in terms of temporal and spatial resolution than previous low to medium spatial resolution aerial images (e.g., Landsat). As mentioned, Sentinel-2 images feature 13 bands with spatial resolutions ranging from 10 to 60 meters. The visible and near-infrared NIR bands have a spatial resolution of 10 meters, while the infrared bands have a resolution of 20 meters and the remaining bands have a resolution of 60 meters. Table 2.2 details the Spectral bands for the Sentinel-2 sensors. Here, VRE: Vegetation Red Edge, NIR: Near Infrared, and SWIR: Specific Wavelength Range.

Table 2.2: Sentinel-2 Spectral Bands.

Sentinel-2 Bands (B)	Sentinel-2A	Sentinel-2B	Resolution(m)
	Central Wavelength (nm)		
B1 Coastal aerosol	442.7	442.2	60
B2 Blue	492.4	492.1	10
B3 Green	559.8	559.0	10
B4 Red	664.6	664.9	10
B5 VRE	704.1	703.8	20
B6 VRE	740.5	739.1	20
B7 VRE	782.8	779.7	20
B8 NIR	832.8	832.9	10
B8A Narrow NIR	864.7	864.0	20
B9 Water vapour	945.1	943.2	60
B10 SWIR Cirrus	1373.5	1376.9	60
B11 SWIR	1613.7	1610.4	20
B12 SWIR	2202.4	2185.7	20

While capturing satellite image, the atmosphere affects the spatial and spectral distribution of the electromagnetic radiation from the Sun before it reaches the Earth's surface; subsequently, it also attenuates the reflected energy recorded by a satellite sensor. Top-of-Atmosphere (TOA) reflectance is a dimensionless quantity measurement that provides the ratio between the radiation reflected and the incident solar radiation on a given surface. Bottom-of-Atmosphere (BOA) reflectance is defined as the fraction of incoming solar

radiation that is reflected from Earth's surface for a specific incident or viewing case. Figure 2.2 shows what is TOA and BOA. Here, E_s is defined as incoming solar radiation from Sun and E_o can be defined as reflected solar radiation from Earth, and E^- and E^+ can be defined as reflection variation due to atmosphere.

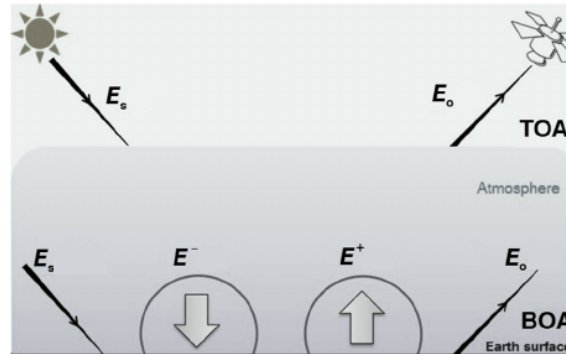


Figure 2.2: Top-of-Atmosphere and Bottom-of-Atmosphere (Mousivand et al., 2015).

Data from Sentinel-2 is available in a variety of processed formats Level-0, Level-1A, Level-1B, Level-1C, and Level-2A as Sentinel-2 products go through many steps of processing before they can be accessed by consumers. All these formats are explained as below:

- Level-0 and Level-1A are not available to users and are compressed raw picture data in instrument source packet (ISP) format (Llewellyn-Jones et al., 2001);
- Level-1B product, granules of $25 \times 23 \text{ km}^2$, comprises the revised geometry needed to create the user-accessible Level-1C products, as well as radiometrically corrected imagery with TOA radiance values;
- Level-1C is created in cartographic geometry using digital elevation models. Level-1C product is radiometrically and geometrically corrected including orthorectification (Japan Association, 2022). Radiometric and geometric correction is nothing but the calibration of pixel values and the correction of errors in those values;
- Level-2A product provides BOA reflectance images derived from the associated Level-1C product. In Sentinel-2, each Level-1C and Level-2A product is composed of $100 \times 100 \text{ km}^2$ tiles in cartographic geometry UTM/WGS84 projection (Langley, 1998; National Geospatial Intelligence Agency, 2022).

2.2 Land Usage and Land Cover

Given the current global population growth scenario, global agricultural production must be expanded further in the future years to satisfy growing demand and changing consumption habits. This will necessitate either agriculture intensification or farmland expansion. Having agricultural land cover and usage maps will aid in choosing specific crops to cultivate and giving detailed information about the behavior of land through time Pichón (1997). This brings us to the critical problem of land cover mapping and land utilization dynamics, which is critical in crop management vs biodiversity balancing. Typically, the collection of processes and necessary information varies per crop. Can generic land cover

information regarding water supplies, soil qualities, and surface energy flow for a specific region made available? or when and where to grow a certain crop? Gathering such broad data can aid in land management to preserve biodiversity and economic return. Furthermore, it may give consistent and regional agricultural growth conditions based on crop kind.

Forecasting parcel/region crop growth, for example, requires knowledge regarding parcel/region features such as crop growth, water supplies, soil parameters, and surface energy flow. Furthermore, land dynamics/properties react differently in different situations for the same crop, therefore, estimating crop output across diverse agricultural land would be advantageous.

Since the early 1990s, the number of satellites has increased dramatically, and the trend of a steady increase in the number of satellites is expected to continue in the future; this brings coverage of the planet Earth with images characterized by an ever-expanding spatial and temporal resolution, and an expanding electromagnetic spectrum; leading to underpinning information layers for a wide range of terrestrial environmental land cover and land use.

2.2.1 Monitoring

Remote sensing is an effective and reliable data source for classifying different land covers. For example, optical remote sensing images have demonstrated that vegetation types can be clearly distinguished by exploiting their spectral signature and the phenological stage at the time of the image acquisition. This application of remote sensing technologies has a clear impact to optimize production efficiency and to increase quality, but also to minimize environmental impact and risk, which includes undesirable variability caused by the human operator.

The key feature of Precision Agriculture comes from positioning systems, principally, Global Navigation Satellite System (GNSS) that are major enablers of ‘precision’ (Gebbers and Adamchuk, 2010). Precision Agriculture is most advanced amongst arable farmers, particularly with large farms and field sizes in the main grain-growing areas where a business model to maximize profitability is the main driver¹. For example, Controlled Traffic Farming (CTF) and auto-guiding systems are the most successful applications on arable land showing clear benefits in nearly all cases (Vermeulen et al., 2010). For Variable Rate Application (VRA) methods, such as optimizing fertilizer or pesticide use to areas of need, the success varies greatly according to the specific factors of the application (Clark and McGuckin, 1996).

The implementation of Precision Agriculture has become possible thanks to the development of technologies (in particular remote sensing) combined with procedures that link mapped variables to appropriate farming management actions such as cultivation, seeding, fertilization, herbicide application, and harvesting by land cover usage mapping and its dynamics.

To track agriculture land usage and cover, according to Shanmugapriya et al. (2019), the following monitoring parameters are important: Vegetation; Crop Condition; Water Status and Crop Nutrient; Crop-Land Evapo-Transpiration; Pest and Disease Infestation; Atmospheric Dynamics; Biodiversity.

¹<https://www.agric.wa.gov.au/generating-more-profit-your-farm-business?nopaging=1>

The remaining subsection discusses how the above-mentioned monitoring metrics are used to track agricultural land use and cover describing the most recent work on the topic.

Vegetation. To determine the crop condition such as nutrient stress and water availability for assessing the crop health and yield, physical parameters, indexes, of the crop system are used; a multitude of indexes have been proposed in the recent years. The Normalized Difference Vegetation Index (NDVI) (Rouse Jr et al., 1973) is the most commonly (Calva and Palmeirim, 2004; Wallace et al., 2004) used index to check the vegetation condition, but soil background and atmospheric noise do hinder the calculation of the NDVI index.

An example of a vegetation index that limits the influence of the soil on remotely sensed vegetation data is the Soil Adjusted Vegetation Index (SAVI) (Huete, 1988). Moreover NDVI, Vegetation Condition Index (VCI), Greenness Index (GI), Leaf Area Index (LAI), General Yield Unified Reference Index (GYURI), and Temperature Crop Index (TCI) are used for mapping and monitoring drought and assessment of vegetation health and productivity (Doraiswamy et al., 2005; Ferencz et al., 2004; Prasad et al., 2006).

Other indexes like the Advanced Very High-Resolution Radiometer (AVHRR) were used to model corn yield and early drought warning in China (Seiler et al., 2000a) and Hadria et al. (2006) used multiple satellites to calculate LAI aiming to estimate the distribution of yield and irrigated wheat in semi-arid areas. Table 2.3 presents a list of vegetation indexes used for agricultural land use monitoring. Appendix A.1 has the individual index formula.

Crop Condition. The health of plants can be determined by their bio-physical parameters. These can be measured through timely spectral information using remote sensing. The physiological changes due to crop stress leads to change in the spectral reflection/emission characteristics (Menon, 2012). This observation of the stress factor during the crop growth is a necessary part to know the probable loss of production.

Crop growth and its development is affected by multiple factors such as available soil moisture, date of planting, air temperature, day length, and soil condition. For example, if temperatures are too high at the time of corn pollination it will result in negatively corn crop yields. For this reason, knowing the temperature at the time of corn pollination can help forecasters better predict corn yields (Nellis et al., 2009).

The occurrence of drought also makes the land incapable for cultivation and renders inhospitable environment for human beings, livestock population, biomass potential and plant species (Siddiqui, 2003). Drought monitoring through satellite based information have been used in recent years and the analysis of the NDVI and VCI indexes have been accepted globally for identifying agricultural drought in different regions with varying ecological conditions (Nicholson and Farrar, 1994; Seiler et al., 2000b; Kogan, 1995; Wang et al., 2001; Anyamba et al., 2001; Ji and Peters, 2003).

Vegetation Phenology Metrics (VPMs) are used in characterizing agricultural vegetation response to varying climatic and land management practices (Reed et al., 1994). The term “vegetation phenology” refers to the description of periodic plant life cycle events that occur throughout the course of the growing season. Remote sensing is commonly used to track vegetation phenology using time series of vegetation indicators (Zeng et al., 2020).

Table 2.3: Vegetation Index and their Applications in Agriculture.

Reference	Application	Index	Sensor
Bausch and Khosla (2010)	Plant nitrogen and plant status	CI	QuickBird
Snyder et al. (2005)	Winter oilseed rape yield prediction	ELAI	Groundbased (CIMEL 33 radiometer)
Chang et al. (2003)	Corn yield predictions	GNDVI	Airborne Camera
Han et al. (2012)	Corn canopy and nitrogen content prediction	MSAVI2	Terra ASTER
Cheng et al. (2013)	Diurnal orchard and canopy water detection	NDII	Airborne MASTER
Zarco-Tejada et al. (2003)	Plant water content estimation	NDWI	MODIS
Hatfield and Prueger (2010)	Leaf chlorophyll content estimation	NPCI	Exotech and CropScan
Mogensen et al. (1996)	Drought of field grown and oilseed rape	RRI	LI-90s and LI-220S
Fensholt and Sandholt (2003)	Canopy water detection	SIWSI	MODIS
Jr et al. (2013)	Crop nitrogen detection	TGI	Landsat TM

Water Status and Crop Nutrient. Remote sensing and GIS play a key role in nutrient and water stress management thus helping in reducing the cost of cultivation as well as increasing the fertilizer use efficiency for the crops. The effective use of water in semi-arid and arid regions is also possible through the application of precision farming technologies. For example, [Das and Singh \(1989\)](#) stated that drip irrigation coupled with information from remotely sensed data such as canopy air temperature difference can be used to increase the water use efficiency by reducing the runoff and percolation losses. Further, they stated that higher spectral reflection was observed in water stressed crops compared to non-stressed ones. Adding to that, vegetation indexes like NDVI, Ratio Vegetation Index (RVI), Perpendicular Vegetation Index (PVI) and Greenness Index (GI) were found to be lower for stressed and higher for non-stressed crops. Furthermore, the availability of micro wave remote sensing has made possible to estimate the presence of soil moisture in the field ([Bandara, 2003](#)).

[Fang et al. \(2008\)](#) found the aspect of nitrogen leaching differently depending upon the soil properties such as soil organic matter content ([Casa et al., 2011](#)), water content ([Delin and Berglund, 2005](#)) and yield zones ([Blackmore et al., 2003](#); [Bramley, 2009](#)) under wet tropical and subtropical climates. This leads to failure of traditional equal spread of fertilizer where some sites are over-fertilized and others remain under-fertilized ([Bredemeier and Schmidhalter, 2005](#)).

Crop/Land Evapo-Transpiration. Drought is a situation which can be defined as “a long-term average condition of the balance between precipitation and evapo-transpiration in a particular area, which also depends on the timely onset of monsoon as well as its potency” ([Wilhite and Glantz, 1985](#)). The vegetation indexes such as Crop Water Stress Index (CWSI) ([Jackson et al., 1981](#)), Surface Temperature (ST) ([Jackson, 1986](#)), Water Deficit Index (WDI) ([Moran et al., 1994](#)), and Stress Index (SI) ([Vidal et al., 1994](#)) provide the relationship between water stress and thermal characteristics of the plants. Correlating land surface temperatures with the vegetation indexes can result into detecting agricultural drought of a region and provide early warning systems to the farmers ([Sruthi and Aslam, 2015](#)). Estimation of evapo-transpiration is critical for assessing the irrigation scheduling, water and energy balance computations and determining CWSI for climatological, and meteorological purposes ([Veysi et al., 2017](#)).

[Batra et al. \(2006\)](#) defined the Evaporative Fraction (EF), as the ratio of ET and available radiant energy, by successfully using AVHRR and MODIS data. Generally, Evaporative Fraction is a ratio of latent heat flux to the sum of latent and sensible heat fluxes ([Nichols and Cuenca, 1993](#)). EF is used to characterize the energy partition over land surfaces. Most of the approaches use simple direct correlations between remote sensed digital data and evapo-transpiration values ([Dutta et al., 2015](#); [Neale et al., 2005](#)).

Pest and Disease Infestation. The effect of biotic and abiotic factors over crops can be easily monitored by remote sensing. [Franklin \(2001\)](#) concluded that relating differences in spectral responses to chlorosis, yellowing of leaves and foliage reduction over a given time period, assuming that these differences can be correlated while monitoring insect defoliation, can help in the classification and interpretation of insects. Also, healthy and unhealthy vegetation cover over different types of vegetation was evaluated using Landsat imagery ([Williams et al., 1979](#)).

De Beurs and Townsend (2008) stated that “MODIS data represent an important tool for insect damaged defoliation and determination of vegetation indexes in plot scale”. According to Riedell et al. (2005), remote sensing is an effective and inexpensive method to identify pest infested and diseased plants; further, they studied detection of specific insect pests and how to differentiate insects from disease damages over oat crops. They also suggested that canopy characteristics and spectral reflection differences between insect infestation and disease infection damages can be measured in oat crop canopies by remote sensing. Similarly, Mirik et al. (2013) suggested that wheat streak mosaic disease in wheat crops can be accurately detected and quantified using Landsat 5 TM images.

Atmospheric Dynamics. Meteorological satellites are used for forecasting weather conditions. They are designed to measure the atmospheric temperature, wind, moisture and cloud cover. The variations in the canopy temperature could indicate the areas of adequate and inadequate water. The Canopy Temperature Variability (CTV) is used in irrigation management and Canopy Air Temperature Difference (CATD) is used as an indicator of crop water stress (Menon, 2012). Monitoring NDVI generated from NOAA-AVHRR (satellite) data can also help in assessing of district level drought (Lee et al., 2010).

Biodiversity. Here, the term biodiversity refers to the variety of species and certain characteristics of species, in particular their distribution and number within a given area. We also use biodiversity more broadly to mean species assemblages and ecological communities (i.e. groups of interacting and interdependent species) (Gaston, 2000). There are two general approaches to the use of remote sensing for biodiversity (Turner et al., 2003). One is the direct remote sensing of individual organisms, species assemblages, or ecological communities from airborne or satellite sensors. The other approach is the indirect remote sensing of biodiversity through reliance on environmental parameters as proxies. For example, many species are restricted to discrete habitats, such as a woodland, grassland, or sea-grass beds that can be clearly identified remotely.

In general, biodiversity is monitored using passive, like visible, near and middle-infrared, and thermal-infrared sensors; Table 2.4 provides examples of ecological variables and sensors useful for quantifying and modeling biodiversity. Source data (NASA Landsat Science, 2013; NASA Landsat Missions, 2022; NASA Earth Observatory, 2022; NASA Terra, The EOS Flagship, 2022; Johnson and Green, 1995; NASA EarthData OceanColor Web, 2022; NASA EarthData NASA Distributed Active Archive Center, 2022; Seiler et al., 2000b; NASA Modis, 2022).

Table 2.4: Examples of Ecological Variables and Data Sources useful for Quantifying and Modeling Biodiversity.

Approach	Ecological Variable	Description	Sensor
Direct	Species Composition	Used for measuring canopy unique spectral signatures	HYPERION, ASTER, IKONOS, Quickbird
	Land Cover	Can discriminate different land surfaces at various resolutions; land cover classification	TM/ETM, ALI, MODIS, ASTER, IKONOS, Quickbird
Indirect	Chlorophyll II	Assessing presence of vegetation and relative greenness measures; calculating productivity and plant health	SeaWiFS, HYPERION, TM/ETM, ALI, MODIS, ASTER, IKONOS, Quickbird
	Ocean color and circulation	Circulation patterns can be inferred from changes in ocean color, sea surface height, and ocean temperature	TOPEX/Poseidon, AVHRR, MODIS, SeaWiFS
Climate	Rainfall	Detection of precipitation and surface moisture. Used for drought management	CERES, AMSR-E
	Phenology	Leaf, turnover, flowering cycles can be inferred. identification of certain phenological species	TM/ETM, ALI, HYPERION, ASTER, IKONOS, Quickbird
Habitat	Topography	Microhabitats change detection due to altitude change	ASTER, IKONOS

2.2.2 Case Studies

To get a better understanding of the Land Usage and Land Cover maps, next we present five case studies: Willamette Basin — Presents the ecological and economic repercussions of different land-use patterns; Portugal — Presents the innovative knowledge map that aids understanding LUC dynamics within Portuguese continental area from 1990 to 2012; Gujarat — Presents MODIS data time-series over the crop year 2012/13 for deriving a crop calendar; China — Presents long-term land cover dynamics map from 1986 to 2016 of Northeast China using multi-temporal Landsat images; Iran — Presents agricultural mapping and land-use patterns of Mashhad basin area, for three crop years 2013/2014, 2014/2015, and 2015/2016;

1 Willamette Basin. Large-scale natural environment changes to human-dominated environments have led to significant losses in form biodiversity as the human population and economy grow. It is a critical challenge to conserve biodiversity while meeting expanding human demands.

For investigating the ecological and economic repercussions of different land-use patterns, [Polasky et al. \(2008\)](#) created a spatially explicit landscape level model. For terrestrial vertebrate species, the spatially explicit biological model includes habitat preferences, area needs, and dispersion abilities between habitat regions to forecast the likely number of species that will survive on the landscape. To anticipate economic returns for a variety of prospective land uses, the spatially explicit economic model includes location attributes and location.

They used a raster grid map of 10,372 parcels from the 1990 land cover Basin map to seek effective land-use patterns that optimize biodiversity conservation objectives for certain levels of economic benefits and vice versa. They discovered land-use patterns that support high levels of biodiversity while still providing economic benefits. Figure 2.3 shows the major biophysical regions of the Coast Mountain Range (Willamette Basin, Oregon, USA).

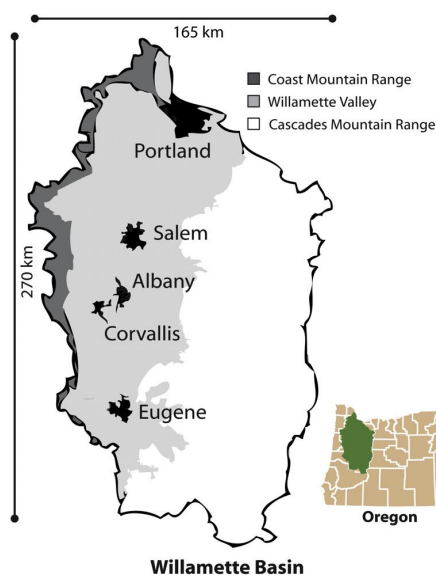


Figure 2.3: Biophysical regions of Willamette Basin, Oregon, USA.

Further, they created a database where each crop species is indexed with its market value. They were able to create this database by observing/using 1980 to 2003 rural-residential land-use value in the Willamette Basin. With this database, they were able to find an efficiency frontier for terrestrial vertebrate conservation and economic returns in the Basin area, Figure 2.4. Where moving from point ‘A’ to point ‘H’ increases the biological score and at the same time decreases the economic return; point ‘I’ represent different estimates for the biological and economic scores for the 1990 land-use pattern.

Increasing the biological score to its maximum at point ‘H’ reduces economic returns to zero. Because certain species need agricultural land as a habitat, some land is kept in agriculture, although it is very unproductive, resulting in economic losses. A tiny region of high-value rural-residential land use on the landscape offsets these economic losses at point ‘H’. Economic returns increase to \$27.6 billion at point ‘A’, but the number of species that can be supported on the terrain drops to 229.3. Approximately 27 of the 257 species modeled would not be predicted to survive in the Basin if land-use practices for the landscape at point ‘A’ were followed exclusively.

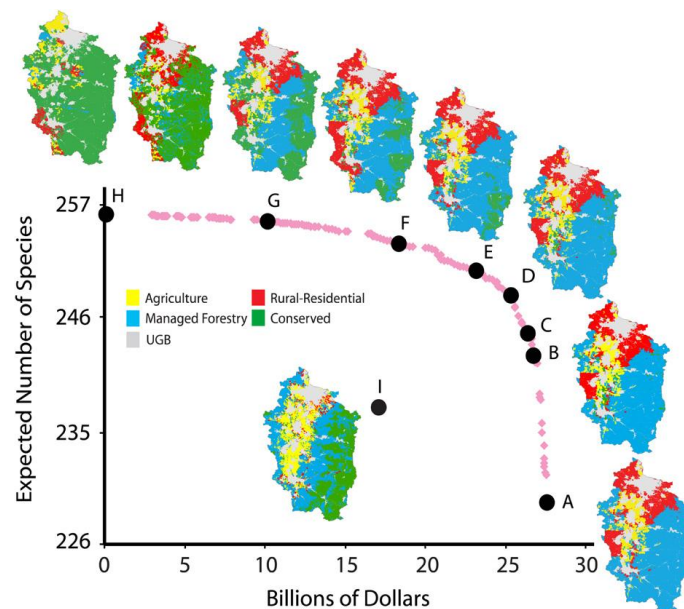


Figure 2.4: Willamette Basin Map Use - Crop Species vs Economic Returns (Polasky et al., 2008).

In a nutshell, the authors present biological and economic models that leverage land-use and land-cover patterns to investigate the combined effects of land-use decisions at a regional landscape scale. The biological model assesses how well a group of species can survive on a terrain given the land cover pattern. For a particular land-use pattern, the economic model estimates the net present value of marketable commodities and services from the landscape.

2 Portugal. The Portuguese territory has experienced relevant Land Use and Land Cover Changes (LUCC) in recent decades. According to Meneses et al. (2018), the revision of existing Land Use and Cover (LUC) datasets allowed to produce new datasets for better understanding of LUCC and LUC over time. They further studied by analyzing the most recent LUC datasets, which cover the entirety of the Portuguese continental area from 1990

to 2012 (see Figure 2.5) and present innovative knowledge that aids understanding LUC dynamics within that period (see Figure 2.6).

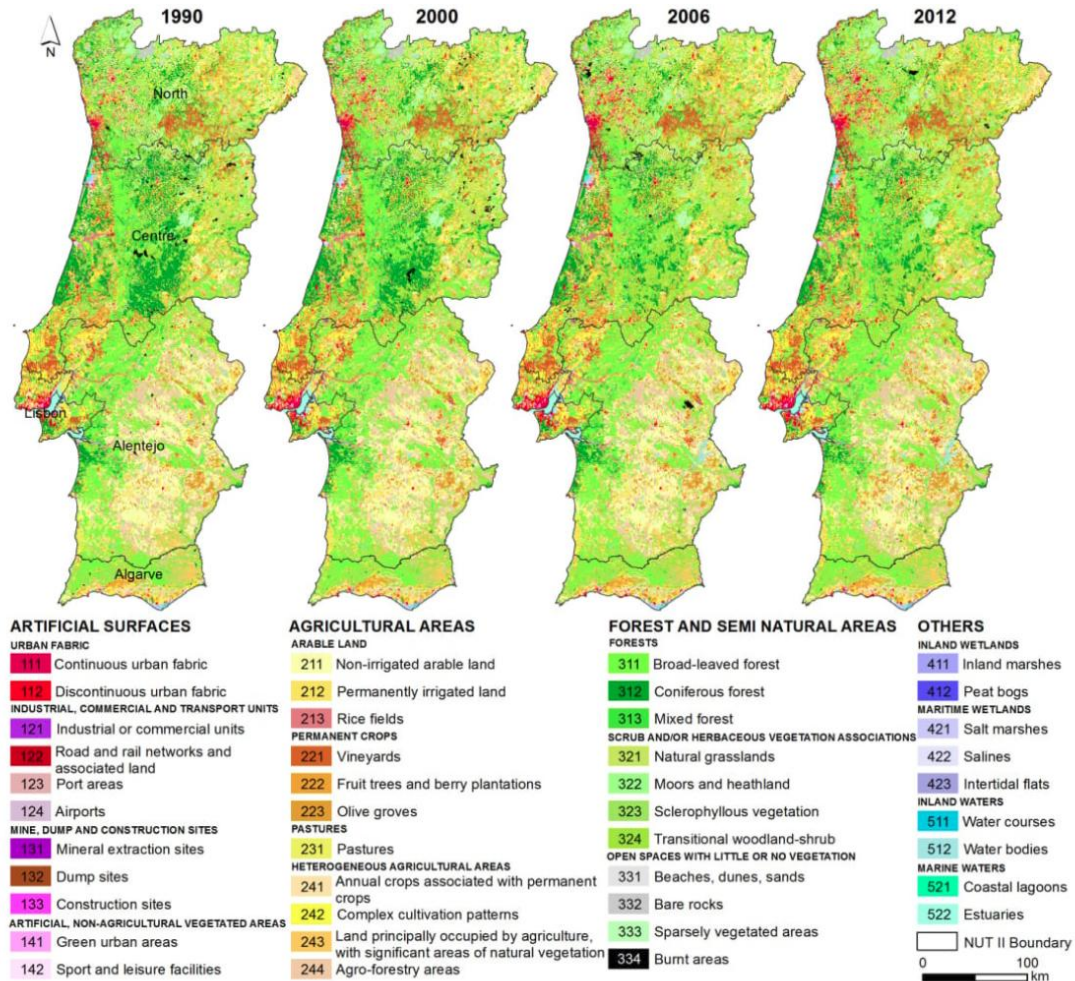


Figure 2.5: Land Use and Land Cover in Portugal (Meneses et al., 2018).

The Mainland Portugal, with an area of $88,962.50 \text{ km}^2$, is divided into five regions: North (23.8% of the area), Centre (31.6%), Lisbon (3.6%), Alentejo (35.4%) and Algarve (5.6%). To determine the LUCC and Coverage of the territory cartography, Meneses et al. (2018) used four years (1990, 2000, 2006 and 2012) maps² and predicted future trends using a CA-Markov model. Figure 2.6 shows a comprehensive examination of relative LUCC per NUTS II unit that shows diverse geographical and temporal patterns. Here, NUTS II unit represents the basic regions for the application of regional policy unit (Camagni and Capello, 2017).

²These maps are available on the European Environment Agency (EEA) and Directorate General of Training (DGT) websites.

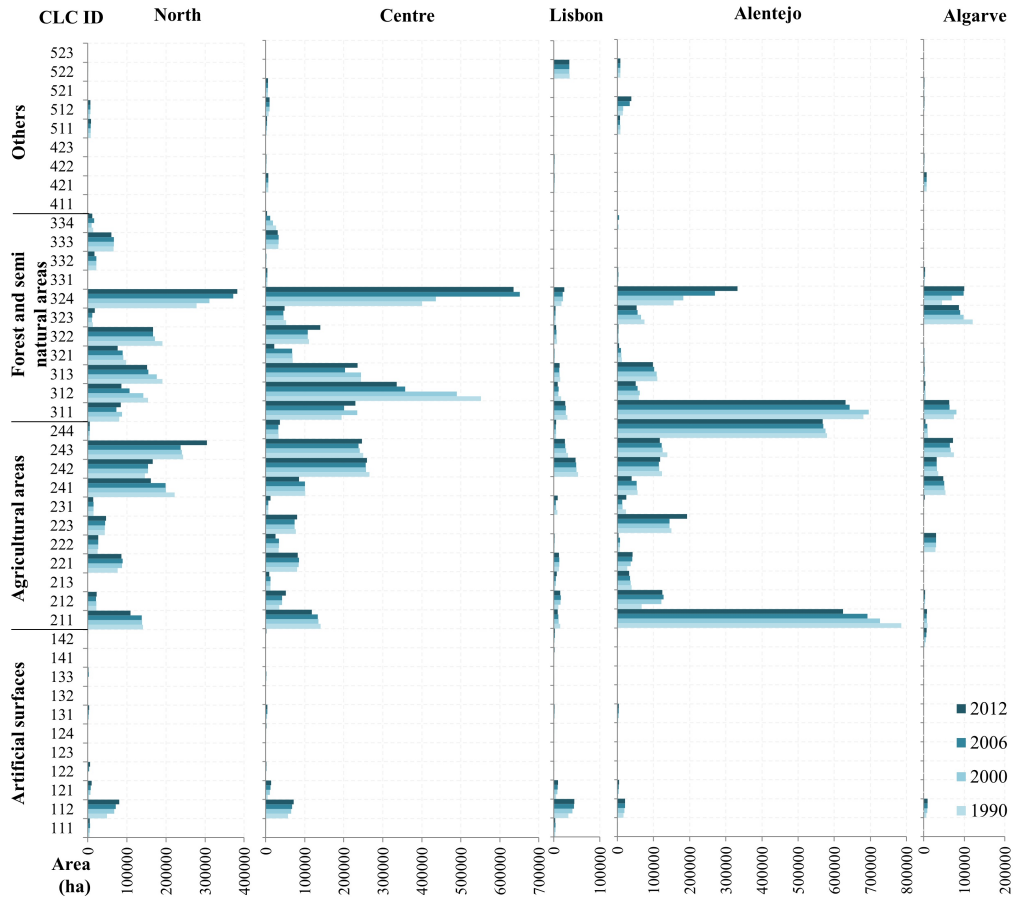


Figure 2.6: Main LUC dynamics in mainland Portugal (Meneses et al., 2018).

For example, around 1990 and 2000, a significant increase in road and rail networks and spaces associated with them were witnessed in the North and Lisbon reflecting, in part, new road infrastructure investments (Estradas de Portugal, 2015) - meanwhile the substantial rise in this LUC type were noticed in the Centre, Alentejo, and Algarve between 2000 and 2006.

In the first period, between 1990 and 2000, the area of industrial or commercial units rose in every NUTS II unit, although the gains were more significant in the north, particularly during the last term. The overall trend of the diminishing area inhabited by this LUC type found in the other NUTS II units was reversed in these regions.

During the first period, the artificial surfaces exhibited a 20.15% rise in the discontinuous urban fabric, with a significant drop in that growth in subsequent periods. Green urban spaces, on the other hand, have expanded in the Center in recent years (11.6%).

Changes in specific agricultural LUC types, such as non-irrigated arable land and irrigated land, were also detected. The 1990s saw an expansion in the permanently irrigated territory, particularly in Alentejo, where the completion of the Alqueva dam increased water availability (quantity) and permitted irrigation systems to be installed. This scenario demonstrates the impact of anthropogenic interventions on the landscape, as well as their contributions to high LUCC over a short time.

Except for burnt (ID 334) and thinly vegetated regions (ID 333) in Alentejo, the relative changes in the classes included in this LUC type were minimal in most cases (1%). During the period 2000-2006, the class of burnt regions grew by almost 20% in the same NUTS II unit, while it declined during the other periods. Furthermore, there was an increase in sparsely vegetated regions, which might be due to natural vegetation regeneration in places impacted by forest fires or abandoned agricultural land. It's crucial to note, nevertheless, that the lower relative LUCC values reflect several hectares due to the size of each NUTS II unit.

3 Gujarat. Cropping intensity, or the number of crops (single, double, and triple) each year in a unit farmland area, is a measure of agricultural intensification. For appropriate agricultural management, information regarding the crop calendar (i.e., the number of crops in a parcel of land, their planting and harvesting dates, and the date of peak vegetative stage) is required.

Patel and Oza (2014) presented MODIS data series of one agricultural year over Gujarat state (India). They used NDVI time-series over the crop year 2012/13 for deriving a crop calendar. Stating that such analysis is very useful for analysing dynamics of kharif and rabi crops. Kharif, June to October, and Rabi, November to March are the two different seasonal crops in northern India (Bisht et al., 2014).

The idea was to monitor the land use of agriculture over a year (see Figures 2.7, 2.8, and 2.9) to determine some key elements like the number of crops per year and their planting, peak/saturation, and harvesting dates over the crop growth cycle. The analysis based on values of NDVI at regular time intervals provides useful information about various crop growth stages and the performance of crops in a season, being also possible to extract the number of crop cycles per year and their crop calendar.

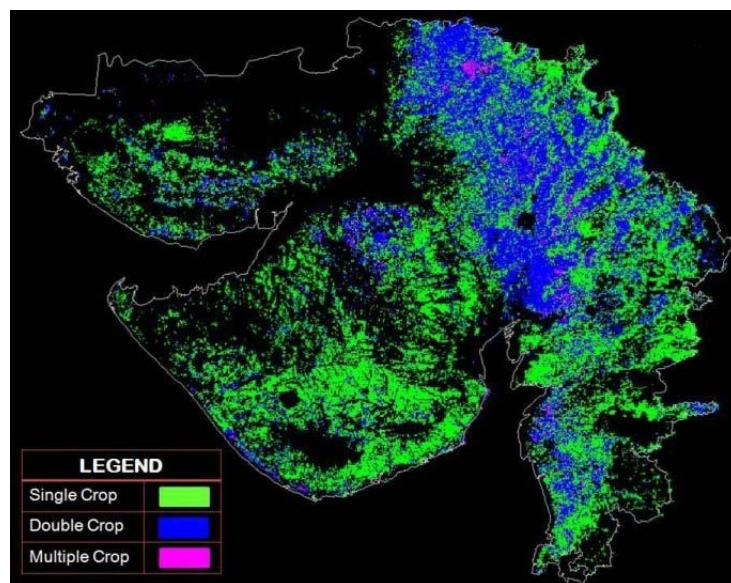


Figure 2.7: Spatial distribution of crops - number of crops (Patel and Oza, 2014).

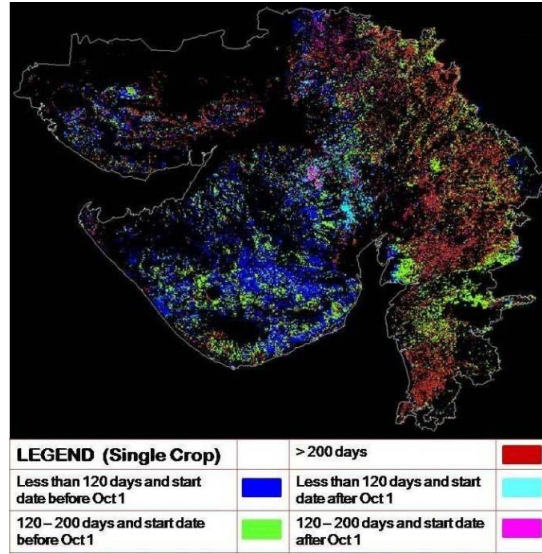


Figure 2.8: Spatial distribution of crops - single crop (Patel and Oza, 2014).

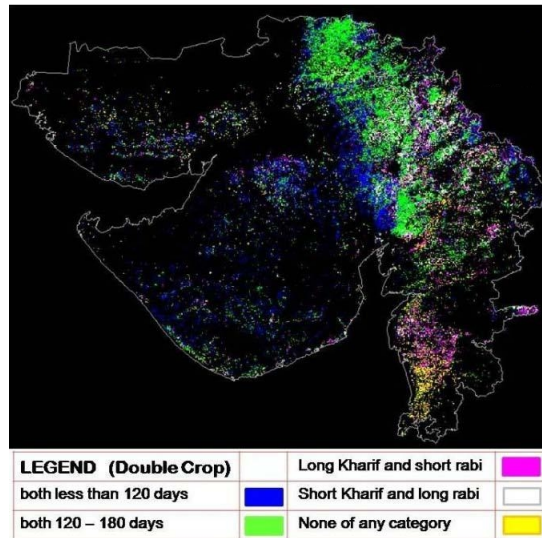


Figure 2.9: Spatial distribution of crops - double crop (Patel and Oza, 2014).

4 China. According to Zhao et al. (2019), Northeast China has a large grain-producing region, an ecologically significant forest region, and the country’s greatest historic industrial base, all of which are experiencing economic downturn. As a result, long-term land cover maps are in great demand and accurate in many regional applications. Using multi-temporal Landsat images, the authors created a collection of continuous yearly land cover mapping products with a 30m resolution.

They also studied long-term land cover dynamics from 1986 to 2016 of Northeast China using multi-temporal Landsat images. The land cover map series of the studied area during the last three decades is found in Figure 2.10. They sampled 2875 locations distributed across Northeast China for training a spectral indices based classification model to detect the change in land cover, and model was able to achieve an accuracy of 88.38% and 80.69% for the year 2015 and 2000 over independent validation dataset, respectively.

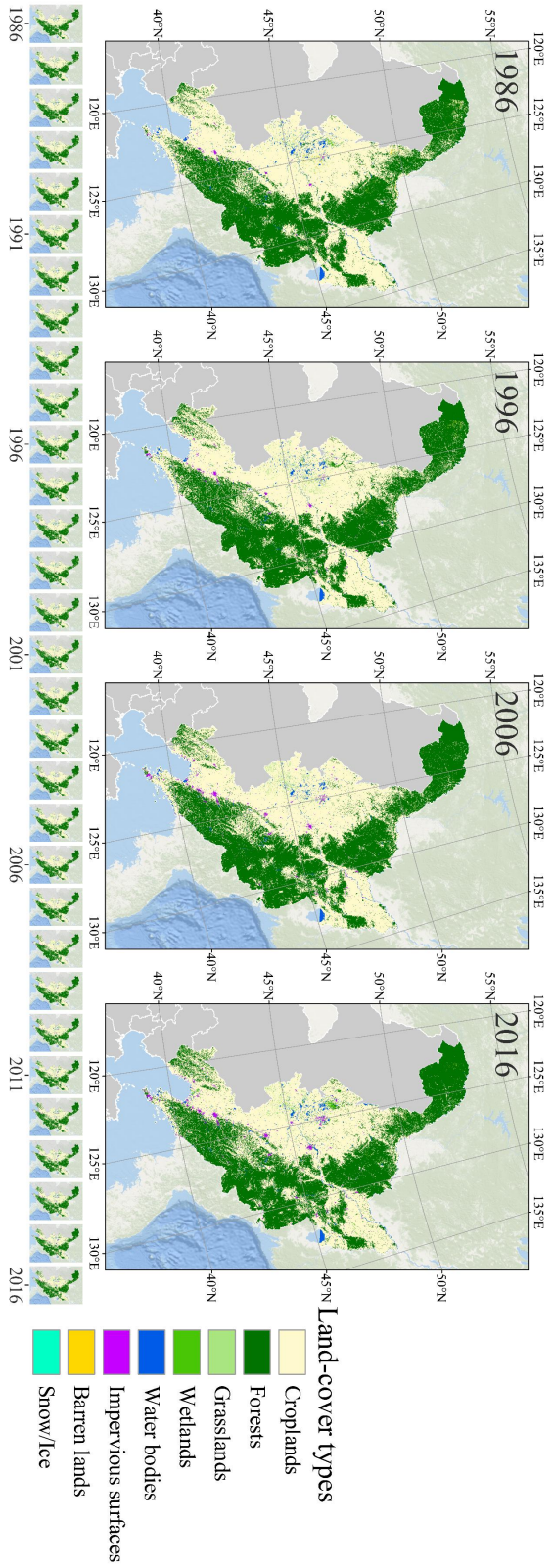


Figure 2.10: The annual land cover map series of northeast China based on multi-temporal Landsat imagery (Zhao et al., 2019).

5 Iran. Increased irrigated land in semi-arid countries of Asia and Africa, driven by the need for additional food production, is placing strain on already stressed available water supplies. To cope with and control this scenario, basin-level monitoring of the spatial and temporal dynamics of irrigated area land use is required to assure optimal water allocation (Pareeth et al., 2019).

Pareeth et al. (2019) created a LULC map at 15m spatial resolution with nine classes for the crop year 2015/2016 using Landsat 8 for the Mashhad basin area, which covers an area of $16,750 \text{ km}^2$ in northeast Iran (refer Figure 2.11). In addition, for three crop years 2013/2014, 2014/2015, and 2015/2016 five irrigated land use categories were extracted. For mapping agricultural land-use patterns over this area, the authors adopted a random forest-based hierarchical technique. They developed their model using the three crop years cycles, obtaining an accuracy of 87.20% and an estimated kappa of 0.85% when paired with a field survey. For the cropping years 2013/2014, 2014/2015, and 2015/2016, the total irrigated area was expected to be 1796.16 km^2 , 1581.7 km^2 , and 1578.26 km^2 , respectively.

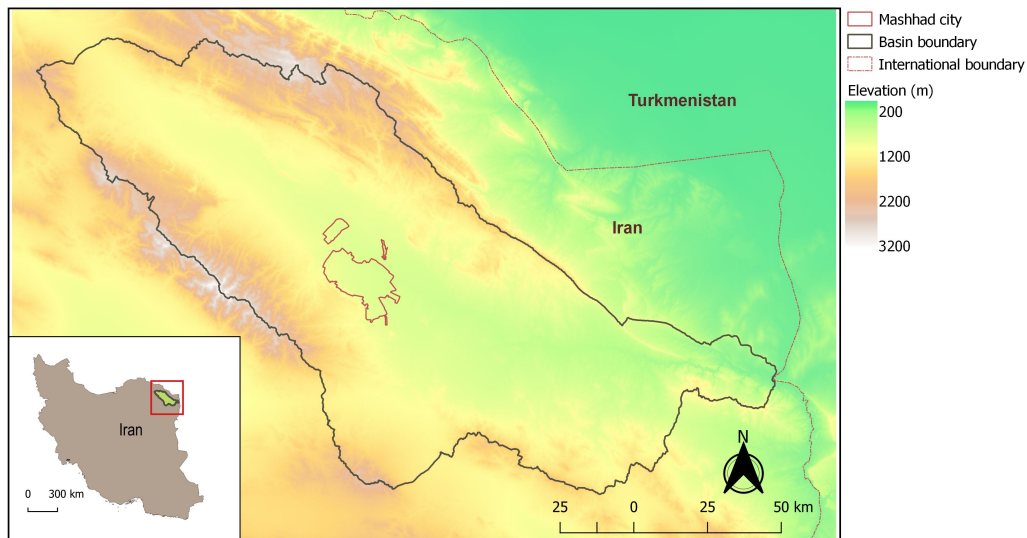


Figure 2.11: Mashhad basin in the northeast of Iran (Pareeth et al., 2019).

2.3 Image Scene Classification

In remote sensing, classifying parts of the high-resolution optical satellite images into morphological categories (e.g., land, water, cloud, etc.) is known as scene classification (Mojajerani et al., 2018). Recently, the challenge of optical satellite image scene classification has been the focal point of many researchers. Scene classification plays a key role in urban and regional planning (Hashem and Balakrishnan, 2015; Rahman et al., 2012), environmental vulnerability and impact assessment (Liou et al., 2017; Nguyen and Liou, 2019) and natural disasters and hazard monitoring (Dao and Liou, 2015), for example. Further, given the current population growth and industrial expansion needs, assessment of land-use dynamics is certainly required for the well-being of individuals.

In prior researches, several methods like look-up tables from big databases, atmospheric corrected images, sensor-specific thresholds rules (Frantz et al., 2018; Main-Knorn et al., 2018; Zhu and Woodcock, 2012) or time-series analysis (Zhu and Woodcock, 2014; Hagolle et al., 2010; Petrucci et al., 2015) were used for automated satellite image classification. Moreover, previous researches focused mainly on classifying individual pixels or objects through image features (Moustakidis et al., 2011; Munoz-Mari et al., 2012) such as color histograms, the gist descriptor (Oliva and Torralba, 2001) and local binary patterns (Ojala et al., 2002) enabling the detection of micro-structures (like points, lines, corners, edges or plain/flat areas). These image features have proved to be effective in image classification to distinguish objects like roads, soil, and water, but can not provide morphological information such as clouds, vegetation, shadows or urban areas (Hu et al., 2015a).

Currently, for image scene classification, the majority of open-access datasets are either limited in data diversity, size or the number of classes. For example, the publicly available dataset UC Merced (Yang and Newsam, 2010) consists of 100 (256 x 256 pixels) images for 21 classes, the Aerial Image Dataset (AID) (Xia et al., 2017) consists of 10,000 images within 30 aerial scene types, the Brazilian Coffee Scene Dataset (Penatti et al., 2015) is composed of 950 (600 x 600 pixels) aerial scene images uniformly distributed over 50 classes, EuroSAT (Helber et al., 2019) comprises of 27,000 (64 x 64 pixels) georeferenced and labeled image patches, PatternNet (Zhou et al., 2018) contains 38 classes with 800 images per class and BigEarthNet (Sumbul et al., 2019) contains of 590,326 Sentinel-2 image patches acquired between June 2017 and May 2018 over the 10 countries.

For image scene classification, there are mainly two types of classification, Object-based classification (OBC) and Pixel-based classification (PBC) (Kim et al., 2011).

Pixel-based classification examines multispectral data to assign a pixel to a class based on spectral similarities between the classes (Sekertekin et al., 2017). Maximum Likelihood Categorization (Dean and Smith, 2003) and Iterative Self-Organizing Data Analysis Technique (Vatsavai et al., 2011) are two of the most widely used approaches for pixel-based classification (Zerrouki and Bouchaffra, 2014; Xiong et al., 2017). However, all of these techniques have one limitation in common: they don't make use of spatial and textural information (Myint et al., 2011).

Unlike single-pixel classification, object-based techniques work with objects made up of multiple homogenous pixels that have been segmented into meaningful groups (Blaschke et al., 2004). Image objects provide shape features essential for categorization in addition to spectral information used in pixel-based classification algorithms (Zhang and Yi, 2012). The problem of under-segmentation and over-segmentation, however, is a possible

drawback of object-based classification (Liu and Xia, 2010).

The remainder of this section will attempt to cover recent work in both object-based and pixel-based classification approaches.

2.3.1 Object-based classification

The generalization capacity of the trained model is often increased if the related scene information is taken into account throughout the scene classification learning process (Zou et al., 2015). The core premise of object-based categorization is to leverage key information (shape, texture, and contextual information) found in image objects and internal connections (Wang et al., 2004). This technique is also in accordance with the viewpoint of geographical or landscape ecology, which says that it is desirable to work on a meaningful item that represents the real spatial pattern rather than a uniform pixels (Blaschke and Strobl, 2001). The object-based procedures are composed of two steps: segmentation and classification (Darwish et al., 2003). The main goal of the segmentation step is to divide the whole image into a series of closed objects that correspond to the spatial pattern. Then, a knowledge base that defines the properties of output object types guides the classification process (Haralick and Shapiro, 1992; Mather and Koch, 2011).

Table 2.5 summarize the most frequent classifiers employed using the object-based technique (Phiri et al., 2020). The table primarily describes the *reference*, *classification application* and, finally, the *classifier* that was applied. Here, RF: Random Forest, SVM: Support Vector Machine, ANN: Artificial Neural Network, KNN: k-Nearest Neighbors.

2.3.2 Pixel-based classification

The pixel has long been the fundamental unit of image analysis, and pixel-based land cover/use classification is one of the most widely used classification methods for remote sensing data (Rujoiu-Mare et al., 2017). In this approach, without taking into account the geographical context of the pixel, pixel's characteristics and spectral features are used to identify and analyze the changes (Collins and Woodcock, 1996; Hussain et al., 2013).

Researchers also took a closer look at pixel-based techniques in classification, highlighting their features, benefits, and drawbacks by applying statistical operators to evaluate each pixel (Coppin et al., 2004; Lu et al., 2004; İlsever and Ünsalan, 2012). Apart from classification, pixel-based techniques can also be used for change detection (Gamba et al., 2006). Change detection is described as “the process of recognizing variations in the state of an item or phenomena by watching it at various intervals,” according to Singh (1989). For example, deforestation, damage assessment, disaster monitoring, urban development, planning, and land management all benefit from land-cover and land-use change data detection.

Table 2.6 summarize the most frequent classifiers employed using the pixel-based technique (Phiri et al., 2020). The table primarily describes the *reference*, *classification application* and, finally, the *classifier* that was applied. Here, RF: Random Forest, CCF: Canonical Correlation Forest, MESMA: Multiple Endmember Spectral Mixture Analysis, SVM: Support Vector Machine, ANN: Artificial Neural Network, MLC: Maximum Likelihood.

Table 2.5: Object-based LULC Classification Survey.

Reference	Classification Application	Classifier
Dong et al. (2020)	Cropland	RF
Csillik and Belgiu (2017)	Wheat, Rice	Ruleset
Delalay et al. (2019)	Settlement Industry	Decision Tree
Derksen et al. (2018)	Crops & Road	Contextual
Glinskis and Gutiérrez-Vélez (2019)	Bare-soil & Forest	ANN
Heryadi and Miranda (2019)	Forest & Water-body	KNN
Laurent et al. (2014)	Brown & Green Leaves	Bayesian
Mongus and Žalik (2018)	Agriculture & Forest	Naive Bayes
Popescu et al. (2016)	Urban & Agriculture	Latent Dirichlet
Zheng et al. (2018)	Roads & Bareland	SVM

Table 2.6: Pixel-based LULC Classification Survey.

Reference	Classification Application	Classifier
Clark (2017)	Bareland and Built-up area	RF
Colkesen and Kavzoglu (2017)	Forest, Soil, and Corn	CCF
Degerickx et al. (2019)	Pavement, Soil, and Tree	MESMA
Denize et al. (2018)	Winter crop and Grassland	SVM
Forkuor et al. (2018)	Agriculture and Urban area	ANN
Gutierrez et al. (2011)	Onion, Sunflower, and Sugar beet	RF
Miranda et al. (2018)	Water, Forest, and Urban Bareland	MLC

Table 2.7 shows a survey of various pixel-based change detection approaches (Hussain et al., 2013). The table primarily describes the *reference*, *changed detecting application* and, finally, the *technique* that was applied. Here, for example, Principal Component Analysis (PCA), Multi Date Direct Comparison (MDDC), Decision Tree (DT), GIS Integration, Fuzzy Change, and Multi-Sensor Data Fusion (MSDF), were used to detect changes in land use and land cover.

Table 2.7: Pixel-based LULC Change Detection Survey.

Reference	Change Detection Application	Approach
Collins and Woodcock (1996)	Forest Ecosystems	Image Differencing
Howarth and Wickware (1981)	Environmental	Image Rationing
Ludeke et al. (1990)	Tropical Deforestation	Regression Analysis
Wilson and Sader (2002)	Forest Harvest Type	Vegetation Index Differencing
Bayarjargal et al. (2006)	Disaster Assessment	Change Vector Analysis
Deng et al. (2008)	Land Use	PCA
Jin and Sader (2005)	Forest Disturbance	Tasselled Cap Transformation
Tomowski et al. (2011)	Urban Disaster	Texture Analysis
Richards and Jia (2006)	Urban Sprawl	Post Classification
Lunetta et al. (2006)	Land Cover	MDDC
Woodcock et al. (2001)	Forest Change	Artificial Neural Network
Huang et al. (2008)	Forest Cover	Support Vector Machine
Im and Jensen (2005)	Land Cover	Decision Tree
Pijanowski et al. (2002)	Land Use	GIS Integration
Vila and Barbosa (2010)	Post Fire Vegetation Regrowth	Spectral Mixture Analysis
Fisher et al. (2006)	Landscape	Fuzzy Change
Deng et al. (2008)	Land Use	MSDF













2.3.3 Rule-based classification algorithm

Rule-based classification algorithm is applicable to both pixel and object based classification (Pradhan et al., 2016). Sen2Cor is a rule-based pixel classification method, this justifies our focus on Rule-based classification in this section.

While capturing satellite images, the atmosphere influences the spatial and spectral distribution of the electromagnetic radiation from the Sun before it reaches Earth's surface. As a result, the reflected energy recorded by a satellite sensor is affected and attenuated, requiring an atmospheric correction.

Sen2Cor is an algorithm whose pivotal purpose is to correct single-date Sentinel-2 Level-1C products from the effects of the atmosphere and deliver a Level-2A surface reflectance product. Level-2A (L2A) output consists of a Scene Classification (SCL) image with eleven classes together with Quality Indicators for cloud and snow probabilities, Aerosol Optical Thickness (AOT) and Water Vapour (WV) maps and the surface (or BOA) reflectance images at different spatial resolutions (60m, 20m, and 10m). Table 2.8 presents the eleven classes with their corresponding color representation in SCL image. Each particular classification process (European Space Agency, 2020) is discussed next and Appendix A.1 presents different reflectance ratio used next.

Table 2.8: List of Sen2Cor Scene Classification Classes and Corresponding Colors (European Space Agency, 2020).

No.	Class	Color Name	Color
0	No Data (Missing data)	black	
1	Saturated or defective pixel	red	
2	Dark features / Shadows	very dark gray	
3	Cloud shadows	dark brown	
4	Vegetation	green	
5	Bare soils / deserts	dark yellow	
6	Water (dark and bright)	blue	
7	Cloud low probability	dark gray	
8	Cloud medium probability	gray	
9	Cloud high probability	white	
10	Thin cirrus	very bright blue	
11	Snow or ice	very bright pink	

Cloud and Snow. Figure 2.12 describes the Sen2Cor Cloud/Snow detection algorithm: it performs six tests and the result of each pixel is a cloud probability ranging from 0 for high confidence clear sky to 1 for high confidence cloudy sky. After each step, the cloud probability of a potentially cloudy pixel is updated by multiplying the current pixel cloud probability by the result of the test. The snow detection follows the same procedure with five different tests resulting in 0 for high confidence clear, no snow to 1 for high confidence snowy pixel.

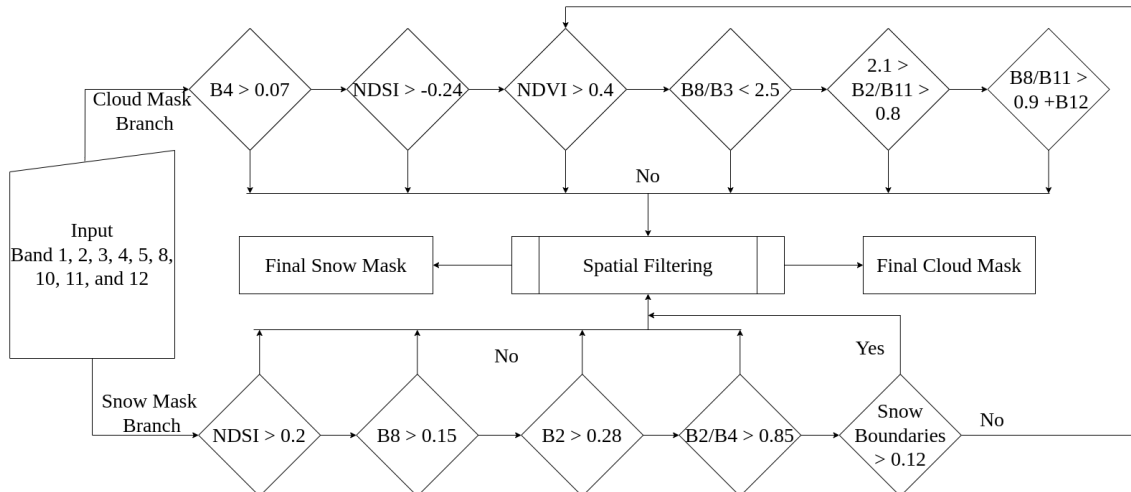


Figure 2.12: Sen2cor Cloud and Snow mask algorithm.

Vegetation. Two filters, namely the *NDVI* (Rouse et al., 1974) and a reflectance ratio (R), are used to identify vegetation pixels. Thresholds of $T1 = 0.40$ and $T2 = 2.50$ are set for *NDVI* and R , respectively. If the *NDVI* and R values exceed the corresponding thresholds, the pixel is classified as vegetation in the classification map.

Soil and Water. Bare soil pixels are detected when their reflectance ratio $R1$ falls below a threshold $T = 0.55$ or exceeds a threshold $T = 4.0$ the pixel is classified as bright water.

Cirrus Cloud. Under daytime viewing conditions, the presence of thin cirrus cloud in the upper troposphere is detected by Sentinel-2 band 10 (B10) reflectance threshold. In the first step, all B10 pixels with a value between $T = 0.012$ and $T = 0.035$ are considered as thin cirrus; in the second step, after generating a probabilistic cloud mask, if the cloud probability is below or equal to 0.35, the pixel is classified also as a thin cirrus cloud.

Cloud Shadow. The cloud shadow mask is constructed using a “geometrically probable” cloud shadow, derived from the final cloud mask using sun position and cloud height distribution and a “radiometrically probable” cloud shadow, derived from a neural network (Kohonen, 1982).

2.4 Summary

Over the last five decades, satellite remote sensing has evolved into one of the most powerful instruments for scanning the Earth on local, regional, and global sizes. Because this space-based study is non-destructive, it enables quick monitoring of the ambient atmosphere, its underlying surface, and the mixed layer of the ocean. This chapter goes into detail on Earth observation, the Sentinel mission, especially Sentinel-2, land usage and land cover (LULC), and how LULC may be monitored. Following that, it is detailed what image scene classification is, as well as the various methodologies accessible and how Sen2Cor

performs image scene classification. Furthermore, a comprehensive examination of existing approaches for LULC and image scene classification is offered, together with an assessment of their operational strengths and weaknesses.

Chapter 3

Active Learning

“Having N labeled training points from a set of classes (C) described by a set of attributes (A) and T testing points, is it possible to use fewer labeled samples ($S \ll N$) during the training phase and achieve the same accuracy over the test set?”

— Prof. Teresa Gonçalves

Active learning is a supervised learning method in which the learner selects labeled instances using a set of rules aimed at reducing labeling complexity (Angluin, 1988; Muslea et al., 2006; Leng et al., 2013; Karlos et al., 2021). Here, according to the definition given by Settles (2009), the number of label requests required and adequate to understand the target notion is referred to as labeling complexity. Note that, this complexity is not related to the concepts of Minimal Description length and Kolmogorov complexity (Vitányi and Li, 2000), where more labeling requests give the same complexity. An active learner can use an oracle to pose questions about unlabeled data. In many current machine learning contexts, active learning is well-motivated when unlabeled data is available or inexpensive to gather, but labeling is difficult, time-consuming, or expensive to obtain.

Several classification applications employing unstructured data, such as speech recognition, text and web page categorization, image and audio retrieval and filtering, require excellent classification methods due to the high cost of labeling and the vast volume of available but unlabeled data (Settles, 2009). In these situations, efficiency refers to a balance of high accuracy and comprehensiveness with low labeling work. For such cases, active learning had been an effective learning environment in which learning approaches aiming at a desirable trade-off can be developed. For example:

- Accurate speech sound categorization takes a long time and needs the skills of linguists with extensive experience. According to Zhu (2005), annotating words takes ten times the time the audio (e.g., one minute of speech takes ten minutes to label), but annotating phonemes takes 400 times the time (e.g., almost seven hours).
- Learning to categorize records (such as articles or web pages) or any other sort of media (such as picture, music, and video files) necessitates assigning precise labels to each document or media file, such as ‘relevant’ or ‘not relevant’. Annotating thousands of these instances can be time-consuming and sometimes repetitive.

This chapter provides an overview of active learning, covering several scenarios, a query strategy structure, and a comprehensive literature analysis. A review of the empirical and theoretical evidence for efficient active learning is presented, as well as a description of problem setting modifications and key challenges, including a review of application of active learning in domain-specific problems. To be specific, Section 3.1 delves into different active learning sampling method, such as membership, stream, and pool; Section 3.2 delves into different query selection methods, such as committee, expected error reduction, and uncertainty; Section 3.3 delves into different baseline parameters for active learning experiments; and finally, Section 3.4 delves into the literature on active learning and domain application.

3.1 Sampling Method

The largest proportion of active learning research is devoted to converting the human notion of questioning into programmable approaches (Tong and Koller, 2001). When these programmable techniques are adjusted to the peculiarities of the dataset in consideration, the resulting strategy can produce excellent results in practice. However, there are a variety of theories that underpin various ‘human-asked’ queries, and no single concept is likely to meet the criteria for all data collections (Donmez and Carbonell, 2008a). Properly selecting methods for each given data collection is thus a crucial practical challenge (Huang et al., 2010). The following are the three main sampling methods to active learning algorithms. The active learning algorithm is also referred as learner below.

- Membership-based sampling: The active learner requests the expert to classify the situations the learning system creates. For each instance, the learner assigns values to the characteristics and observes the response. This allows the learner the freedom to frame the data instance that will be most useful to him or her at the time (Angluin, 1988).
- Stream-based sampling: The active learner is supplied with a stream of unlabeled cases from which it chooses one for the expert to label. This may be thought of as active learning in an online pool (Freund and Schapire, 1997a).
- Pool-based sampling: the learner is given a pool of unlabeled cases that are dispersed independently and identically. At each step, the active learner selects an unlabeled instance to request a label from the expert using a querying function (Lewis and Gale, 1994a).

3.1.1 Membership-based sampling

Learning through membership queries also known as query synthesis was among the first active learning contexts to be explored. Six separate kind of synthesized queries were defined in this context (Angluin, 1988): membership, equivalence, subset, superset, disjointness, and exhaustiveness queries. Except for the membership category, which produces a simple binary (1/0), the answer to each of these queries consists beside the binary response, a counterargument in the event of a negative response. Figure 3.1 illustrates the process.

The query synthesis idea has been applied to regression learning problems, such as learning

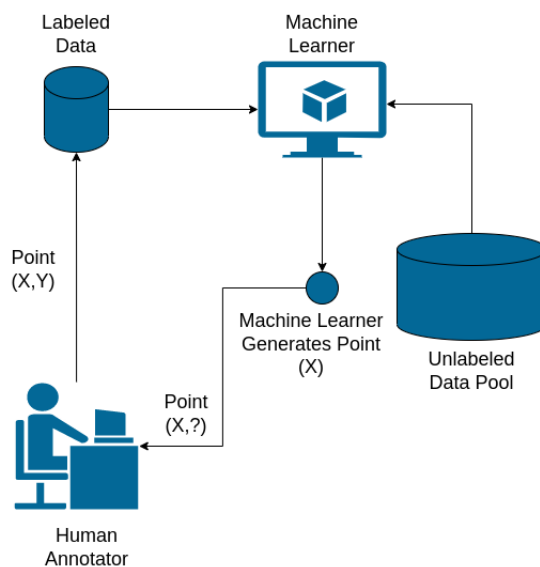


Figure 3.1: Membership-based sampling active learning life cycle (Settles, 2009).

to predict the exact position of a robot hand supplied with the angular position of its robot arm as parameters (Cohn et al., 1996a).

While query synthesis is viable for many applications, identifying such arbitrary queries may be challenging in the case of a human annotator. For example, in Freund et al. (1997)’s work, membership query learning was used with human annotators to train a neural network to classify handwritten characters. They ran into an impasse: some of the learner’s inquiry pictures were devoid of recognized letters, comprising unnatural composite symbols with no natural semantic information. Likewise, membership queries for natural language processing tasks may yield jibber-jabber text or audio streams. To address these constraints, the stream-based and pool-based models (described further in this section) were constructed.

The membership inquiry situation is described in a new and promising real-world application by King et al. (2004a, 2009). They have a ‘cyborg biologist’ who can conduct a series of unsupervised biomedical investigations on the yeast variant (*Saccharomyces cerevisiae*) to discover metabolic pathways. Examples include a combination of chemical solutions used to create a starter culture and a specific yeast variant. Within the starter culture, if the yeast variant advances, it is labeled. All experiments are conducted autonomously by a research lab bot using an active learning method based on inductive logic programming. When compared to naively executing the least costly experiment, this active strategy saves three times the amount of money on supplies and a hundred times the amount of money when compared to randomly created tests. This approach may be a potential method for automated scientific discovery in disciplines where labels are produced by experimentation instead of human experts.

3.1.2 Stream-based sampling

Stream-based sampling can be used instead of membership-based sampling (Helmbold and Panizza, 1997; Atlas et al., 1989). The key presumption is that obtaining an unlabeled

sample is inexpensive or minimal; consequently, it can be selected from the real distribution first, and the learner can then choose rather or not to demand its annotation (Zhu et al., 2007, 2010). Figure 3.2 illustrates the process.

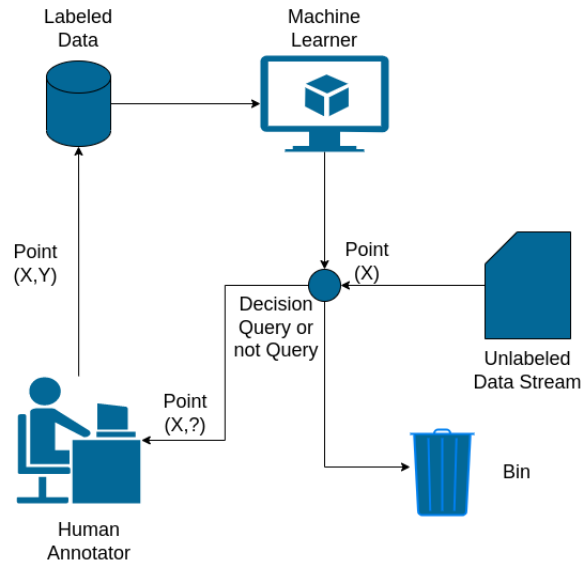


Figure 3.2: Stream-based sampling active learning life cycle (Settles, 2009).

In stream-based or sequential active learning, the learner must determine whether to query or ignore a single unlabeled sample, retrieved from the source of data one at a time (Dasgupta et al., 2009). If the data distribution is homogeneous, stream-based sampling may behave similarly to membership query learning (Chu et al., 2011). Even if the distribution is non-uniform and even more importantly undefined, queries will always be acceptable since they are based on a real underlying distribution.

Label efficient learning refers to the trade-off between the cost of asking a query and the cost of mistakes in a stream-based context (Cesa-Bianchi et al., 2006). Cesa-Bianchi et al. (2005) introduced a minimum-variance approach to guide instance labeling from data streams using an ensemble of classifiers. When the number of the acceptable query is fixed, they came up with matching upper and lower bounds for the greatest possible confidence interval for the best possible ‘label efficient learning’.

There are several approaches to structure a query instance (Dagan and Engelson, 1995): ‘informativeness measure’ evaluates samples’ informativeness and creates a biased random selection with more informative instances being chosen more frequently; another approach is to compute an explicit region of uncertainty, the section of the data space that is still unclear to the classifier and query this examples that fall inside it. Straightforwardly, region selection is accomplished by establishing a minimal boundary on an informativeness measure that defines the region; the instances exceeding this boundary are then inquired (Mitchell, 1982).

A more sensible technique would be to divide a section of the complete model class in terms of version space. The set of hypotheses consistent with the current labeled training set is referred as the version space. In other words, an instance in the uncertainty region is some unlabeled data for which two models from the same model class but with different parameter values disagree. However, calculating this region fully and explicitly is compu-

tationally expensive, and it must be maintained after each successive query (Seung et al., 1992; Dasgupta and Hsu, 2008; Yang et al., 2015; Mayer and Timofte, 2020).

Part-of-speech tagging (Tur et al., 2005), sensor scheduling (Krishnamurthy, 2002), and learning ranking functions for information retrieval (Yu et al., 2021) have all been investigated using the stream-based scenario. Fujii et al. (1999) employed selective sampling for active learning in word meaning disambiguation, such as determining if the word ‘bank’ in a given context, relates to riverside land or a financial institution. Not only does the strategy save time on annotation, but it also reduces the size of the database used in nearest-neighbor training, allowing the classification process to run faster.

3.1.3 Pool-based sampling

The vast amounts of unlabeled data collected for many real-world learning problems inspire pool-based sampling, which assumes a small pool of labeled data and a large pool of unlabeled data (Lewis and Gale, 1994a). In pool-based sampling, queries are often picked from an unlabeled pool that is deemed closed i.e., fixed or dynamic; however, this is not always the case. In most cases, instances are greedily queried using an informativeness metric applied to all unlabeled instances in the pool. Figure 3.3 illustrates the process.

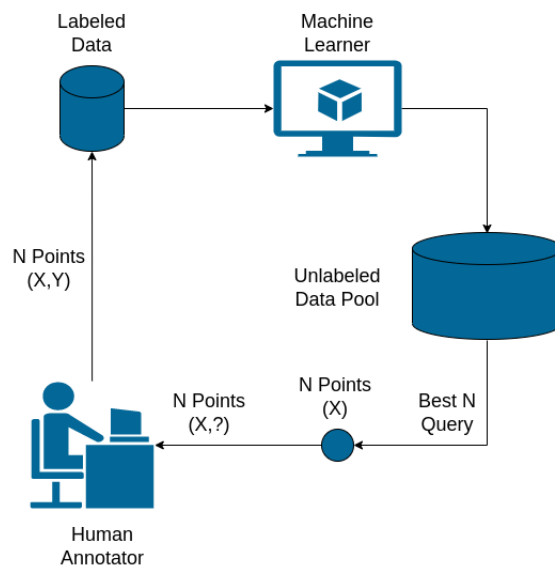


Figure 3.3: Pool-based sampling active learning life cycle (Settles, 2009).

The majority of existing research on pool-based active learning focuses on creating plausible criteria for labeling instances. Uncertainty sampling is a prominent criterion that asks the classifier about the instance that is the most unsure (Lewis and Gale, 1994a). For instance, querying the point closest to the SVM-trained decision border (Vapnik, 1998; Tong and Koller, 2001) or a point near a boundary is more representative if it is located in a denser neighbourhood, and they offer density-weighted criteria (Nguyen and Smeulders, 2004; Donmez et al., 2007). Establishing a distance function through clustering is another method for determining representativeness (Donmez and Carbonell, 2008b; Dasgupta and Hsu, 2008). Another technique determines representativeness by estimating probable label assignments for unlabeled occurrences (Huang et al., 2010).

The main difference between stream-based and pool-based active learning is that stream-based scans the data sequentially and makes query judgments individually, whilst the pool-based reviews and ranks the whole collection before selecting the best query. While the later situation appears to be considerably more common in application papers, there are times when the stream-based approach is better.

Regardless of the sample scenario, however, there are rarely substantial links between the human-designed criterion and the performance measure of interest. Furthermore, what a person considers to be excellent questions may not be appropriate for every data collection or context. This shortcoming suggests the necessity to select among a variety of methods in a data-dependent and adaptable manner (Baram et al., 2004).

Many real-world problem have been explored using the pool-based scenario, including text classification (Lewis and Gale, 1994a; McCallumzy and Nigamy, 1998; Tong and Koller, 2001; Hoi et al., 2006a), information extraction (Thompson et al., 1999; Settles and Craven, 2008), image classification and retrieval (Tong and Koller, 2001; Zhang and Chen, 2002), video classification and retrieval (Yang et al., 2003; Hauptmann et al., 2006), voice recognition (Tur et al., 2005), and cancer detection (Liu, 2004), to mention a few.

3.2 Query Selection Methods

In every iteration of all active learning scenarios, an unlabeled sample from a specified distribution is chosen and evaluated for its informativeness. Several different methods of evaluating approaches have been offered in the literature, and this section gives an overview of some frameworks that are in use.

3.2.1 Committee

The query-by-committee (QBC) algorithm is a conceptually driven query selection method (Seung et al., 1992). It entails keeping a committee of models, each of which is trained on the current labeled set but representing opposing hypotheses. The inquiry candidates are then voted on by each committee member. The most instructive question is usually the one in which they disagree the most.

The process of lowering the number of queries necessary to understand a topic is known as labeling complexity (generalization error), and the majority of previous study on the field has focused on this (Hanneke, 2007).

To reduce the generalization error, the first step is accomplished by picking a committee of two random hypotheses that are compatible with the labeled set, as done by Seung et al. (1992). This may be done more broadly for generative model classes by randomly picking an arbitrary set of models from a likelihood function. McCallumzy and Nigamy (1998) and Dagan and Engelson (1995) used the Dirichlet distribution (Wong, 1998) over training sets to sample regression techniques whereas Normal distribution (Patel and Read, 1996) to sample hidden Markov models.

Abe et al. (1998) presented query-by-boosting and query-by-bagging for various model classes, such as discriminative or non-probabilistic models, using the well-known ensemble learning methods of boosting (Freund and Schapire, 1997b) and bagging (Breiman, 1996)

to create committees. Another ensemble-based strategy proposed by [Melville and Mooney \(2004\)](#) expressly encourages diversity among committee members.

Till now, the ideal committee size for the division has not been considered in any of the reviews described; it relies upon the model class or application. [Muslea et al. \(2000\)](#) created a committee of two models by separating the feature space. Even small committee sizes (e.g., two or three) have been proved to be effective ([Seung et al., 1992](#); [McCallum and Nigamy, 1998](#); [Settles and Craven, 2008](#)).

Another key challenge is comprehending the fundamental concepts that control query selection criteria. [Shannon \(1948a\)](#) suggest that the information value of a query may be approximated from the committee’s disagreement, and an increase in disagreement can result in a substantial information gain. [Shannon \(1948a\)](#) demonstrate the benefits of QBC over random sampling in two toy classification challenges. When using QBC, the information gain of a query approaches a finite value when the number of inquiries hits infinity. When utilizing random sampling, information gain approaches zero. In both cases, the asymptotically restricted information gain is responsible for the exponentially decreasing generalization error as an inverse power law. But random sampling performs poorer than QBC in terms of information gain.

Based upon uncertainty, in each iteration of active learning process, [MacKay \(1992\)](#) dynamically generated informative sub-areas inside the uncertainty region for query selection, and this region represents the most informative sub-areas within data space. When the number of the committee exceeds two, further study on QBC reveals a significant rise in the selection of sub-areas for the selected queries ([Freund et al., 1992](#)).

QBC was applied in a text classification challenge by [Liere and Tadepalli \(1997\)](#), using Winnow ([Littlestone, 1988](#)) as a base classifier. Winnow classifiers are well-suited to high-dimensional feature spaces containing a large number of irrelevant characteristics, such as those found in text corpora. When comparing QBC to a single Winnow classifier, the empirical findings demonstrate one to two orders of magnitude decrease in labeling effort. In regression scenarios, QBC may be used to measure disagreement about the variance among the committee members’ output predictions ([Burbidge et al., 2007](#)).

3.2.2 Expected Error Reduction

One possible query selection approach is to minimize the model’s generalization error. It means that, selection is not focused upon how much model changes, but the goal is to predict a model’s projected future error on the remaining unlabeled examples and minimize the expected 0/1 loss. Unfortunately, in the majority of circumstances, predicted error reduction is the most computationally costly query framework. Not only should the predicted error across unlabeled examples be estimated for each query, but a new model must be progressively re-trained for each conceivable query labeling, which iterates over the whole pool. This leads to a drastic increase in computational cost for many model classes such as binary logistic regression. The incremental training technique is efficient and precise for multivariate model classes such as Gaussian random fields ([Zhu et al., 2003](#)), making this approach fairly practical.

By forecasting the future error rate with a loss function, [Roy and McCallum \(2001b\)](#) offer a way to directly optimize the projected error rate decrease. The loss functions assist the

learner in selecting those occurrences that increase the learner’s confidence in the unlabeled data. This approach calculates the expected error across a sample in the pool rather than the entire distribution. The authors use naive Bayes for their classification and class probability estimations, however SVMs or other models with complicated parameter space can also be used (Zhu et al., 2003; Moskovitch et al., 2007). Baram et al. (2004) implements this approach using SVMs and finds that it outperforms the original naive Bayes algorithm; in the SVM-based technique, the class probabilities are estimated using logistic regression. Mitra et al. (2004a) demonstrated how to enable vector machine learning using a probabilistic active learning technique; identifying all genuine support vectors ensures that future errors are minimal; the approach assigns a confidence factor to all occurrences inside the border, close to the real support vectors, as well as interior locations far from the support vectors, using the k closest neighbour technique. The cases are then selected probabilistically using the confidence factor as a criterion.

Guo and Greiner (2007) proposed an ‘optimistic’ form that uses uncertainty sampling as a contingency method when the human annotator offers incorrect labeling and biases the expectation toward the most likely label for computational ease. This approach has the benefit of being both near-optimal and independent of the model class. All that’s needed is a suitable objective function and a method for calculating bayesian label probabilities. For example, naive Bayes, Gaussian random fields, logistic regression, and support vector machines have all been effectively used applying this methodology. In theory, the general technique may be used to optimize any generic performance measure of interest, such as accuracy, recall, F1-measure, or area under the ROC curve, in addition to minimizing loss functions.

3.2.3 Uncertainty

Uncertainty sampling is perhaps the most basic and often used query framework (Lewis and Gale, 1994a). In this paradigm, an active learner inquires about the situations for which there is the least amount of classification certainty. For probabilistic learning models, this strategy is frequently used. When employing a probabilistic model for binary classification, for example, uncertainty sampling simply asks the instance whose posterior probability of being affirmative is closest to 0.5 (Lewis and Catlett, 1994). Seung et al. (1992)’s Query-by-Committee approach selects samples for labeling by the expert for which the selected classifiers disagree. According to the authors, their approach may be used with any classifier that predicts a class and gives a probability estimate of the prediction confidence. This method has been used in information extraction tasks, for example, with statistical sequence models (Culotta and McCallum, 2005).

Tong and Koller (2001); Campbell et al. (2000); Schohn and Cohn (2000) used SVMs as the induction component where the querying function is based on the classifier. To be specific, the technique seeks to choose instances that are most informative for dividing the hyperplane. This is similar to uncertainty sampling, in which the algorithm picks the cases about which it has the most doubts. When using SVMs, the classifier is most unsure about the instances that are near to the dividing hyperplane’s margin. Tong and Koller (2001) has suggested variations including the MaxMin Margin and Ratio Margin approaches, which similarly employ SVM as the learner.

The use of entropy as an uncertainty measure is a more broad and arguably the most common uncertainty sampling technique. The amount of information required to ‘encode’

a distribution is measured in entropy, an information-theoretic metric; as a result, it's frequently used in machine learning as a measure of uncertainty or impurity. Additionally, the entropy-based method, easily generalizes to probabilistic multi-label classifiers and probabilistic models for more complicated structured examples like sequences and trees (Settles and Craven, 2008; Hwa, 2004).

Non-probabilistic classifiers can also use uncertainty sampling methodologies. A decision tree classifier was employed in one of the first studies on uncertainty sampling (Lewis and Gale, 1994a). Similar methods have been used in active learning using nearest-neighbour classifiers, in which each neighbor is allowed to vote on the class label, with the proportion of votes indicating the posterior label likelihood (Lindenbaum et al., 2004).

Uncertainty sampling may also be used in regression situations i.e., learning tasks where the output variable is a continuous value rather than a set of discrete class labels. In this case, the learner simply looks for the unlabeled instance with the biggest output variance in the model's prediction. The entropy of a random variable is a monotonic function of its variance under the Gaussian assumption, hence this technique is quite similar to entropy-based uncertainty sampling for classification (Fedorov, 2013; Cressie, 2015).

3.3 Baseline Specification

Active learning aims to reduce the cost of building a predictive solution by enabling the learner to determine which examples should be labeled for training. Most active learning recent studies, however, have presumed that the cost of obtaining labels is the same in all cases. A decline in the amount of labeled data does not always imply a reduction in cost in domains where labeling costs vary (Settles et al., 2008).

We presented a detailed empirical research of baseline specification that are required while using active learning irrespective of the application such as, initial training sample; batch mode and size; training label and computing costs; selection diversity.

3.3.1 Initial Training Sample Selection

Active learning performance can be increased by carefully selecting the initial training samples. Using a fuzzy-c clustering approach (Bezdek, 2013), proposed Yuan et al. (2011) three initial training data selection processes: center-based, border-based and hybrid. Center-based selection chooses samples with a high degree of membership in each cluster; border-based selection selects samples from the clusters' edges; hybrid selection combines center-based and border-based selection. According to their findings, the hybrid selection is able to significantly improve the effectiveness of active learning when compared to randomly selected initial training samples.

The number of queries creating irrelevant labels might be reduced if the labeled set is properly initialized, improving the performance of the classifiers learnt at the start of the learning process. Selecting a suitably labeled set seeks to early understand the distribution of the data to categorize, allowing valuable queries to be selected in subsequent cycles (Motta et al., 2009). However, this is a work that must be completed in the absence of any previous evidence on the notion to be learned. In the literature, there are several

approaches relying on randomness alone or leveraging the existing dataset.

The use of a collection of examples already labeled by some standard or stochastic sample is a frequent technique to initialize the initial training set (Sun and Hardoon, 2010). The most typical strategy is to start the labeled set by randomly picking training cases from each class (Warmuth et al., 2003; Xu et al., 2003; Schütze et al., 2006). The initial locations of the dividing hyperplanes are computed in a linear discriminant manner, which does not require any real data, since they roughly bisect the space of all potential data points. Individual committee members are randomly started with different hyperplanes so that they reflect various initial hypotheses, which is how Liere and Tadepalli (1997) employ QBC.

On the other hand, Random sampling, can be time-consuming, especially when dealing with a substantially skewed dataset. This is one of the key concerns of initialization techniques based on the input space’s density that discard redundant instances in densely populated portions of the feature space while maintaining instances from sparse regions available for querying. Dima and Hebert (2005) assumed that instances from densely populated regions are, with a high likelihood, indicative of the same class and that repeated searches on these regions would overlook under-represented classes while increasing the number of unnecessary queries.

To determine the sample selection, Cebron and Berthold (2007) examined the prospective measure of the density of the input space in a predetermined neighbourhood around the sample. Any instance in dense input space has a significant impact on the potential of being an instance in query. Seed inquiries have the highest potential score, such as those located in heavily inhabited areas. Rare or outlier samples, on the other hand, are not considered as part of the selection process since they are uncommon and hence less beneficial to the classifier.

Clustering is another frequent method for the initialization step that aims to discover the working set’s internal structure. Using K-means clustering, Kang et al. (2004) suggest a technique that splits unlabeled examples into groups and then picks medoids from each cluster supposedly to be the most representative instances from each cluster. When working with high-dimensional input spaces, such as text corpora, the centroid itself may be difficult to classify because it is most certainly a synthetic instance. However, because the cluster synthetic centroids will be assigned the same label as the representative instance, they may be utilized as training instances without incurring any additional labeling costs. The centroids in this situation are referred to as model examples (Kaufman and Rousseeuw, 2009). Experiments on diverse text datasets have demonstrated that if the initial training set is picked using this strategy, the active learner achieves greater accuracy sooner than when random sampling is used (Nguyen and Smeulders, 2004; Li and Anand, 2007).

3.3.2 Myopic vs. Batch Mode

In most active learning studies, inquiries are chosen in sequential order, one at a time. However, in other cases, such as with large ensemble approaches and numerous structured prediction tasks, the time necessary to induce a model is sluggish or expensive. Consider the possibility of a distributed, parallel labeling environment, such as numerous annotators operating on various labeling workstations over a network at the same time. Selecting queries in serial may be inefficient in these circumstances. Batch-mode active learning, on

the other hand, allows the learner to query examples in groups, making it more suitable for parallel labeling settings or models with sluggish training methods.

In Myopic Active Learning (MAL) a single instance is queried at a time, whereas in Batch Mode Active Learning (BMAL) a batch of samples is picked and labeled concurrently. Single instance selection techniques require retraining the classifier for each classified instance, whereas BMAL offers the benefit of not requiring the model to be retrained numerous times throughout the selection phase (Gui et al., 2021). On the other hand, BMAL faces various barriers (Yang et al., 2021): choosing E samples from a pool of U instances might cause computing issues since the number of possible batches C_E^U can be rather high, depending on the values of U and E ; additionally, designing an appropriate method to measure the overall information carried by a batch of samples can be quite challenging; finally for each iteration, one needs to ensure low information redundancy within a batch of chosen instances. Gu et al. (2014) discussed sample selection with the highest density and least redundancy. For example, dense regions are supposed to be representative and informative, whereas the chosen instances from dense region could be not beneficial because of the redundancy among them (i.e. instances may include similar information). Some recent research uses a clustering step after selecting the top E to diversify and select only \hat{E} samples (Citovsky et al., 2021).

The most difficult aspect of batch-mode active learning is assembling the optimum query collection. Because it ignores the duplication of information content across the ‘best’ samples, myopically querying fails in choosing the ‘best’ query collection based on some instance-level query technique. To tackle this, a few batch-mode active learning approaches have been proposed. Xu et al. (2007) and Brinker (2003) examined a method for SVMs that explicitly accounts for variance among batch instances; they are specifically interested in the centroids of clusters of instances nearest to the decision boundary. Although Hoi et al. (2006a) used the features of sub-modular functions to create batches that are guaranteed to be near-optimal, most of these techniques apply greedy heuristics to ensure that cases in the batch are both varied and informative. Guo and Schuurmans (2007), on the other hand, used batch formation for logistic regression as a discriminative optimization issue, attempting to directly design the most informative batch. These methods, for the most part, outperform random batch sampling, which is typically superior to basic ‘best’ batch construction (Hoi et al., 2006b).

3.3.3 Batch Size

Active learning seeks to find the most effective technique to query unlabeled data and construct a learner with the least amount of human supervision. There are two kinds of learning methods, Sequential and Multiple Instance (also known as batch) (Settles et al., 2007).

Sequential active learning methods, have several drawbacks when used in association with complex and costly models like neural networks. Training deep networks takes a long time and updating the model after each label is costly in terms of both manual labeling time and computing resources. Furthermore, due to the local optimization approaches used to train neural networks, a single point is unlikely to have a substantial influence on performance.

Batch active learning is typically advantageous in practical applications because the cost of collecting a chunk of labels for training is often substantially lower than the cost of

gathering the same number of consecutive single label queries (Chakraborty et al., 2014). When the time taken to update the model and choose the next example is excessively long, this is true. However, there is an inherent trade-off between efficiency and performance when labeling budget limits exist, since greater batches result in fewer model updates and higher prediction errors (Smith et al., 2017).

According to Citovsky et al. (2021), when datasets expand in size to contain hundreds of thousands or even millions of labeled samples, the active learning batch size should grow as well. The problem with very high batch sizes is twofold: first, the risks of lower adaptivity increase, and second, the batch sampling technique must scale effectively with the batch size and not constitute a computing bottleneck itself (Deng et al., 2009; Kuznetsova et al., 2020; Van Horn et al., 2018). The batch size, according to Chakraborty et al. (2014), should be determined by the quality and complexity of the samples in the unlabeled stream, as well as the present classifier’s level of confidence on the unlabeled data instances.

Under the heading of information extraction and natural language understanding, Bach and Badaskar (2007) attempted to combine the batch size and instance selection problems into a single optimization function that maximizes diversity, uncertainty, and redundancy while also including a batch size-dependent penalty term.

According to the Shao et al. (2019), when more samples are chosen at the beginning of the training process, fewer samples may be used in later phases to exploit data recommendations; if more samples were allocated to later iterations, the model would have higher variation in the early iterations but a better chance of biasing samples for active learning in the later rounds. Lourentzou et al. (2018), on the other hand, states that the optimal batch size is determined by the dataset and machine learning application to be addressed.

3.3.4 Batch Diversity

A batch mode learning approach is suitable when training expenses are high. Another rationale for adding several inquiries to a training set at each iteration is to avoid aggravating the annotator by repeatedly making the same inquiry, which might result in a little increase in the retrained learner model that is not obvious to the user (Chen et al., 2010).

In batch mode, retraining happens after a batch of instances has been queried, rather than after each query. This method introduces a new barrier in the form of variation within the batch of instances to query. This needed diversity cannot be obtained merely by selecting the most instructional instances seen by the current learner (Schohn and Cohn, 2000; Warmuth et al., 2001).

Brinker (2003) developed a new strategy for choosing batch mode members that included a diversity metric. This method chooses the queries having the biggest relation to already picked samples in terms of coordinate angles and are located around the decision boundary. To eliminate duplication among chosen examples, Hoi et al. (2006a) proposes a batch mode strategy based on the Fisher information matrix (Papathanasiou, 1993). Li and Sethi (2006) used conditional error to calculate diversity within chosen cases. In order to increase SVM performance, Hoi et al. (2009) proposed a novel semi-supervised SVM batch mode with two goals: increasing the quantity of training instances and assuring their diversity. In semi-supervised mode, SVM trains a kernel function using labeled and unlabeled data; the most useful and diverse instances to query are then identified using this strategy.

If active learning algorithms take into consideration the different characteristics of examples in the dataset, they can provide additional benefits. When trained on a dataset with various types of samples that reflect the complete distribution, the resulting classifier will perform effectively. [Baram et al. \(2004\)](#)'s Kernel Furthest-First method chooses the cases that are the farthest away from a collection of labeled samples. Intuitively, it selects the instance from the unlabeled set, which is the most different from the labeled instances currently being used to train the classifier. [Mitra et al. \(2004a\)](#)'s probabilistic approach uses a confidence factor to choose samples that are distant from the current boundary. This type of information assists the active learner in selecting cases from the dataset that are varied in nature. [Nguyen and Smeulders \(2004\)](#)'s active-learning system picks a diverse set of samples because it prioritizes cluster representatives, and each cluster represents a distinct set of data examples.

3.3.5 Labeling Cost

Although the goal of active learning is to minimize the overall cost of training an accurate model, reducing the number of labeled instances does not always imply a reduction in total labeling cost. In tasks like parsing or information extraction, for example, [Baldrige and Osborne \(2004\)](#) and [Culotta and McCallum \(2005\)](#) suggested strategy for decreasing annotation effort in active learning is to use the already trained model to aid in the labeling of query instances by pre-labeling them. However, such solutions do not represent or justify the expense of labeling. Instead, they try to save money by reducing the number of annotation operations that are necessary for a query that has already been chosen.

[Kapoor et al. \(2007\)](#) suggested a rule-based paradigm that considers both the costs of labeling and the costs of misclassification. If the instance is in the training set, each possible query is assessed by adding the labeling cost to the estimated future misclassification costs. [King et al. \(2004b\)](#) abbreviated real labeling costs with a similar rule-based approach; they propose a 'bot researcher' who can perform a series of unassisted biological experiments to reveal metabolic pathways while conserving resources.

The cost of annotating an instance is still believed to be constant and known to the learner in all of the aforementioned situations and indeed almost everywhere in the cost-sensitive active learning literature ([Margineantu, 2005](#); [Tomanek et al., 2007](#)). [Culotta and McCallum \(2005\)](#) pioneered the use of variable labeling costs in data extraction; they suggest a novel paradigm that decreases both the number of examples to label and the difficulty of categorizing each one. The suggested technique distinguishes between boundary and classification annotations and quantifies the number of activities a user must make to label a training example. Annotating boundaries is frequently more difficult than annotating classifications ([Vijayanarasimhan and Grauman, 2009](#)).

While empirical studies show that in situations where annotation costs are variable and unknown, such as when labeling costs are a function of elapsed annotation time, learned cost models may be trained to estimate proper annotation durations using cost-sensitive active learning ([Haertel et al., 2008](#)).

On the other hand, the computing cost of an algorithm is an important factor to consider. In this part, we look at the time complexities of the active learning algorithms outlined above when it comes to choosing the best instance for labeling. [Cohn et al. \(1996a\)](#)'s active learning methods, which use a blend of Gaussians and locally weighted regression,

outperform feedforward neural networks. Which are costly to compute variance estimates and retrain. Training is linearly proportional to the number of data instances when using a mixture of Gaussians, but prediction time is independent. On the other hand, there is no training time for a memory-based model like locally weighted regression, but there are prediction costs.

Table 3.1 presents the baseline specification review summary following an in-depth evaluation of several methodologies, allowing one to choose an acceptable strategy for baseline specification with the goal of minimizing overall training label cost. The table essentially provides the *reference*, *approach*, and, lastly, the *outcome description*.

3.4 Active Learning and Application

‘Does active learning work?’ is an important question. The overwhelming empirical findings in the literature imply that it does (e.g., the majority of papers in the bibliography of this chapter). Consider how active learning technologies are rapidly being used in a range of real-world applications by software businesses and large-scale research programs like CiteSeer, Google, IBM, Microsoft, and Siemens (Settles, 2011). Active learning approaches appear to have progressed to the point of practical usage in many scenarios, based on several published findings and rising industry acceptance.

Nonetheless, in the majority of published outcomes, active learning reduces the amount of annotated samples instances required to achieve a predefined level of accuracy. Even for basic query algorithms like uncertainty sampling, this is generally the case. According to Tomanek and Olsson (2009), 91% of academics who used active learning in large-scale labeling tasks were able to completely or substantially accomplish their goals. Regardless of these findings, according to the study, 20% of respondents said they would not utilize active learning since individuals expressed uncertainty regarding the effectiveness of active learning within their case. Probably because when active learning is used in practice, extra complexities emerge (performance cost). This section examines a few of the domain-specific active learning solutions used in Remote Sensing practice (Goetz et al., 1983; Jarvis, 1983).

3.4.1 Remote Sensing

Many interesting geospatial applications employing large geographic image datasets are becoming feasible due to the development of machine learning. Land use and land cover classification (Castelluccio et al., 2015; Tracewski et al., 2017), identification and comprehension of patterns and interests in urban environments (Hu et al., 2015b; Albert et al., 2017), geospatial pattern recognition (Zhou et al., 2014; Cordts et al., 2016), and content-based image retrieval (Ferecatu and Boujemaa, 2007; Wan et al., 2014) are just a few of the remote sensing challenges that can benefit from the active learning methodologies. Image geolocalization (prediction of the geolocation of a query image) is another major research field (Lin et al., 2013b, 2015).

It is expensive to get good annotated data in hyperspectral imaging for remote sensing applications. Liu et al. (2016) presented the hyperspectral image classification approach to solving this problem, in which their system picks training samples that optimize two selection criteria i.e., representativeness and uncertainty. Their method was tested against

Table 3.1: Baseline Specification Review Summary.

Reference	Approach	Outcome Description
Initial Training Sample Selection		
Yuan et al. (2011)	fuzzy-c clustering	center, border, hybrid based
Sun and Hardoon (2010)	prelabeled	already labeled
Warmuth et al. (2003)	random	randomly from each class
Cebron and Berthold (2007)	density	high potential score around dense area
Kang et al. (2004)	k-means clustering	representative cluster point
Batch Mode		
Xu et al. (2007)	SVM	variance among batch
Hoi et al. (2006a)	sub-modular functions	varied and informative batch
Batch Size		
Bach and Badaskar (2007)	optimization function	maximizes diversity
Chakraborty et al. (2014)	quality and complexity	lower adaptivity increase, less computing bottleneck
Lourentzou et al. (2018)	ML application	depending upon, dataset and problem
Batch Diversity		
Brinker (2003)	diversity metric	high decision boundary angles
Hoi et al. (2006a)	fisher information matrix	eliminate duplication
Li and Sethi (2006)	conditional error	eliminate duplication
Hoi et al. (2009)	semi-supervised SVM	eliminate duplication
Baram et al. (2004)	kernel furthest first	farthest away samples
Nguyen and Smeulders (2004)	clustering	choose cluster centroid
Variable Labeling Cost		
Kapoor et al. (2007)	rule-based paradigm	labeling and misclassification
Culotta and McCallum (2005)	rule-based paradigm	label and categorizing difficulty measure

a range of other classification algorithms that used various query methods i.e., random sampling, maximum uncertainty sampling, and QBC, showing that, by actively selecting training samples, the suggested algorithm was able to obtain greater accuracy with fewer training examples (Tuia et al., 2011).

Collecting ground truth, particularly in developing and rural regions, is difficult, and manually labeling a huge collection of training data is expensive (Chen and Zipf, 2017). To address this issue, Chen and Zipf (2017) recommended using satellite imagery and Volunteered Geographic Information (VGI) (Coleman et al., 2009) data to categorize it. They used a feed-forward neural network (Bishop and Nasrabadi, 2006) and two classic CNNs: LeNet (LeCun et al., 2015) and AlexNet (Krizhevsky et al., 2012), in their strategy. When compared to Deep-OSM (Lange et al., 2020) and MapSwipe (Herfort, 2018), their first testing results showed that their performance in particular, F1 score and accuracy is much higher than DeepOSM but not as excellent as the MapSwipe where three participants vote on each image. Deep-OSM can anticipate misregistered freeways in OpenStreetMap (OSM) data by identifying freeways and characteristics from satellite images using OSM data to train neural networks (Mooney and Minghini, 2017); Deep-OSM's deep learning architecture is a basic one-layer CNN. MapSwipe is a smartphone technology that let participants identify landmarks and freeways in pictures.

An excellent demonstration of applying active learning methods in remote sensing is to select the n most ambiguous samples for segmentation of multispectral images using SVMs for binary classification (Mitra et al., 2004b); Their query technique chooses the sample that is closest to each binary SVM's current separating hyperplane. Ferecatu and Boujemaa (2007) employed an SVM classifier in their active learning approach for remote-sensing visual retrieval; Their criterion for selection was to keep the number of potential photographs given to the user to a minimum.

Acquiring annotated data for remotely sensed image-based land cover classification is time-intensive and expensive, especially in remote locations. As a first step toward the goal of building classifiers with as minimal annotated training points as possible, Rajan et al. (2008) devised an approach that effectively updates current classifiers using minimum labeled data points; they select the unlabeled data that enhances the gain ratio between the posterior probability density function computed from the current training set and the (new) training set acquired by including that sample.

Rajan et al. (2008) employed pool-based active learning where the classifier adopts when there is a significant difference in the spectral signatures between labeled and unlabeled data, whereas the gain ratio is computed by the divergence (McCallumzy and Nigamy, 1998). Making an effective strategy for categorizing a set of spatially/temporally connected images with different spectral characteristics.

For multi-class remote sensing image classification, Tuia et al. (2011) presented two batch-mode active learning methods. The first approach incorporates kernel space variety into SVM margin sampling, while the second is an entropy-based variation of the query-by-bagging algorithm. Demir et al. (2010) also examined a range of multi-class SVM-based batch-mode active learning algorithms for interactive remote sensing image classification, with one result suggesting cluster-based diversity criteria for relevant query selection. Patra and Bruzzone (2010) also presented a quick cluster-assumption-based active learning approach, but only took the uncertainty criteria into account. In follow-up work, Patra and Bruzzone (2011) suggested a batch-mode active learning approach for handling multi-

class classification issues using a SVM classifier with an open agent architecture (Cheyer and Martin, 2001), which takes into account both uncertainty and diversity requirements. The efficiency of the suggested approach was validated by their findings on two separate datasets (hyperspectral and multispectral). The number and spectra of electromagnetic radiation that each band provides are the key differences between hyperspectral and multispectral (Adam et al., 2010).

Annotators are asked to annotate data samples in the most active learning approaches in the literature. In a recent example by Huijser and van Gemert (2017) annotators are asked to annotate the decision boundary. They employ a linear classification model in their strategy. A deep generative model (Goodfellow et al., 2014; Salimans et al., 2016) was also employed to generate samples based on a limited number of labeled samples in the procedure. From the machine learner’s perspective, the majority of existing active learning research is focused on the mechanics and advantages of identifying relevant instances for labeling (Settles, 2011). The fact that users have no influence over which instances are labeled is a disadvantage of the query approach (Bernard et al., 2017), which may impair the performance of an active learning model (Huang et al., 2017). User-based visually-supported active learning procedures, suggested by Seifert and Granitzer (2010), allow the user to pick and label examples provided by a machine learner. Their research revealed that limiting human input to only tagging cases that the algorithm chooses is inefficient. Giving users a more active part in visual example selection and adjusting labeling procedures on top of customized visualization approaches can improve labeling efficiency.

Júnior et al. (2017)’s work on GPS trajectory categorization shows that active learning may be used with human-in-the-loop (Wu et al., 2021) to assist domain experts with semantic labeling of movement data. They pose three research questions: (1) Is there a machine learning approach that allows for the development of a decent classifier for automatic trajectory classification with a smaller number of human-labeled trajectories? (2) Does active learning work well with trajectory data? (3) How can we make it easier for the user to label trajectories? To address these research problems, they created ANALYTIC, a web-based interactive tool that uses active learning and a simple interface to visually help domain experts perform GPS trajectory categorization. To begin with trajectory labeling, users may choose from six (conventional ML) classifiers (Ada_boost, decision tree, Gaussian naive Bayes, k-nearest neighbours, logistic regression, and random forest) and one of three query procedures (uncertainty sampling, QBC, and random sampling). Only binary categorization is supported by their interactive interface. They also provide a series of empirical evaluation studies using three different datasets for trajectories (animals, fishing vessels, and GeoLife). Their examples showed how ANALYTIC interface may aid the domain expert in the active learning process, particularly in trajectory annotation, by providing a range of visual solutions that make the labeling work easier. They found that by learning from sets of manually labeled data, ML systems may infer semantic labels established by domain users from trajectories. Active learning techniques, in particular, can minimize the number of trajectories that need to be categorized while maintaining high-performance metrics. Their ANALYTIC application shows the annotation process for subject specialists.

In terms of using a human-in-the-loop system to help domain experts with labeling, PEARL (Anwar, 2022), an Artificial Intelligence accelerated platform for quick land cover mapping, was released for experimental usage by Development Seed (Development Seed, 2022) and Microsoft AI for Earth (Microsoft, 2022a). PEARL is a novel way to quick,

accurate land cover mapping that combines human intelligence with scalable AI. It takes advantage of Microsoft’s Planetary Computer (Microsoft, 2022b) initiative’s capabilities and research to drastically decrease the time and effort required to build land cover maps, allowing scientists and researchers to focus on the most critical environmental and climatic research problems. PEARL’s rapid model retraining is a major feature. Users may retrain the model in the browser on the fly to generate checkpoints that can be used to progressively enhance the model in the direction they choose. PEARL’s initial release includes two fully convolutional network segmentation (Long et al., 2015) models that were trained with nine and four land cover classes, respectively, using labeled data from the Chesapeake Conservancy’s dataset (Conservancy, 2022). The overall F1 score for each of these beginning models is around 90%. Users can enhance the model’s performance in a certain region and even create additional Land Usage and Land Cover (LULC) classes.

Table 3.2 summarizes the literature review on the topic of active learning for remote sensing applications. The table covers the numerous active learning *query selection methodologies* and *machine learning models* used, *goal* of the work, and what *sensor* data was used. Here, KRR: Kernel Ridge Regression, QBB: Query-by-Bagging, LAI: Leaf Area Index, GPR: Gaussian Process Regression, VHGR: Variational Heteroscedastic Gaussian Regression, KNN: K-Nearest Neighbor, RBFK: Radial Basis Function Kernel, DBN: Deep Belief Network, FFNN: Feed-forward Neural Network, and MG: Multivariate Gaussians.

Whilst, active learning has been effectively applied to a wide range of challenges in several fields, no structured and complete assessment of active learning methodologies has been conducted. For example, much of the research has been disjointed, with various datasets used in different application domains and inadequate coherence to assess active learning methodologies.

Table 3.2: Literature Review of Active Learning and Remote Sensing Application.

Reference	Goal	ML algorithm	Query method	Sensor
Verrelst et al. (2016)	LAI mapping	KRR	entropy QBB, angle-based diversity	Sentinel-3 OLCI
Upreti et al. (2019)	vegetation cover	GPR	euclidean distance-based, cluster-based diversity	Sentinel-2
Zhou et al. (2020)	Chlorophyll mapping	GPR	entropy QBB, euclidean distance diversity	Landsat-8 OLI
Verrelst et al. (2020)	vegetation nitrogen	VHGR	entropy QBB, euclidean distance diversity	EnMAP
Ahmad et al. (2018)	image scene classification	SVM fuzzy KNN	random selection, distance to the boundary	AVIRIS, ROSIS-03
Pasolli et al. (2013)	image scene classification	SVM	random selection, margin sampling, breaking ties, distance from the closest support vector	QuickBird
Krishnapuram et al. (2005)	mine detection	logistic classifier	mutual information	hyper-spectral electro-optic
Luo et al. (2005)	underwater zooplankton	SVM	breaking ties, least certainty	SIPPER II
Demir et al. (2010)	image scene classification	SVM with RBFK	uncertain sampling, kernel-clustering	LIDAR
Liu et al. (2016)	image scene classification	DBN	random sampling, QBC, maximum uncertainty sampling	ROSIS-3
Chen and Zipf (2017)	label images with buildings and roads for humanitarian aids	FFNN LeNet AlexNet	random sampling	MapSwipe Data
Mitra et al. (2004b)	segmentation of images	SVM	uncertain sampling, distance from hypersurface	IRS-1A
Rajan et al. (2008)	land cover classification	MG	posteriori probability distribution, KL divergence	AVIRIS, IKONOS
Tuia et al. (2011)	image scene classification	SVM	margin sampling, entropy-based QBC	AVIRIS, ROSIS, QuickBird
Patra and Bruzzone (2010)	image scene classification	SVM	cluster-based uncertainty	AISA Eagle
Patra and Bruzzone (2011)	image scene classification	SVM	cluster-based diversity & uncertainty	AISA Eagle

3.4.2 Abbreviating Training Cost

Over the years, due to the enrichment of paired-label datasets, supervised machine learning has become an important part of any problem-solving process. Active Learning gains importance when, given a large amount of freely available data, there's a lack of expert's manual labels. In Active Learning, the classifier ranks the unlabeled pixels based on predefined heuristics and automatically selects those that are considered the most valuable for improvement; the expert then manually labels the selected pixels and the process is repeated. The system builds the optimal set of samples from a small and non-optimal training set, achieving a predefined classification accuracy.

Traditional supervised learning, such as binary or multi-class classification, is used to create high-predictive accuracy models from labeled training data. Labeled data, on the other hand, isn't cheap in terms of labeling costs, time spent, or the number of instances consumed (Zou and Hastie, 2005). As a result, the objective of active learning is to maximize the effective use of labeled data by allowing the learning algorithm to pick the instances that are most informative on their own (Fu et al., 2013). In comparison to random sampling, the goal is to get better results with the same amount of training data or get the same results with fewer data (Vapnik, 1999).

Active Learning is an iterative process that cycles over selecting new examples and retraining models. In each iteration, the value of candidate instances is calculated in terms of a usefulness score, and the ones with the highest scores are queried once (Yu et al., 2020) and its corresponding label is retrieved. The instance's value usually refers to the reduction of uncertainty in the context of "To what extent does knowing the label of a particular instance aids the learner reducing the ambiguity over instances that are similar?" (Fu et al., 2014). In uncertainty sampling, one of the most common methods measures the instance's value in terms of predictive uncertainty (Settles, 2009), which leads the active learner to choose the cases where its current prediction is the most ambiguous. Almost all predictions are probabilistic, as are the measurements used to quantify the level of uncertainty, such as entropy (Hüllermeier and Waegeman, 2021).

For many real-world learning challenges where there is a limited collection of labeled data and a large amount of unlabeled data, pool-based active learning can be used. Here, samples are chosen greedily from a closed (i.e., static or non-changing) pool using an information measure such as entropy (Lewis and Gale, 1994b). Many real-world machine learning areas have been examined using the pool-based active learning; these include (but are not limited to) text classification (Tong and Koller, 2001; Hoi et al., 2006a), information extraction (Settles and Craven, 2008), image classification and retrieval (Zhang and Chen, 2002), video classification and retrieval (Hauptmann et al., 2006), speech recognition (Tur et al., 2005), and cancer detection (Liu, 2004).

On the other hand, uncertainty sampling in active learning is perhaps the most basic and often used query framework. In this paradigm, an active learner inquires about situations for which there is the least amount of certainty of how to classify them (Wang and Brenning, 2021). If the underlying data distribution can be completely categorized by some hypothesis, then drawing $O(1/\varepsilon)$ random labeled examples, where ε is the maximum desirable error rate, is enough, according to the presumably approximately accurate (PAC) learning model (Devonport et al., 2021). According to the research (Settles, 2009; Carbonneau et al., 2018), considering a pool-based active learning scenario where we can get some number of unlabeled examples for free (or very cheaply) from distribution; The

(unknown) labels of these locations on the real line are a sequence of zeros followed by ones, and objective is to find the place (decision boundary) where the transition happens while paying as little as possible for labels. Because all additional labels can be inferred, a classifier with an error less than ε can be attained with just $O(\log 1/\varepsilon)$ queries, leading to an exponential decrease in the number of classified cases. Of course, this is a basic binary toy learning challenge that is one-dimensional and noiseless.

Further exploration of *Abbreviating Train Cost* over particular domain application is presented in Chapter 8.

3.5 Summary

Recently, for many real-world learning situations with a small collection of labeled data and a significant number of unlabeled data, this chapter provides an in-depth introduction to active learning theory. Following that, various sampling and query selection approaches are described. In addition, regardless of the domain application, extensive empirical study on baseline specification employing active learning is outlined. Finally, the chapter presents a comprehensive overview of current active learning approaches in remote sensing and examines their merits and drawbacks in the context of practical applications such as reducing classifier training costs.

Chapter 4

Evidence Function Model

“Is it feasible, regardless of the Machine Learning algorithm, to create an Artificial Intelligent system that can predict the likelihood of a new instance being incorrectly labeled by a previous learner without the context of the true label?”

— Prof. Luís Rato

Evidence? According to the Oxford definition, evidence is the collection of findings or materials that can be used to determine if a notion or statement is true. In philosophical terms, it is defined as an essential understanding of a discipline or art that also encapsulates core logical ideas. The notion of evidence is critical in the research-oriented area since it collects all factoids available and uses it in a range of methods to determine when a claim is true or false. Under statistical evidence, observations are analyzed using a probabilistic model (Royall, 2004). Thus, to dismiss or confirm a claim, statisticians collect knowledge from scientific occurrences, materials, and instruments like hypotheses, experiments, and models with prior knowledge. For example, in the Bayesian approach, the evaluation is based on a certain posterior probability that attempts to measure the researcher’s conviction in the hypothesis testing. The more the belief in the truth of a theory, the higher the possibility of the assumption being correct (Kruglanski, 1989).

In the last decade, the usage of distances and divergences has considerably changed from the statistical, probability, and information theory studies into other scientific areas like machine learning, biomedical sciences, engineering, and ecology (Lubischew, 1962; Efron, 2004; Markatou and Sofikitou, 2019). The statistical distance (or divergence) can be referred to as a distance between random variables, probability distributions, or between a single point and a population (Wootters, 1981a). Statistical distances are used for measuring the goodness of fit test, estimation, prediction or model selection (Lindsay et al., 2014). Alternatively, statistical distances can effectively be used to construct evidence functions (Iwamura et al., 2004) that provide an effective way of hypothesizing parametric and semi-parametric models (Chen and Ho, 2008).

In ML, the study of “The Estimation of Prediction Error” can drive algorithms in the same manner that it does for human cognitive behaviour (Sugrue et al., 2005). A learner could use the goal of decreasing error response as a strategy to improve learning. In such a strategy, while learner’s prediction may provide a false value, and the expected result

is a true value, the learner would be retrained trying to optimize model’s score. This method of machine learning, known as error-driven learning, aims to encourage learning by simulating (Wigmore, 2022).

Although the use of distances and divergences in scientific areas like machine learning have increased drastically recently, the lack of any available algorithm able to give insight into the uncertainty prediction using the relation between the train and test sets remains our prime motivation. In this chapter, we discuss an approach for ‘misclassification detection’ through ‘confidence estimation’ and ‘notion of evidence’ and present a brief literature review related to its interpretation, portraying to develop a theorized idea of using distance (between un-seen observation and train set) as uncertainty prediction.

The remainder of the chapter is organized as follows: Section 4.1 talks about what is confidence estimation and the notion of evidence and provides a general and statistical meaning of it with respect to ML models; Section 4.2 explains the Mahalanobis and Euclidean distance; Section 4.3 details the modeling of Evidence Function Model, where a general understanding of Mahalanobis distance and how it has been used in Evidence Function Model is presented; Section 4.4 articulates where Evidence Function Model can be used.

4.1 Confidence Estimation and Notion of Evidence

Confidence estimation is a well-studied area of both parametric and non-parametric statistics (Vickers, 2005). In supervised learning, mainly in a classification problem, the model should learn from known examples (label-paired data represented by attribute vectors with corresponding labels) and predict the unknown label for a new example (Caruana and Niculescu-Mizil, 2006). In this regard, Koriati et al. (1980) presented “Reasons for Confidence” for answering the question “How to classify new instances but then choose only those with the highest confidence?”. Similarly, Ferrettini et al. (2020) worked upon calculating the confidence score in classifying the instance x for a class C ; the hedged predictions for the labels of new objects included quantitative measures of their accuracy and reliability. These quantitative measures are probably valid under the assumption of randomness, in machine learning (Gammerman and Vovk, 2007).

In the medical domain, the reliability associated with a given prediction is essential in developing clinical tools. Nouretdinov et al. (2011) proposed a “Transductive Conformal Predictor (TCP)” for MRI images; TCP generates the most likely prediction with a valid measure of confidence and the set of all possible predictions for a given confidence level.

In statistics, while referring to the ‘confidence estimation’, there is a long literature related to the use of distance measures. Markatou and Sofikitou (2019) created an estimator to measure the goodness of fit for statistical models using the minimum distance principle. However, having multi-spatial data with multivariate classes, the data distribution does not precisely reflect the ‘clustering’ phenomenon in terms of feature space (Poggi et al., 2017b,a); in such a situation, data needs to be transformed so that the data-space representation forms a clustering or explanation for the prediction. Gammerman and Vovk (2002), presented a computing model for confidence estimation to predict high-dimensional IID (independent and identically distributed) data. Their method is based on realistic approximations of the algorithmic ‘theory of randomness’ metrics of confidence. Algorithmic

randomness is an area of mathematics that uses computability theory to create a formal description of randomness (Li et al., 2008; Rute, 2016). Gammerman and Vovk (2002) describe an SVM approach and sketch the fundamental concepts of algorithmic randomness and its approximation.

On the other hand, while referring to the ‘notion of evidence’, Juutilainen and Rönning (2007) stated that “using the distance weighted k-nearest-neighbour method, the distance reflects the expected squared prediction error when a quantitative response variable is predicted based on the training dataset”. This leads to the observation that the distance can be applied, for example, in assessing the uncertainty of the prediction (Tibshirani, 1996). For example, considering a binary classification problem, SVMs work by determining a separating hyperplane between the two classes, and for a new point P , SVM classifies it as class A or B according to which side of the hyperplane P is in (Cortes and Vapnik, 1995). Here, the confidence score (in terms of this distance measure) is the relative distance to the hyperplane.

Kobayashi (2019) asserted that “evaluation of prediction at a query point by the current prediction model, is impacted by the information given by the training dataset about a query point”. Further, Markou and Singh (2003) adopted a threshold-based joint density function approach in detecting the similarity between the training data and a new observation. Markatou and Sofikitou (2019) also stated that statistical distances could be interpreted as loss functions and used as evidence functions. Alternatively, the Bayesian decision theory (Feldman and Yakimovsky, 1974) gives a decision that minimizes the expected probability of misclassification as long as the true class distributions are given (Han et al., 2015). In this regard, the quadratic discriminant function (Kimura et al., 1987) and Mahalanobis distance (McLachlan, 1999), are the most popular discriminant functions (Welch, 1939) derived from a multidimensional normal distribution (Lindsay et al., 2008).

Learners’ uncertainty about the picked examples is somehow not explained by traditional/common uncertainty sampling (Yang et al., 2018). Sharma and Bilgic (2017) filled this need by employing an evidence-based paradigm. The authors concentrated on two forms of uncertainty for the model: conflicting-evidence uncertainty, given the existence of considerable but inconsistent evidence for each class and insufficient evidence, given the existence of insufficient evidence for either class. In their empirical assessments, using naive Bayes over different datasets for the binary classification tasks, they found that conflicting-evidence uncertainty outperforms in terms of learning efficiency both traditional and insufficient-evidence uncertainty sampling. Sampling values were selected randomly from their posterior distributions. Further explaining, instances of uncertainty due to conflicting evidence has a lower density in the labeled set than insufficient evidence. This means there is less support in the training data for the perceived conflict than for the insufficiency of the evidence.

Referring to all the previous work and approaches in the area of ‘confidence estimation’ and ‘notion of evidence’, we propose a statistical distance-based method that estimates the uncertainty of the expected prediction at a new query point. The general idea is that the distance between a new observation and the training dataset distribution should reflect the expected prediction about the new query point. This means that the proposed approach evaluates the prediction by estimating the relative distance of the hypothesized prediction to the data-space distribution.

The measurement of misclassification can be a pivotal factor, especially in domains where the reliability associated with a given prediction is essential and the measurement of uncertainty is crucial, like in the case of the medical domain or weather forecast where the accuracy of the model is at the highest expectancy. Additionally, knowing the prediction uncertainty could also help further research areas like ‘Active Learning’ (Settles, 2009) and ‘Disagreement based Active Learning’ (Hanneke, 2014). The reasoning is that given a large amount of freely available data where labels are insufficient providing a ‘detection of misclassification’ could help in choosing and generating labeled datasets where human input is required for data with higher uncertainty. This creates a motivational link with previous Chapter 3 Active Learning.

4.2 Mahalanobis Distance

A statistical distance quantifies the distance between two statistical objects, which can be two random variables within the same distribution, multiple density estimation or samples, or the distance between independent points (Wootters, 1981b). Two commonly used distances are Euclidean and Mahalanobis.

In Euclidean spaces, independent variables are usually represented in orthogonal axes and the distance between any two points can be measured using the standard Euclidean distance (Bell, 1923). In statistics, however, correlated variables may have different scales with axes no longer being orthogonal with the Euclidean distance losing its meaning. Under a Gaussian assumption, the Mahalanobis Distance (MD) (Mahalanobis, 1936) can be used instead. The MD can be seen as first performing a re-scaling of the variables to become uncorrelated with normalized variance, and then using the usual Euclidean distance on the transformed variables. The MD can also be used to measure the distance from a point to a multivariable distribution specified by its mean vector and covariance matrix. Thus, for any given point, the larger the MD, the further away from the centroid. Figure 4.1 shows an example of Mahalanobis Distance, where the unit distance of Y (purple dots) is much smaller compared to the unit distance of Y' (yellow dots) from the center of a distribution X (blue tiny dots). These distances would have been the same using the Euclidean Distance as it does not consider the distribution co-relation. The lemma stating, “The Mahalanobis Distance can be used to measure the distance from a point to a multivariable distribution specified by its mean vector and covariance matrix” (Danielsson, 1980; Darmochwał, 1991; Barhen and Daudin, 1995; McLachlan, 1999; De Maesschalck et al., 2000) is provided in Appendix A.2 with proof.

The Mahalanobis Distance (Δ) between point \mathbf{x}_i and a distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ is given by Equation (4.1).

$$\Delta = \sqrt{(\mathbf{x}_i - \boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})^\top} \quad (4.1)$$

4.3 Evidence Function Model

The following scenario aims at better understanding what we mean by the Evidence Function Model (EFM):

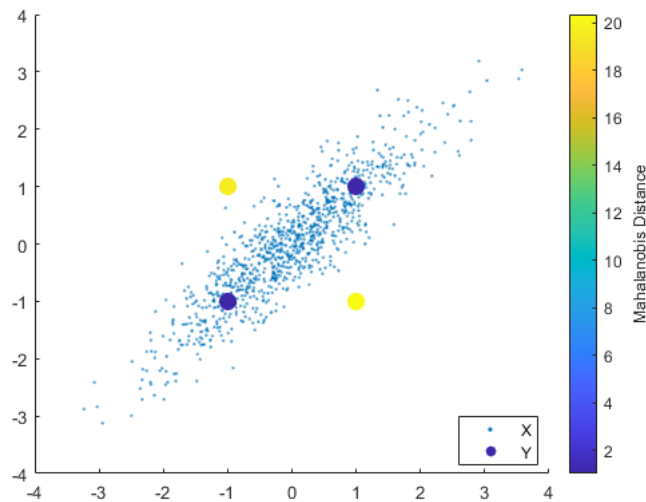


Figure 4.1: Example of Mahalanobis distances. A distribution of points described by 2 attributes (the X and Y axis).

Consider a multi-classification problem to distinguishing images of dogs, cats, and mice. Given a new input without true class, is it feasible to determine, whether the new input is classified properly or misclassified? We are not discussing the related confidence of the classification here, but rather a binary metric indicating whether a **classification** is **correct** or **incorrect**.

Mathematically speaking, assume that the training set $X = (x_1, x_2, \dots, x_n)$ consists of n observations, with each $x_i = (t_i; \text{label})$, where t_i is the predictor or feature vector and label is the class.

Our goal is to build the (best-fit) approach that outputs the likelihood measure of the classifier prediction P being misclassified based on X , using built classifiers $C(X) \rightarrow P$, or, in other words, outputting the prediction uncertainty (0/1) over unseen data.

Let us first define the following high-level terminology before moving further:

1. Classification: Given M features and N labels/classes, build a classifier that predicts a class;
2. Cross-validation: In order to fine-tune the modeling parameters, evaluate the performance of ML models on subsets of the available input data;
3. Dataset-split: Partition the available data;
4. Feature Transformation: Transform existing feature space into a new (generalized) feature space.
5. Distance: A distance between two points in data/feature space representation.
6. Mahalanobis Distance (Δ): A distance between point and a distribution, using distribution mean μ and covariance Σ .

Table 4.1 lists the symbol names and their meanings for convenience of explanation and to keep the integrity of the naming standard.

Table 4.1: Notation and their explanation.

Notation	Interpretation
Δ	Mahalanobis Distance
D	Whole dataset (Train+Test)
F	Feature set
C	Different classes
K	Total number of classes
Te	Test set
Tr	Train set
p	a point
N	Total number of points
A	a small subset of Tr
B	a large subset of Tr (note: $A \cap B = \phi$ and $A, B \subset Tr$)
B_c	a subset of B with only c class, where $c \in C$
ΔY_X	A vector Δ between all points in Y to X (μ_X)
μ_X	Mean of X
Σ_X	Covariance of X
C_{alg_X}	Classifier trained using alg Algorithm over X
$P_{Y_{C_{alg_X}}}$	Prediction made over Y using C_{alg_X} model
U_{P_Y}	Uncertainty of the prediction $P_{Y_{C_{alg_X}}}$ (value: 0 and 1)

The generalized formulation of the problem statement is illustrated in the Figure 4.2. Here, given a train set Tr , a classifier model C is developed that predicts P over the test set Te . Now, “Can we leverage feature space information between train and test sets to detect the misclassification or uncertainty of predictions made?”

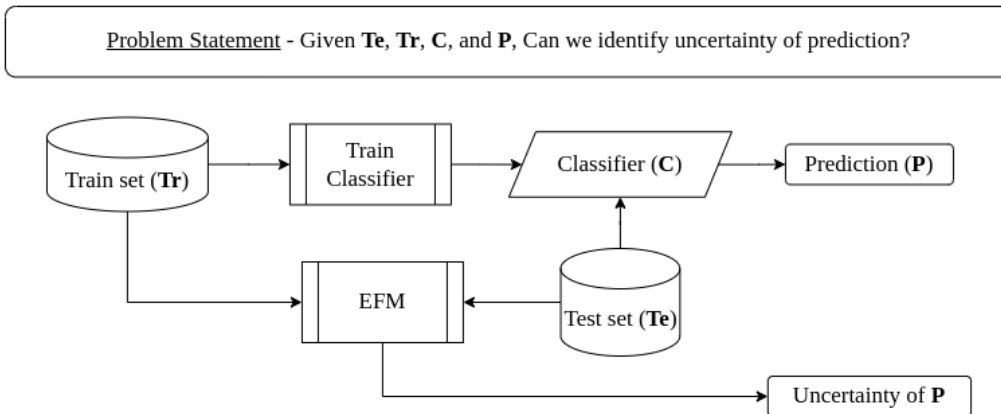


Figure 4.2: Generalized problem statement.

The idea of using distance Δ between unseen observations Te and the train set Tr and the identification of prediction P uncertainty is explored and developed further as **EFM**.

The proposed approach EFM, is a binary model that detects ‘misclassification of the prediction’, and to built it, the process is divided into four distinct modules; the connection between them is shown in Figure 4.3:

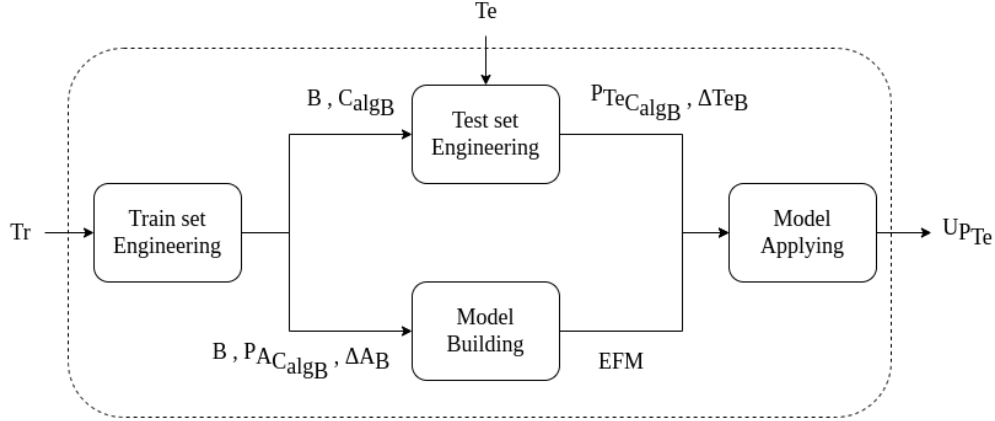


Figure 4.3: EFM: main modules and data-flow.

- **Train set Engineering** takes the train set, Tr , as input and is responsible for four tasks: dataset split, feature transformation, train a classifier, and make predictions, generating four outputs: B , ΔA_B , C_{algB} , and $P_{A_{C_{algB}}}$, respectively;
- **Model Building** takes B , $P_{A_{C_{algB}}}$, ΔA_B as inputs from *Train set Engineering* module and builds a model, outputting EFM;
- **Test set Engineering** takes B , C_{algB} from *Train set Engineering* module and a test set (Te) as inputs and is responsible for two tasks: feature transformation and make a prediction. It generates two outputs: $P_{TeC_{algB}}$ and ΔTe_B ; and
- **Model Applying** takes the outputs of modules *Model Building* and *Test set Engineering* as inputs and produces the uncertainty of the prediction $P_{TeC_{algB}}$, $U_{P_{Te}}$ (value: 0 and 1).

Train set Engineering. As mentioned, this module is responsible for four tasks: dataset split, feature transformation, train a classifier, and make a prediction. Therefore, we have broken down module into 4 steps, and Figure 4.4 shows the data flow between them.

Step 1: Given Tr , the first step divides it into two subsets **A**, a smaller and **B**, a larger. Therefore, referred to as a dataset split task. **Note:** $A \cap B = \phi$ and $A, B \subset Tr$, and Tr consist of features F and classes C .

For example, in case of images, one image can be considered as **A** while the remaining as **B**. In the case of other data, divide a smaller and larger subsets of the data points.

Step 2: Given A and B from step 1, step 2 divides B into K class-wise subsets, one for each class $c \in C$, resulting B as a $\bigcup_{c=1}^K B_c$.

ΔA_{B_c} is calculated using Equation 4.2 for each point p in A . Where, N is a total number of points in A , μ_{B_c} is a mean of B_c , and Σ_{B_c} is a covariance of B_c .

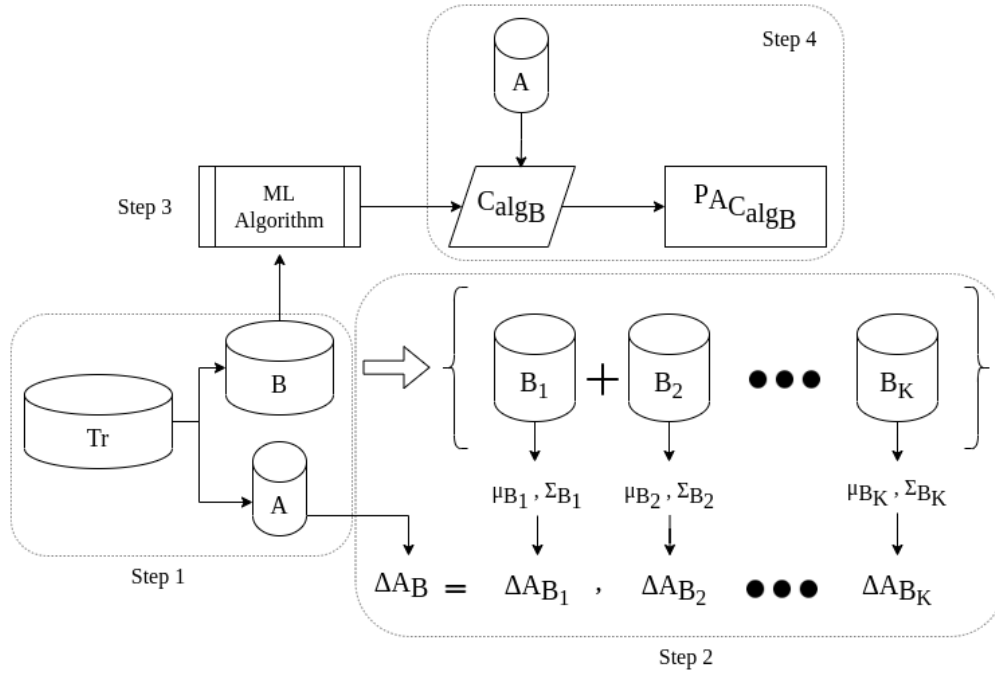


Figure 4.4: *Train set Engineering*: overall process flow, step 1 to step 4.

$$\Delta A_{B_c} = \sqrt{(Ap - \mu_{B_c}) \Sigma_{B_c}^{-1} (Ap - \mu_{B_c})^\top} \text{ where, } \forall p \in N \quad (4.2)$$

At the end, for each point p in A , $\Delta \mathbf{A}_B = \{\Delta A_{B_1}, \Delta A_{B_2}, \dots, \Delta A_{B_K}\}$. Therefore, referred to as a feature transformation task.

Step 3: Given B from step 1, any ML algorithm (for example, distance-based, tree-based, or neural network-based) can be used to train the classifier C_{alg_B} . Therefore, referred to as a train classifier task.

For example, if a distance-based K-Nearest Neighbor (KNN), a tree-based Extra Tree (ET), or neural network-based Convolutional Neural Network (CNN) classifiers are used to train over B then the resultant models would be C_{KNN_B} , C_{ET_B} , and C_{CNN_B} .

Step 4: Given A from step 1, for each point p in A , a prediction P_A is made using a classifier C_{alg_B} from step 3. In short, $C_{alg_B}(A) \rightarrow P_A$ is calculated. Referred as $P_{A_{C_{alg_B}}}$. Therefore, referred to as a make predictions task.

At the end of *Train set Engineering* module, outputs B , $P_{A_{C_{alg_B}}}$ and ΔA_B are passed to *Model Building* module; and outputs B and C_{alg_B} are passed to *Test set Engineering* module.

Model Building. The second module is responsible for training a EFM using the inputs B , $P_{A_{C_{alg_B}}}$ and ΔA_B from previous module.

The training of the EFM can be done using any ML algorithm which takes two features as input: Mahalanobis Distance ΔA_B and Prediction $P_{A_{C_{alg_B}}}$ made over A using a classifier

C_{alg_B} . Meaning, for EFM training, feature set = $(\Delta A_B, P_{A_{C_{alg_B}}})$ and True Class Label C will come from B .

Model Building module result **EMF** is passed to *Model Applying* module.

Test set Engineering. This module is responsible for two tasks, feature transformation and make a prediction. It uses as inputs B , C_{alg_B} from *Train set Engineering* module and test set Te . Therefore, we have broken down module into 2 steps process, and Figure 4.5 shows the data flow between them. (**note:** step 1 is similar to as *Train set Engineering* module step 2 process. Here, instead of A , we are using here Te .)

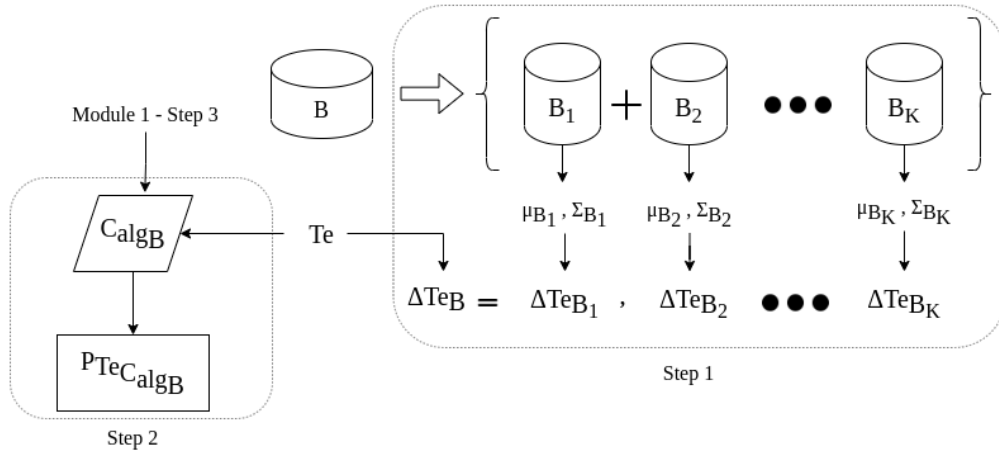


Figure 4.5: *Test set Engineering*: overall process flow, step 1 and step 2.

Step 1: Given B from step 1 and Te , the next step is to divide B into K class-wise distribution, resulting B as a $\bigcup_{c=1}^K B_c$.

ΔTe_{B_c} is calculated using Equation 4.3 for each point p in Te . Where, N is a total number of points in Te , μ_{B_c} is a mean of B_c , and Σ_{B_c} is a covariance of B_c .

$$\Delta Te_{B_c} = \sqrt{(Te_p - \mu_{B_c}) \Sigma_{B_c}^{-1} (Te_p - \mu_{B_c})^\top} \text{ where, } \forall p \in N \quad (4.3)$$

At the end, for each point p in Te , $\Delta Te_B = \{\Delta Te_{B_1}, \Delta Te_{B_2}, \dots, \Delta Te_{B_K}\}$. Therefore, referred to as a feature transformation task.

Step 2: Given Te , for each point p in Te , a prediction P_{Te} is made using a classifier C_{alg_B} from step 3 of *Train set Engineering* module. In short, $C_{alg_B}(Te) \rightarrow P_{Te}$ is calculated. Referred as $P_{Te_{C_{alg_B}}}$. Therefore, referred to as a make predictions task.

At the end of *Test set Engineering* module, results $P_{Te_{C_{alg_B}}}$, ΔTe_B are passed to *Model Applying* module.

Model Applying. Finally, this module is responsible for one task: produce the Uncertainty $U_{P_{Te}}$ (value: 0 and 1). This module uses the outputs of modules *Model Building* and *Test set Engineering* as inputs.

Given inputs, $P_{Te_{algB}}$, ΔTe_B , from *Test set Engineering* module, and the trained EFM model from *Model Building* module, it generates $U_{P_{Te}}$ using the Statistical Distance relation between the train set, Tr , and test set, Te . Therefore, referred to as an uncertainty produce task.

When there is a mismatch between the input P_{Te} and P'_{Te} calculated by EFM, it is referred as ‘classification prediction error’ or ‘misclassification detection’, making EFM a binary model. When EFM predicts 1, the EFM predicted a different class based upon the feature data space representation compared to existing feature value based predictor.

To summarize overall EFM modeling process, first of all, using a subset of train set, we transformed train set’s feature as Mahalanobis distance and prediction. Following that, we built an EFM model with engineered feature sets and tested it by changing the test set feature to Mahalanobis distance and prediction. Finally, the EFM model outputs whether or not there is ‘misclassification detected’ over test set.

4.4 Summary and Application of EFM

Recently, there has been a sharp growth in the usage of distances and divergences in scientific fields like machine learning. Our primary motive, however, remained to be the absence of a publicly accessible method that might provide information about the prediction uncertainty utilizing the relationship between the train and test sets.

First, this chapter provides a thorough analysis of recent developments in the fields of ‘confidence estimation’ and ‘notion of evidence’, covering current techniques and strategies. Then, we put forth a statistical distance-based evidence function model that identifies and assesses the degree of misclassification (in 0/1) of each independent ML model’s prediction over any new data.

The application of the proposed method is adopted in Chapter 7, where we hypothesize that it is possible to detect the misclassification of prediction using the relationship between the train and test sets for different ML models like KNN, ET, and CNN used for Sentinel-2 image scene classification in classifying six classes Cloud, Cirrus, Shadow, Snow, Water, and Other.

Additionally, in Chapter 8, we contrasted EFM’s performance in “Abbreviating Labeling Cost for Sentinel-2 Image Scene Classification via Active Learning” with entropy-based query selection procedures, demonstrating the value of EFM.

Chapter 5

Experimental Datasets

“Instincts are experiments. Data is proof.”

— Alistair Croll

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs (Russell and Norvig, 2010). It infers a function from labeled training data consisting of a set of training examples (Mohri et al., 2018), thus making supervised machine learning heavily data depended. This requires building a well-structured dataset such that the learning algorithm generalizes from the training data to unseen instances.

Before we continue, let’s go over a few things surrounding the Sentinel-2 image: Sentinel-2 image is also known as Sentinel-2 Product; each product is a size of $100 \times 100 \text{ km}^2$; Level-1C product comprises Top-of-Atmosphere (TOA) reflectances; scene is the classification of sections of satellite images into morphological categories e.g., land, water, cloud. **Note:** here on wards, we are only going to use Level-1C Sentinel-2 products.

To validate the established approaches, we employed a total of three distinct datasets. The construction of each dataset is covered in this chapter. To be specific, Section 5.1 presents an *Image Scene* dataset made up of 60 Sentinel-2 images that have been labeled into six classes; Section 5.2 describes a *Waterbody* dataset made up of 49 Sentinel-2 images that have been labeled into one class; and Section 5.3 describes two *Unlabeled* Sentinel-2 images.

5.1 Image Scene Dataset

Hollstein et al. (2016) used false-color RGB images to generate a dataset of manually tagged Sentinel-2 products. Figure 5.1 and Table 5.1 demonstrate the geographical distribution of data collection. The dataset consists of images collected globally and contains 6.6 million points (exactly 6,628,478 points) from 60 different products.

These 60 products has a wide range of surface types and each pixel is classified into one of the six classes shown in Table 5.2. The entire dataset is structured as *product_id*, *latitude*,

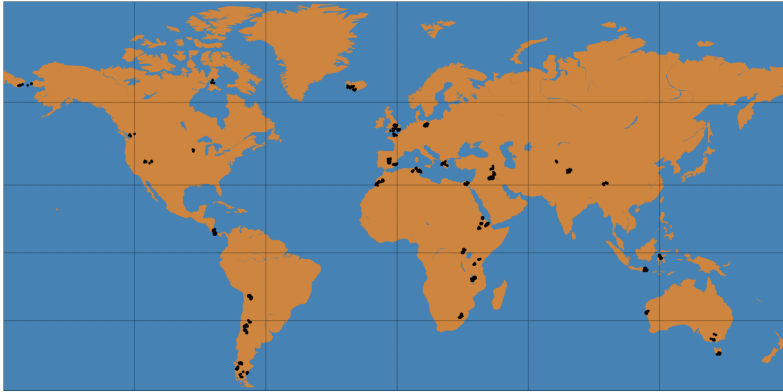


Figure 5.1: Global distribution of scenes.

Table 5.1: Selected Products Geographical Distribution.

Continent	No. of Product
Africa	14
America	12
Asia	6
Europe	22
Oceania	6
Total	60

longitude, *class*. The detailed product-wise number of points for each class can be found in Appendix A.4.

Table 5.2: Surface Types and Overall Distribution of Classes.

Class	Surface Types	Points	Distribution (%)
Cloud	opaque clouds	1031819	15.57
Cirrus	cirrus and vapor trails	956623	14.43
Snow	snow and ice	882763	13.32
Shadow	from clouds, cirrus, mountains, buildings	991393	14.96
Water	lakes, rivers, seas	1071426	16.16
Other	remaining: crops, mountains, urban	1694454	25.56
Total	-	6628478	100.00

As mentioned, [Hollstein et al. \(2016\)](#) used false-color RGB images to classify images. First of all, Level-1C products (all bands) were spatially resampled to $20m$. Afterwards, bands 1, 3 and 8 were used to classify Cloud and Shadow, bands 2, 8 and 10 were used to classify Cirrus and Water, and bands 1, 7 and 10 were used to classify Snow and Other. Additionally, the authors used a two-step approach to minimize human error: the labeled images were revisited to re-evaluate past decisions. Refer Subsection 2.1 to know more about Level-1C and Level-2A products.

Knowing Level-1C *product_id* and *coordinates* for individual pixels, we added surface reflectance information to each entry to build and assess a ML classification models and

also added Sen2Cor scene classification class for further comparison with the Sen2Cor algorithm. In this regard, the overall (input-label) dataset creation process is described in Figure 5.2, which perform the following steps:

1. For each *product_id* in [Hollstein et al. \(2016\)](#) dataset, Level-1C products were downloaded from CREODIAS platform¹;
2. For each downloaded Level-1C product, a corresponding Level-2A product was generated using *Sen2Cor v2.5.5*² at 20m. Afterwards, for each Level-2A product, Scene Classification (SCL) was retrieved to compare it with ML models for ML-Sen2Cor assessment; Here, SCL refers Sen2Cor class, refer Subsection 2.3.3 or Table 2.8;
3. Downloaded Level-1C products were re-sampled to 20m allowing spatial analysis and 13 bands of imagery were retrieved for the ML modeling.

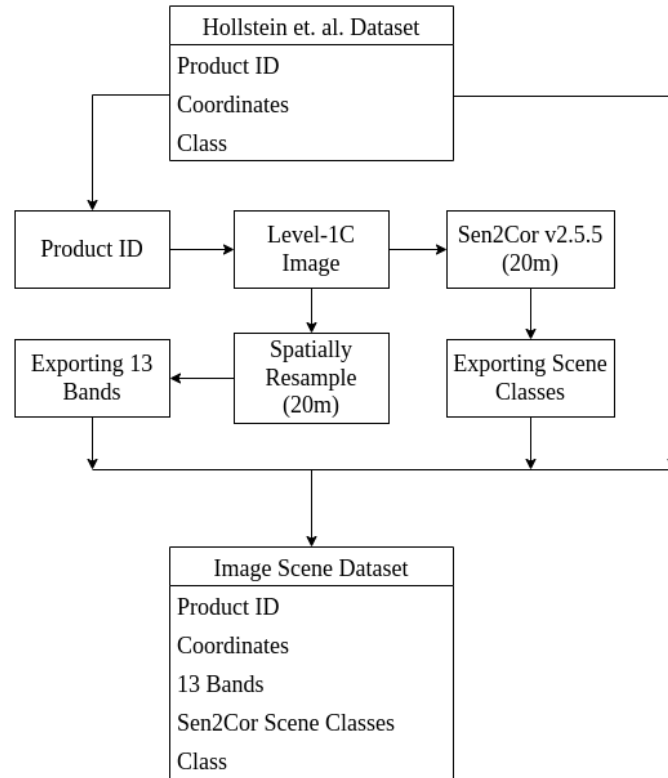


Figure 5.2: Image Scene dataset generation process.

As there are eleven classes in Sen2cor scene images (refer Table 2.8) and only six classes in [Hollstein et al. \(2016\)](#) dataset (refer Table 5.2) a class mapping was done as presented in Table 5.3. Thus, the **Image Scene dataset** is composed of **18 attributes**: *product_id*, *latitude*, *longitude*, *B01*, *B02*, *B03*, *B04*, *B05*, *B06*, *B07*, *B08*, *B8A*, *B09*, *B10*, *B11*, *B12*, *class*, *SCL*. Here, B refers the band; each band represents the surface reflectance (ρ) value at a different wavelengths.

¹CREODIAS platform is a cloud-based one-stop shop for all Copernicus satellite data and imagery, as well as the Copernicus services information (<https://creodias.eu/>).

²Sen2Cor v2.5.5 - http://step.esa.int/main/snap-supported-plugins/sen2cor/sen2cor_v2-5-5/

Table 5.3: Image Scene dataset - Class Mapping for Sen2Cor Assessment.

Mapped Class	Corresponding Sen2Cor Class
Cloud	Cloud high probability
Cirrus	Thin Cirrus
Snow	Snow
Shadow	Shadow, Cloud Shadow
Water	Water
Other	No Data, Defective Pixel, Vegetation, Soil, Cloud low and medium probability

Surface reflectance is defined as the fraction of incoming solar radiation that is reflected from Earth’s surface for a specific incident or viewing cases. So, in general, the reflectance values range from 0.0 to 1.0 and are stored in floating-point data format. Nonetheless, there are pixels with reflectance greater than 1.0; unlike negative reflectance, objects that have reflectance greater than 1.0 are not unnatural. Circumstances that would lead to the observance of reflectance greater than 1.0 are: nearby thunderstorm clouds that provide additional illumination from reflected solar radiation; the area receiving solar radiation is directly perpendicular to the sun; surfaces act as mirrors or lenses and reflect incoming direct sunlight in a concentrated way rather than diffusely, such as shiny buildings, waves, or ice crystals.

Figure 5.3 details the class-wise ρ value distribution using the violin plot. Here, we can observe that for each class, the ρ value for each band is different, meaning that each band has its ρ value according to a different type of surface/class. For example, for all classes, B10 ρ value is zero, apart from the Cirrus class; this is because B10 is responsible for the detection of thin cirrus ([European Space Agency, 2020](#)).

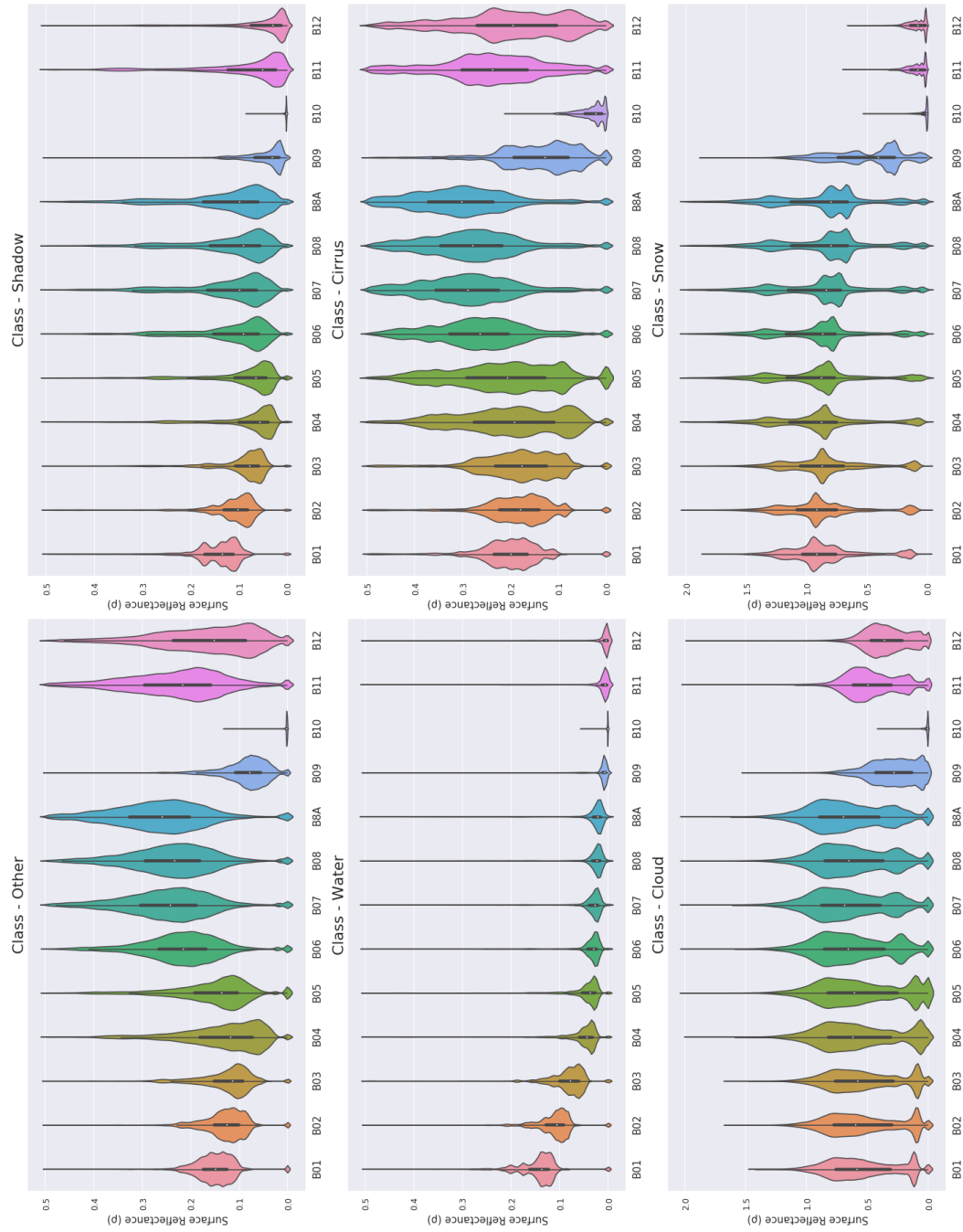


Figure 5.3: Class-wise band surface reflectance (ρ) value distribution.

Table 5.4 presents the number of points of the dataset per band and class with $\rho > 1$. We can observe that the class with a higher proportion of $\rho > 1$ values is the Snow and Cloud. This can be explained taking into account that snowy and cloud surfaces reflects incoming sunlight in a concentrated way rather than diffusely acting as a mirror, producing observed $\rho > 1$.

Table 5.4: Number of points with Surface Reflectance (ρ) greater than 1.0.

Bands\Class	Other	Water	Shadow	Cirrus	Cloud	Snow	Total
B01	47	229	1076	5013	34334	259562	300261
B02	587	313	2694	5589	47265	285053	341501
B03	190	289	1232	3742	40254	256421	302128
B04	536	429	3855	6897	79380	300538	391635
B05	516	447	4300	8099	84426	305134	402922
B06	546	477	4609	8597	993355	299270	1306854
B07	576	559	4653	8858	121825	290569	427040
B08	517	424	3942	7880	96182	277403	386348
B8A	607	597	4513	8903	133901	281007	429528
B09	0	0	0	6	3730	60674	64410
B10	0	0	0	0	0	0	0
B11	597	0	1	0	10112	0	10710
B12	43	0	0	0	671	0	714
Total	4762	3764	30875	63584	1645435	2615631	4364051

5.2 Waterbody Dataset

[Escobar \(2020\)](#) published a single class dataset of the water-body area consisting of 2.3 million (exactly 2,355,498) water-body points from 49 different Sentinel-2 products/images each one from a different country. By superposing band 4 (red), band 3 (green), and band 2 (blue), [Escobar \(2020\)](#) was able to reconstruct a true-color satellite image of the water bodies. To create masks of the water bodies, the author used the Normalized Difference Water Index (NDWI) ([Xu, 2006](#)). By using NDWI and a custom threshold higher than the one, the author was able to define a mask where white represents water, and black represents everything else but water. Figure 5.4 display one of the tagged images. The [Escobar \(2020\)](#) dataset is structured as *waterbody name, country name, geometry*. **Note:** here, published geometry only contained the area with water, resulting in a single class ‘Water’.

To generate waterbody dataset for the experiments, first, waterbody shapefiles were created using the published geometry. Then, the relevant Level-1C products were downloaded using the SentinelAPI³. Downloaded Level-1C images were then spatially resampled to 20m (to allow multispectral analysis) and 13 bands were retrieved. The overall process is described in Figure 5.5.

³SentinelAPI accepts a GeoDataFrame ([Mans, 2011](#)) of bounding boxes covering the shapefile; it then utilizes the Sentinelsat library (part of SentinelAPI ([European Space Agency, 2022j](#))), which creates a Python API, exploiting the Copernicus Open Access Hub ([Copernicus, 2022](#)) for direct download of the selected Sentinel-2 Level-1C images.

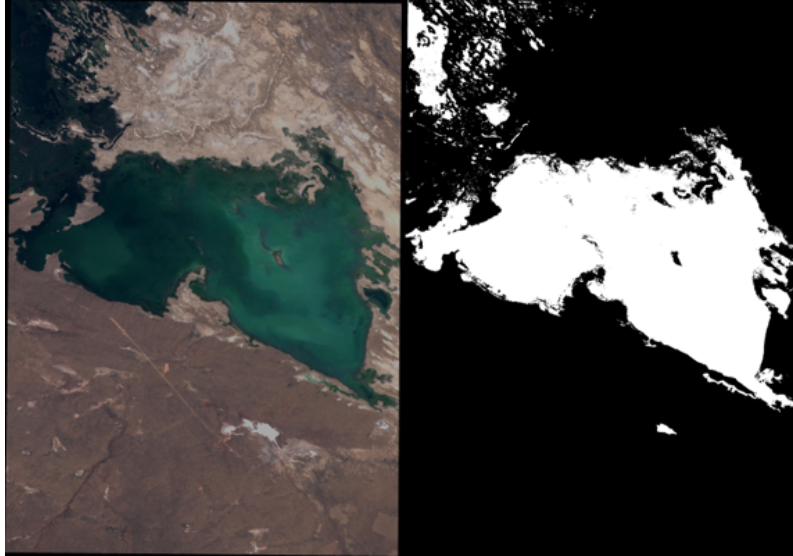


Figure 5.4: Waterbody in Kazakhstan (Escobar, 2020). Left: true color image, Right: mask.

Thus, the **Waterbody dataset** is composed of **17 attributes**: *product_id*, *latitude*, *longitude*, *B01*, *B02*, *B03*, *B04*, *B05*, *B06*, *B07*, *B08*, *B8A*, *B09*, *B10*, *B11*, *B12*, *class*. Here, B refers the band; each band represents the surface reflectance (ρ) value at a different wavelengths.

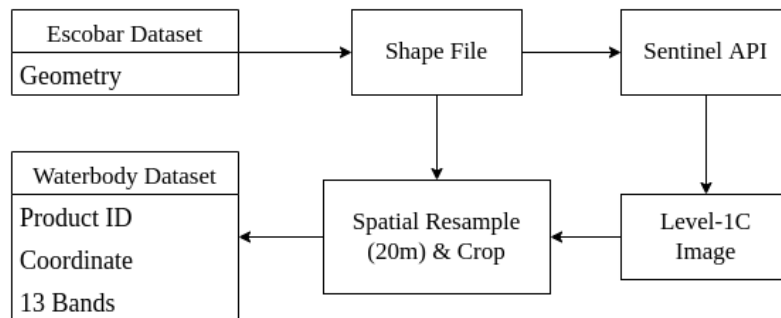


Figure 5.5: Waterbody dataset generation process.

5.3 Unlabeled Dataset

To have a better grasp of the experimental results presented in next chapters, we randomly selected two unseen Sentinel-2 images (from a pool of unlabeled images). Here, unseen means that these two images do not belong to any of the previously mentioned datasets. They are from two different geographical regions: (1) Fiji, Figure 5.6 and (2) Portugal, Figure 5.7.

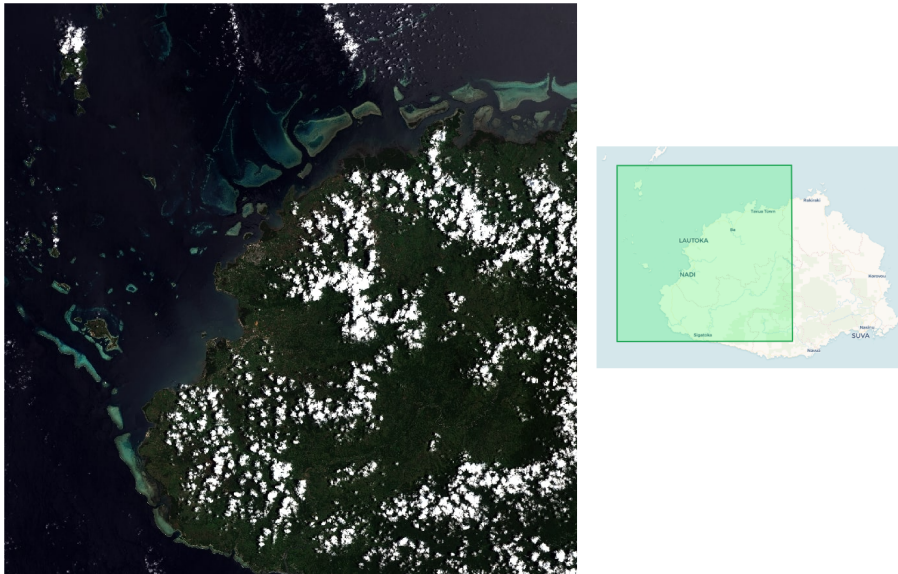


Figure 5.6: Fiji RGB image between $(17^{\circ}11'04'' \text{ S}, 176^{\circ}59'59'' \text{ E})$ and $(18^{\circ}10'26'' \text{ S}, 178^{\circ}02'16'' \text{ E})$ coordinates.

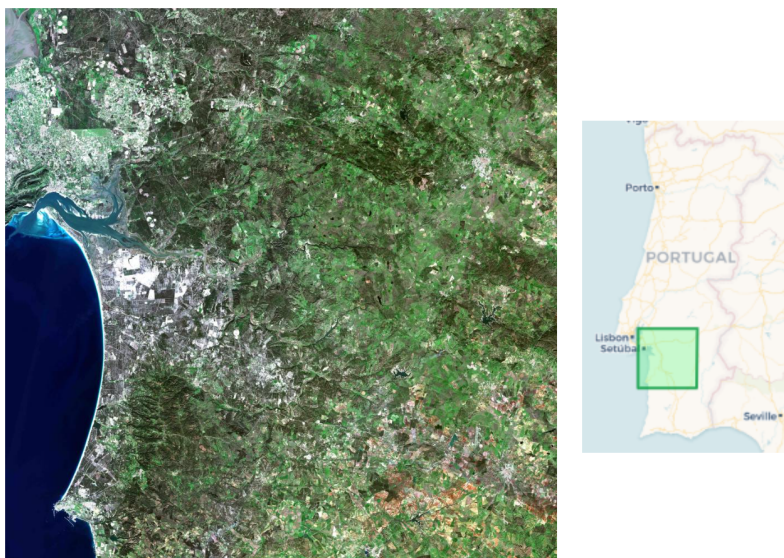


Figure 5.7: Portugal RGB image between $(38^{\circ}50'56'' \text{ N}, 9^{\circ}00'00'' \text{ W})$ and $(37^{\circ}51'10'' \text{ N}, 7^{\circ}45'07'' \text{ W})$ coordinates.

5.4 Summary

Chapter *Experimental Datasets* can be summarized as three datasets, [Hollstein et al. \(2016\)](#) with 6.6 million points divided into six classes (Cloud, Cirrus, Shadow, Snow, Water, Other), and [Escobar \(2020\)](#) with 2.3 million points with a Water class, were acquired; *Image Scene dataset* was built by extending [Hollstein et al. \(2016\)](#) dataset, adding 13 bands and the Sen2Cor class for each point/pixel; *Waterbody dataset* was built by extending [Escobar \(2020\)](#) dataset, adding 13 bands for each point/pixel; a collection of two Level-1C images as a dataset of *Unlabeled Images*.

Chapter 6

Image Scene Classification: Modeling and Results

“There are no such things as applied sciences, only applications of science.”

— Louis Pasteur

As mentioned in Chapter 2, a lot of researchers in the field of remote sensing have recently turned their attention to scene classification, or the segmentation of regions into morphological categories such as land, ocean, cloud (Mohajerani et al., 2018).

Further, given the continuous increase in the global population, the food manufacturers are advocated to either intensify the use of cropland or expand the farmland, making land cover and land usage dynamics mapping vital in the area of remote sensing. In this regard, identifying and classifying a high-resolution satellite imagery scene is a prime challenge. Several approaches have been proposed either by using static rule-based thresholds with limitation of diversity or neural network with data-dependent limitations.

Focusing on the problem of optical satellite image scene classification, this chapter proposes and evaluates ML models. Furthermore, the proposed ML models were tested against the Sen2Cor software used for calibrating and classifying Sentinel-2 images scenes.

Using different spectral and temporal resolutions satellite imagery, different CNN-based models (Li et al., 2019; Mohajerani et al., 2018; Zhang et al., 2019) were proposed to define cloud masks and land cover change. Further, Baetens et al. (2019) compared 32 reference cloud masks using Maccs-Atcor Joint Algorithm (MAJA) (Lonjou et al., 2016), Sen2Cor (Louis et al., 2016) and Function of Mask (FMask) (Qiu et al., 2019) respectively achieving 91%, 90%, and 84% accuracy. Apart from this, while multi spectra/temporal based methods achieve higher performance over cloud and land cover classification, they are complex and need multi spectra/temporal data during the learning phase, which is not always available. Besides, none of the previous studies emphasized the problem of detecting more than one class (Cloud, Cirrus, Shadow, Snow, Water, Other) using a single ML model. In this chapter, an inductive approach to learning from different surface reflectance is undertaken, which simplifies the inference stage of learning and improves the generalization ability of models.

The remainder of the chapter is organized as follows: Section 6.1 talks about the different machine learning algorithms used to develop the classifier models; Section 6.2 details the experimental setup and evaluation matrix used; Section 6.3 shows the results achieved; Section 6.4 exhibit in-depth discussion; Section 6.5 compares the use of spectral indices vs. Sentinel-2 raw bands for image scene classification; and finally, Section 6.6 demonstrates the practical use of classifier model leading to summary presented in Section 6.7.

6.1 Machine Learning Modeling

In this chapter, we evaluated ensemble methods, Random Forest & Extra Tree, distance based, K-Nearest Neighbors, and a deep learning based method, Convolutional Neural Network using the built image scene dataset for satellite imagery scene classification. This section introduces shortly the used classification algorithms.

Decision Tree (DT) Decision tree is an efficient inductive machine learning technique (Quinlan, 1996; Kuhn and Johnson, 2013) where the model is trained by recursively splitting the data (Pal and Mather, 2003). The data splitting consists of a tree structure root-nodes-branches-leaf, which successively tests features of the dataset at each node and splits the branches with different outcomes. This process continues until a leaf or terminal node representing a class is found. Each split is chosen according to an information criterion which is maximized or minimized by one of the ‘splitters’. Each node represents a feature in a classification category, and each subset specifies a value the node may accept. A Decision tree structure is shown in Figure 6.1.

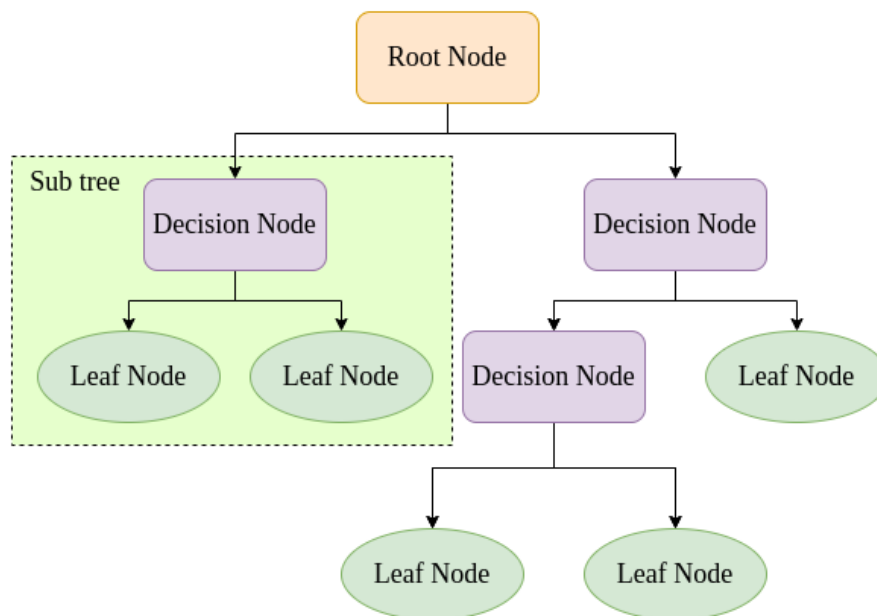


Figure 6.1: Decision tree structure (Charbuty and Abdulazeez, 2021).

A decision tree is sensitive to where it splits and how it splits. Generally, the bias-variance trade-off depends on the depth of the tree: a complex decision tree (e.g. deep) has a low bias and a high variance. Also, the tree makes almost no assumptions about the target

function but it is highly susceptible to variance in data making decision trees prone to overfit (Hastie et al., 2009).

Decision trees were used in many applications, including identification of land cover change (Al-Obeidat et al., 2015), mapping of global forest change (Hansen et al., 2013) and differentiating palustrine wetland types (Wright and Gallant, 2007).

Random Forest (RF) Random Forest (RF) is a tree ensemble algorithm (Bernard et al., 2009), which aims to reduce the decision tree variance at the small cost of bias. During the random forest learning process, bagging is used to minimize the variance by learning from several trees over various sub-samples of the data and/or a subset of the data-feature space (Belgiu and Drăguț, 2016).

Bagging is a method of averaging several decisions, and an RF classifier may be thought of as a collection of decision trees (Elith et al., 2008). The user needs to specify two parameters to initialize the RF algorithm (Hastie et al., 2009). These parameters are M and m which, respectively, represent the number of trees to be grown and the number of variables to be utilized to divide each node. First, N bootstrap samples are obtained from the training dataset's (some percentage e.g. first two-thirds) and to assess the accuracy of the predictions, the remaining of the training data (also known as out-of-bag (OOB) data) is employed. The optimal split among the predictor variables is then determined by growing an unpruned tree from each bootstrap sample, where each node has m predictors randomly picked as a subset (Akar and Güngör, 2012). Unpruned trees are larger trees with all nodes and branches present (Kwok and Carter, 1990). In the end, any observation is classified using all the individual trees, and the final decision is averaged.

Extra Tree (ET) The Extreme Random Tree (extra tree) and Random Forest algorithms are comparable because both use multiple decision trees, and as a result, both have many of the same benefits. Using them both, high-dimensional feature data may be handled successfully without the need for feature selection, and with high classification accuracy (Xia et al., 2015).

According to Azpiroz et al. (2021), the main difference between a random forest and extra tree lies in the fact that, the best splitting threshold or feature is not chosen when the extreme random tree divides at a node; instead, the splitting node is randomly chosen. This leads to fewer splitters and more diversified trees to evaluate when training (Geurts et al., 2006).

K-Nearest Neighbors (KNN) The KNN method assumes that related objects are located nearby to one another i.e. related items are close together in the data-feature space. Every data point that is close to another one is assumed to belong to the same class in KNN. To put it another way, it assigns a new data point a category based on similarities. These similarity can be distance, proximity, or closeness (Kramer, 2013).

According to KNN algorithms, the nearest neighbor of the categorization-required data point is denoted by the number k . If k is five, it will search for the five closest neighbors to that data point.

Convolutional Neural Networks (CNNs) CNNs are Neural Networks that receive an input, assign importance learnable weights and biases to various aspects/objects within the input, and are able to differentiate one input from other. Due to the sparse interactions and weight sharing, Neural Networks (NN) are best suited for processing large-scale imagery (Liu et al., 2018). When considering CNNs, the connection between the previous layer and the next layer is referred to as sparse interaction. Whereas, in weight sharing, layers share the same connection weights.

Recently researchers are proposing complex and deeper structures like, for example, AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan and Zisserman, 2014), and GoogLeNet (Szegedy et al., 2015), having depths of 8, 19, and 22, respectively (Lin et al., 2013a). In other words, CNN exploits domain knowledge about feature invariances within its structure and they have been successfully applied to various image analysis and recognition tasks (Abdel-Hamid et al., 2013), making it an effective technique for labeled tabular data classification (Brownlee, 2018).

6.2 Experimental Setup

From image scene dataset, ten products (one each from Asia and Oceania, two from Africa and America, and four from Europe) out of 60 were randomly chosen for the test set, while the remaining 50 were used to train the proposed classifier models. The train and test set distribution can be seen in Table 6.1.

Table 6.1: Train and Test sets: Class-wise Point Distribution (%).

Class	Train set		Test set		Whole Dataset	
	Points	Distribution	Points	Distribution	Points	Distribution
Cloud	897,504	86.98	134,315	13.02	1,031,819	15.57
Cirrus	780,635	81.60	175,988	18.40	956,623	14.43
Snow	728,012	82.47	154,751	17.53	882,763	13.32
Shadow	835,678	84.29	155,715	15.71	991,393	14.96
Water	954,416	89.08	117,010	10.92	1,071,426	16.16
Other	1,520,085	89.71	174,369	10.29	1,694,454	25.56
Total	5,716,330	86.24	912,148	13.76	6,628,478	100.00

To summarize, we use 50 Products (5,716,330 samples) for training, 10 Products (912,148 samples) for testing, 13 features, one band value for each sample as a feature, and six classes, Cloud, Cirrus, Shadow, Snow, Water, Other as a label. Python and Scikit-learn were used as programming languages and libraries, respectively.

Precision, recall and F1 score are performance measures that can be used to evaluate ML models. Precision is defined as the ratio between the number of correct positive and all positive results whereas, in recall all relevant samples (all samples that should have been identified as positive) are considered instead of all positive results; F1 is the harmonic mean of Precision and Recall. These measures are calculated per class considering one class as positive and all the other classes as negative using Equations 6.1.

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}, \quad F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (6.1)$$

Here, TP , TN , FP , and FN stand for *True Positive*, *True Negative*, *False Positive*, and *False Negative*. When aiming to have an unique performance value, precision, recall and F1 are averaged; this average can be calculated over the per class values (macro-average) or by summing the true positive, false positive and false negative for all classes and calculating the performance measures over these counts (micro-average).

To fine-tune the classification algorithms, a RandomizedSearchCV with 5 folds cross-validation procedure over the training set was used. The assessment was done using the micro-F1 measure over 200 iterations. Table 6.2 shows the best parameter values for Random Forests (RF) and Extra Trees (ET) algorithms. (`np.linspace`¹)

Table 6.2: Fine-tune Parameter values for Random Forests (RF) and Extra Trees (ET) Algorithms.

Parameter	RF	ET	Search Space
criterion	gini	gini	[gini, entropy, log_loss]
max_depth	20	20	np.linspace(start = 20, stop = 100, num = 20)
min_samples_split	50	10	[2, 5, 10, 20, 50]
min_samples_leaf	1	1	[1]
max_features	sqrt	sqrt	[auto, sqrt, log2]
n_estimators	242	279	np.linspace(start = 100, stop = 300, num = 50)
bootstrap	True	True	[True, False]

KNN parameters were fine-tuned in the same way as described previously (a Randomized-SearchCV with 5 folds cross-validation and assessment using the micro-F1 measure over 200 iterations) and are displayed in Table 6.3.

Table 6.3: Fine-tune Parameter values for KNN Algorithm.

Parameter	KNN	Search Space
n_neighbors	1	np.linspace(start = 1, stop = 10, num = 10)
leaf_size	2	np.linspace(start = 2, stop = 30, num = 10)
p	2	[1, 2]
weights	uniform	[uniform, distance]
algorithm	auto	[auto, ball_tree, kd_tree, brute]

Using the CNN architecture (LeCun et al., 1989) as a base reference and adopting CNN architecture presented by Brownlee (2018), we proposed a CNN model consists of an input layer, 1D convolutional layers, a dropout layer, a max-pooling layer, followed by one flatten, two dense, and an output layer. The CNN parameter values $epochs = 8$, $batchsize = 32$, $filters = 24$, and $kernelsize = 4$ were used during the experiment. Figure 6.2 shows the CNN model structure.

¹numpy.linspace - returns evenly spaced numbers (num) over a specified (start,stop) interval. <https://numpy.org/doc/stable/reference/generated/numpy.linspace.html>

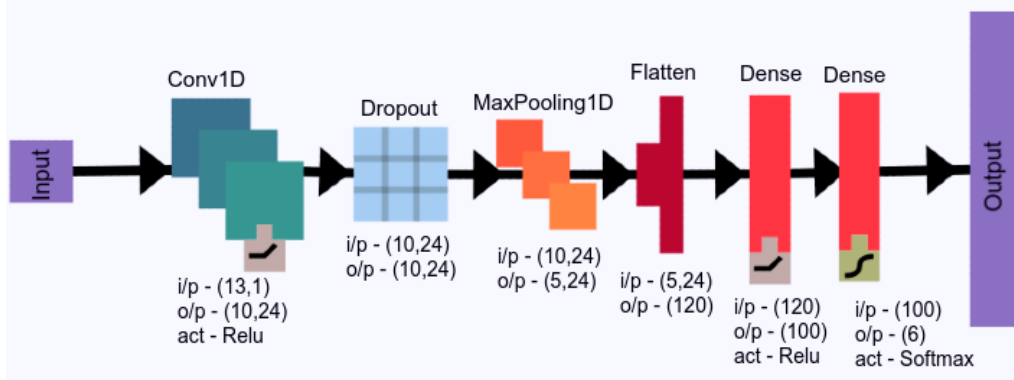


Figure 6.2: Proposed Convolutional Neural Network (CNN) architecture (Raiyani et al., 2021).

Following is a description of each layer:

- Input layer: The input representation of this layer is a matrix value of 13 bands;
- Convolutional-1D layer: This layer is used to extract features from input. Here, from the previous layer, multiple activation feature maps are extracted by combining the convolution kernel. In our architecture, we used a convolution kernel size of 4;
- Dropout: A random portion of the outputs for each batch is nullified to avoid strong dependencies between portions of adjacent layers;
- Pooling layer: This layer is responsible for the reduction of dimension and abstraction of the features by combining the feature maps. Thus, the overfitting problem is prevented, and at the same time, computation speed is increased;
- Flatten layer: Here, the (5×24) input from the previous layer is taken and transformed into a single vector giving a feature space of width 120;
- Dense layer: In this layer, each neuron receives input from all the neurons in the previous layer, making it a densely connected neural network layer. The layer has a weight matrix W , a bias vector b , and the activation function of the previous layer.
- Softmax activation: It is a normalized exponential function which is used in multinomial logistic regression. By using the softmax activation function, the last output vector of the CNN model is forced to be a part of the sample class (in our case, the output vector is 6).

6.3 Classification Results

As mentioned in the previous section, the classification models were assessed over the test set that comprises 10 products from all five continents. Table 6.4 shows precision, recall and Table 6.5 shows micro-F1 values for Random Forests (RF), Extra Trees (ET), K-Nearest Neighbors (KNN), Convolutional Neural Networks (CNN) along with Sen2Cor Scene Classification (SCL).

Table 6.4: Precision and Recall Results over the test set: Random Forest (RF), Extra Trees (ET), K-Nearest Neighbors (KNN), Convolutional Neural Network (CNN) and Sen2Cor (SCL).

Class	Precision					Recall					Support
	RF	ET	KNN	CNN	SCL	RF	ET	KNN	CNN	SCL	
Cloud	77	81	71	79	62	91	90	83	90	94	134315
Cirrus	78	82	68	78	91	67	76	56	75	10	175988
Shadow	89	91	70	91	96	77	73	67	75	54	155715
Snow	93	94	81	96	86	88	86	87	86	31	154751
Water	96	93	84	93	84	86	87	83	87	83	117010
Other	74	74	67	74	39	91	96	69	92	97	174369
Overall	83	84	73	84	59	83	84	73	84	59	912148

Table 6.5: Micro-F1 Results over the test set: Random Forest (RF), Extra Trees (ET), K-Nearest Neighbors (KNN), Convolutional Neural Network (CNN) and Sen2Cor (SCL).

Class	RF	ET	KNN	CNN	SCL	Support
Cloud	83	86	76	84	75	134315
Cirrus	72	79	61	76	18	175988
Shadow	83	81	68	83	69	155715
Snow	90	90	84	91	46	154751
Water	91	90	84	90	84	117010
Other	82	83	68	82	56	174369
Overall	83	84	73	84	59	912148

Analysing Table 6.4 and Table 6.5, the following observations can be made:

Looking at micro-F1, CNN performs similar to Random Forest and Extra Trees. The difference in micro-F1 is small (almost zero) and we cannot state that CNN outperforms the others. Moreover, one can state that each algorithm performs better than the others on specific classes; for example, ET has higher micro-F1 over classes Cirrus, Cloud, and Other, whereas RF has higher micro-F1 over Water and CNN over Snow.

Looking at precision and recall for Cirrus and Shadow classes, it is noticeable that Sen2Cor has high precision but low recall. This means that Sen2Cor is returning very few results of Cirrus and Shadow, although most of its predicted labels are correct.

Overall, the four Machine Learning algorithms generate models with similar performance with differences that range from 0% to 18% between the “best” and the “worst” for specific class. For example, Cirrus has a “best” micro-F1 of 79% with ET and a “worst” micro-F1 of 61% with KNN.

Overall, with regard to the classes, there is a great variation: precision values are above 90% for classes Snow and Shadow and less than 75% for the Other class; for recall, the highest values are obtained for the classes Cloud and Other (values above 80%) and the lowest for the Cirrus and Shadow classes (values between 67% and 77%). Regarding the micro-F1 measure, (except KNN) the only class with values below 80% is the class Cirrus; classes Snow and Water have values above 90%.

Comparing the performance of ML algorithms with Sen2Cor, especially for the Cirrus and Snow classes, ML approaches are superior. For these classes, Sen2Cor micro-F1 values are below 50%; these low values are due to the big difference between precision and recall, for Cirrus precision is above 90% while recall is 10%; for Snow precision is above 85% and recall around 30%. Considering the micro-F1 measure, the ML models present an increase of about 25 points from 59% to 84% when compared to the Sen2Cor Scene Classification algorithm.

To check if there is a significant difference between Sen2Cor and ML models i.e., if the difference in micro-F1 is significant or not, the McNemar-Bowker test (McNemar, 1947; Bowker, 1948) was performed.

The McNemar-Bowker’s test is a statistical test used on paired nominal data with $k \times k$ contingency tables following a dichotomous trait to determine if there is a difference between two related groups. Here, k is the number of categories/labels, and the McNemar-Bowker (B) value is calculated using the Equation 6.2, where, $O_{i,j}$ is the count of the i^{th} row and j^{th} column in the crosstab. A crosstab is a table that shows the relationship between $k \times k$ variables.

$$B = \sum_{i=1}^{k-1} \sum_{j=i+1}^k \frac{(O_{i,j} - O_{j,i})^2}{(O_{i,j} + O_{j,i})} \quad (6.2)$$

The acquired B value follows approximately a chi-square distribution (Lancaster and Seneta, 2005), with $df = (k - 1)/2$ degrees of freedom. The probability density function is calculated using Equation 6.3, where, Γ denotes the gamma function, which has closed-form values for integer $(df/2)$.

$$f(B, df) = \frac{B^{df/2-1} e^{-B/2}}{2^{df/2} \Gamma(df/2)} \quad (6.3)$$

Using Equation 6.3 for comparing Sen2Cor and ML models, our null hypothesis, the difference between two groups is statistically significant was proved by obtaining a (p -value) less-than 0.05.

We randomly picked a image, Figure 6.3 from the test set with the classifications Cloud, Shadow, and Other, described as white, brown, and green, respectively and created classification images using the ET model, Figure 6.4 and Sen2Cor, Figure 6.5 to comprehend the distinctions between the ML model and Sen2Cor.

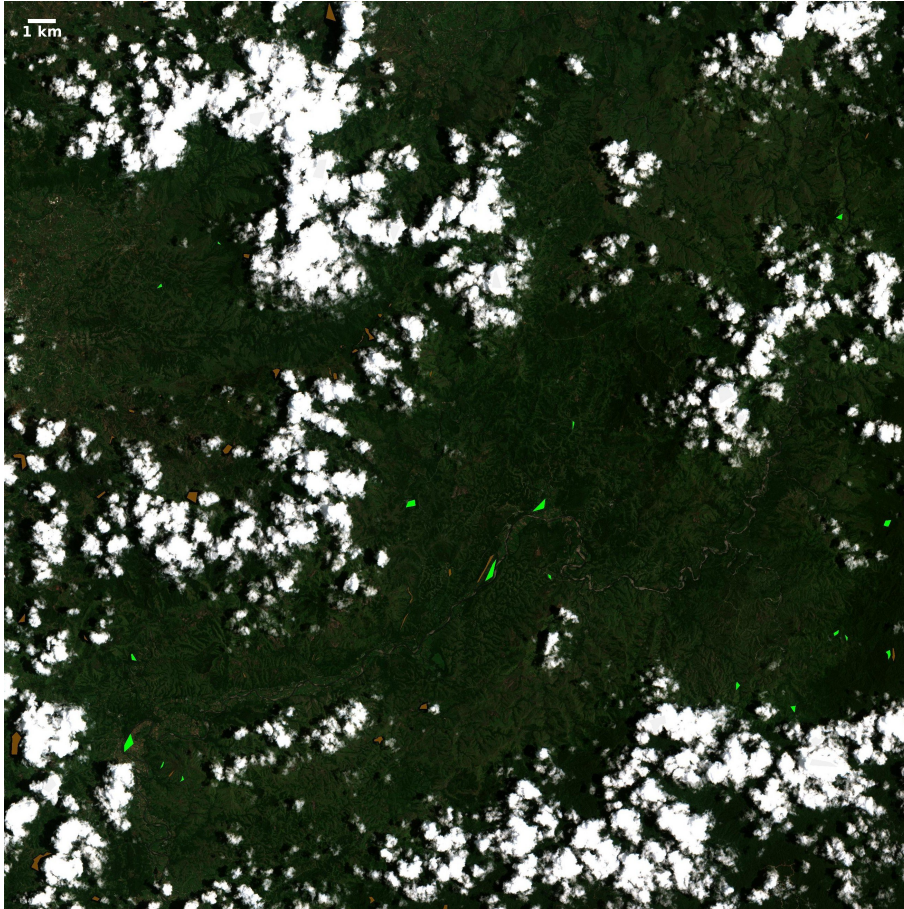


Figure 6.3: RGB image of Lautoka Area, Fiji between (17°42'58'' E , 177°35'46'' S) and (18°03'24'' E, 177°54'01'' S) coordinates.

After analysing Figure 6.3 closely, it is possible to say that for each cloud present in the image there is an equivalent shadow. This is not true in Figure 6.5, concluding that the ML model is classifying cloud and cloud shadow more accurately than Sen2Cor; Sen2Cor classification is missing the majority of cloud shadows, whereas the ML model captures them all, Figure 6.4.

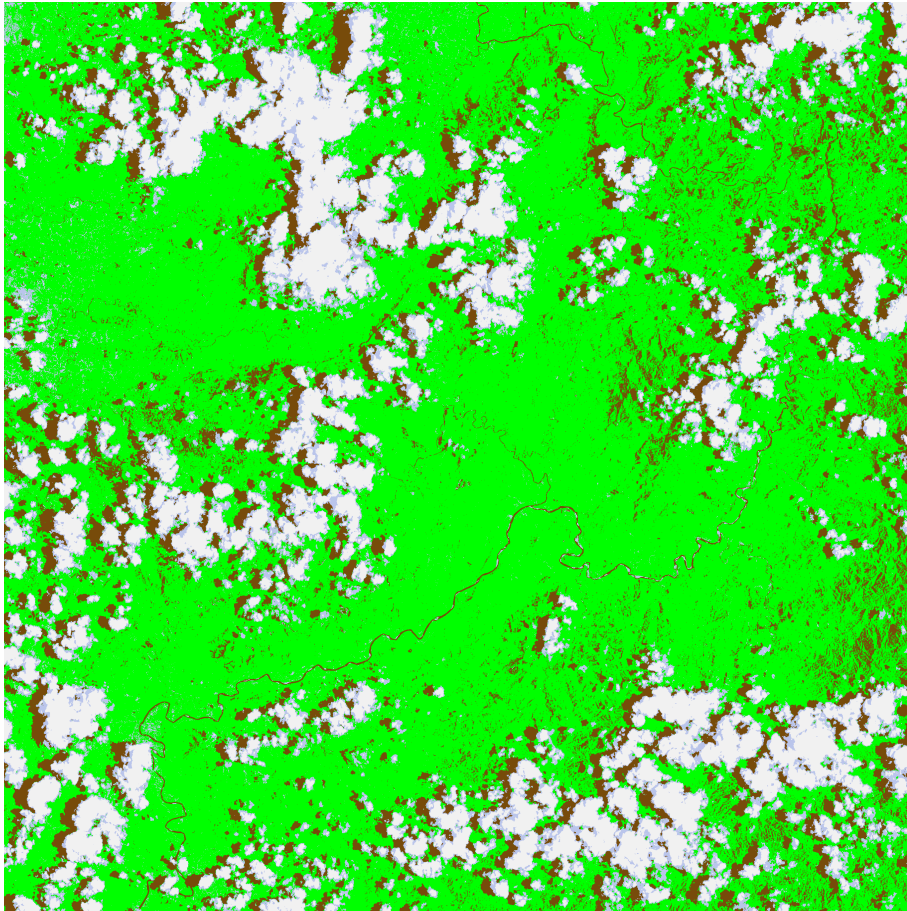


Figure 6.4: Extra Tree classified image of Lautoka Area, Fiji. Color Labels: Cloud (White), Shadow (Brown), Other (Green).

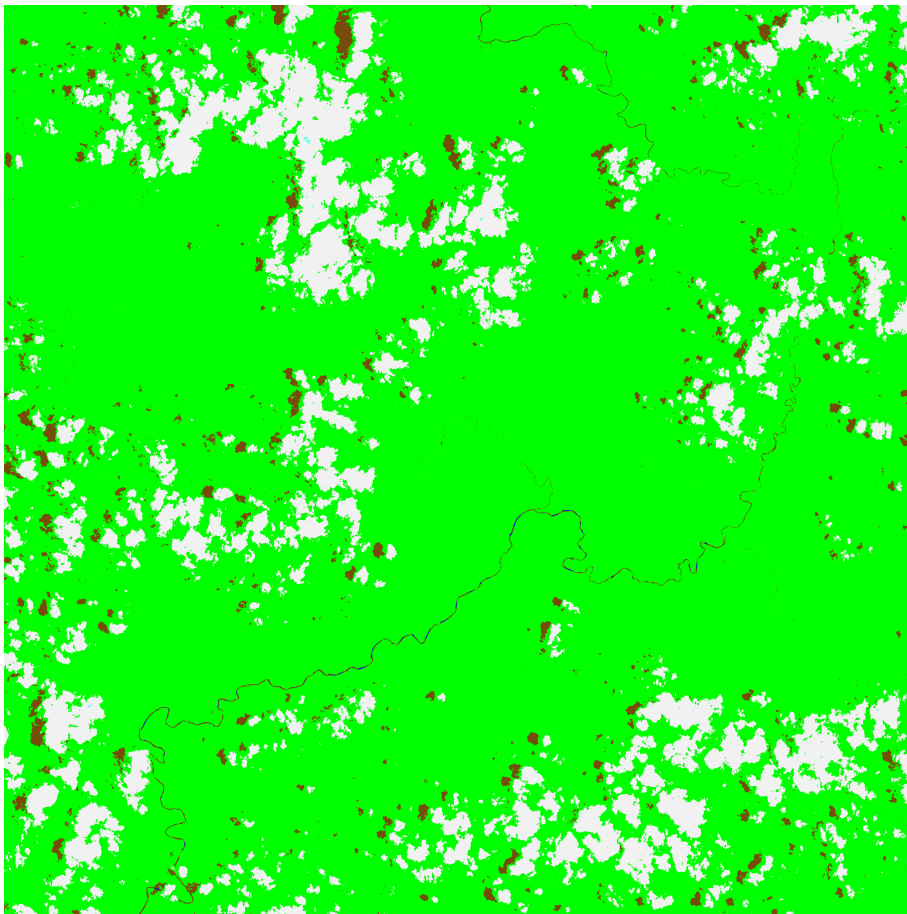


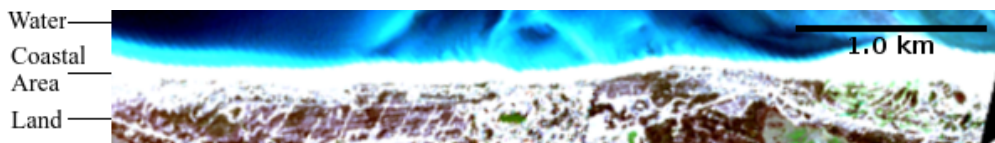
Figure 6.5: Sen2Cor classified image of Lautoka Area, Fiji. Color Labels: Cloud (White), Shadow (Brown), Other (Green).

6.4 Discussion

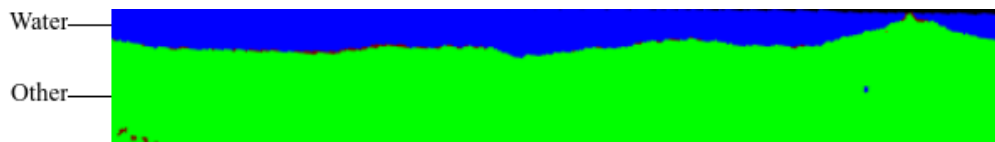
This section discusses specific elements of the modeled ML image scene classification algorithms and equivalent assessment with Sen2Cor.

Surface Reflectance and Marginal Variation Assessment: Marginal variance, for example, can be referred to as a bright cloud reflectance vs white sand reflectance, where the ρ value of bands is larger due to the characteristics of the surface.

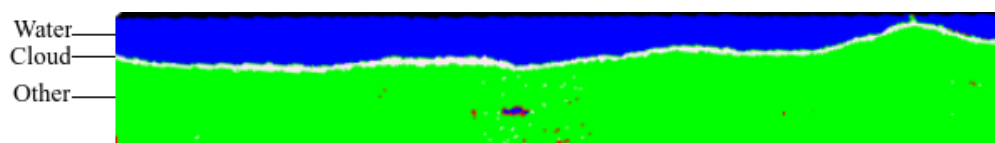
We state that a static rule-based approach, like the one used by Sen2Cor that heavily rely on surface reflectance, it uses a sensor-specific threshold based method that tends to miss marginal variation between two surfaces leading to misclassification. To support this claim and to prove the general-ability of the ML model, we used a specific image, Figure 6.6 with brighter surface reflectance values (note that this image does not belong to the dataset): Figure 6.6a shows 3 visible parts: water, coastal area sand and land surface; Figure 6.6b shows the generated classification images using the ET model and Figure 6.6c shows the Sen2Cor output. After analysing Figure 6.6 closely, it is possible to say that the bright coastal area/sand present in the Figure 6.6a is classified as Cloud by Sen2Cor represented as a white line but, the ML algorithm classifies it as Other which is indeed is the correct classification. This enables us to conclude that, the ML model is able to better capture marginal variation in surface reflectance values when compared to Sen2Cor.



(a) Coastal Area RGB Image.



(b) ML Classification.



(c) Sen2Cor Classification.

Figure 6.6: A Coastal Area Image (Lisbon, Portugal, between $(38^{\circ}29'28'' \text{ N}, 8^{\circ}55' \text{ W})$ and $(38^{\circ}26'11'' \text{ N}, 8^{\circ}49'18'' \text{ W})$) with brighter surface reflectance. Color Labels: Water (Blue), Cloud (White), Other (Green)

Pedantic Assessment of ML Model: A pedantic model in machine learning is one that is highly sensitive to a smallest feature changes; it is overscrupulous (Raiyani et al., 2021).

Since Figure 6.4 shows shadows of all the clouds, we analyzed the sensitivity of the ML model. For that we randomly selected (not from dataset) three separate L1C image patches and applied the ET classifier to check if the classifier was too “pedantic” to the region or not.

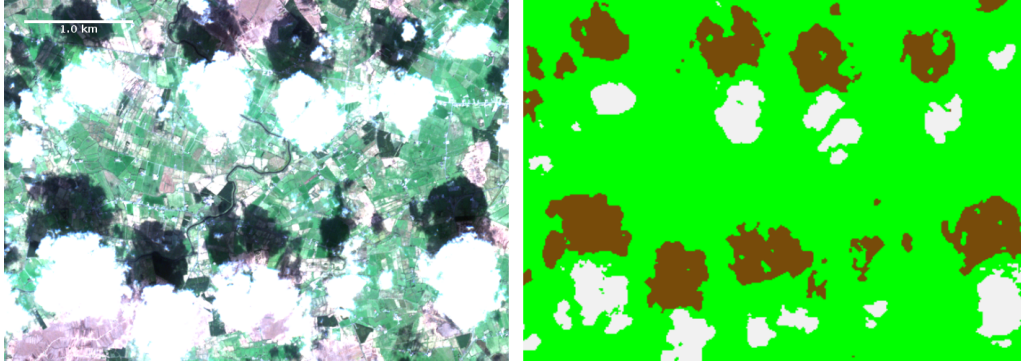


Figure 6.7: Ballyhaunis, Ireland, area between ($54^{\circ}04'02''$ N , $8^{\circ}50'03''$ W) and ($54^{\circ}01'$ N, $8^{\circ}44'44''$ W) coordinates.

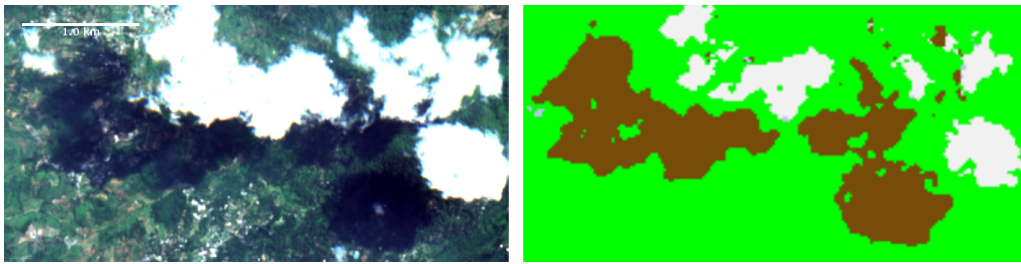


Figure 6.8: Sukabumi, Indonesia, area between ($6^{\circ}37'13''$ S , $106^{\circ}53'43''$ E) and ($6^{\circ}38'22''$ S, $106^{\circ}55'55''$ E) coordinates.

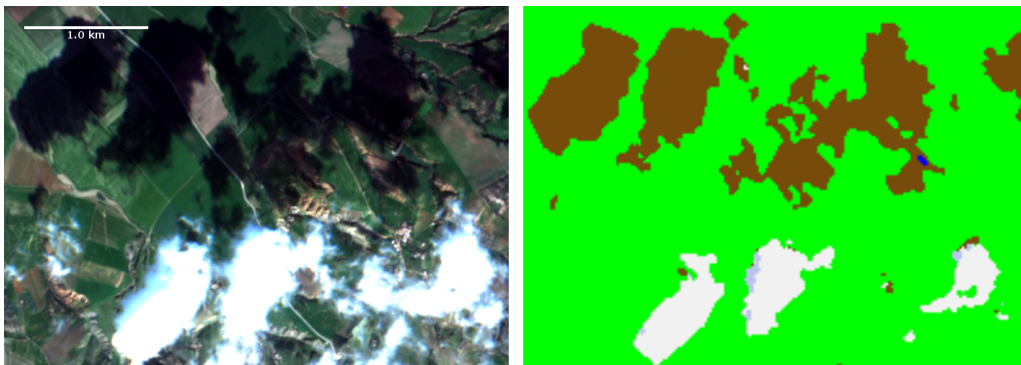


Figure 6.9: Béja, Tunisia, area between ($36^{\circ}57'45''$ N , $9^{\circ}45'21''$ E) and ($36^{\circ}57'45''$ N, $9^{\circ}48'08''$ E) coordinates.

Figure 6.7 to 6.9 presents the geometric independent region, Ballyhaunis, Sukabumi, and Béja images and the corresponding classified images. It can be observed that for each image, the ML model is capturing, with high precision, the shadows of the (low, medium, and opaque) clouds, proving the general-ability of the ML model. To this extent, we can say that the ML models are sensitive and can detect even minor shadows from low and medium probability clouds.

Here, it's important to point out that this test of model sensitivity was limited to the classes Cloud and Shadow.

Separate from the sensitivity analysis and observing the model behavior across all figures, the detection of shadow does not reduce the usable area since the classifier is constructing (near to perfect) a mask. Here, the usable area means that the classifier is not stretching the mask to an area where classes do not exist.

ML Model Biasness Assessment: According to [Mehrabi et al. \(2021\)](#), machine learning algorithms can be biased towards the data they were trained and cannot perform as significant over new data.

Therefore, we studied the biasness of the model towards the achieved results, posing a question, 'will the ML model be able to classify a new, non seen product with high performance?'

To answer above question: from image scene dataset, we selected 59 products for training and 1 for testing. The main reason to split the dataset in this way was to make sure that the knowledge about a region is not essential to classify that region. Meaning, it would be interesting to pick a complete region as test while the rest of the points compose the training set; we replicated this procedure for each of the 60 products i.e., using 1 product for test and the rest 59 products for training; the $F1_{avg}$ results are presented in Table 6.6.

Table 6.6: Scene Biasness Test Results: $F1_{avg}$ values of ML algorithms and Sen2Cor.

Class	DT	RF	ET	CNN	Sen2Cor	Support
Other	63.29	72.30	74.16	74.43	64.96	1,694,454 (25.56%)
Water	63.81	73.40	76.69	73.88	80.73	1,071,426 (16.16%)
Shadow	53.98	63.96	61.45	64.63	50.57	991,393 (14.96%)
Cirrus	47.58	56.63	42.97	51.58	24.08	956,623 (14.43%)
Cloud	65.25	75.08	75.33	72.67	75.04	1,031,819 (15.57%)
Snow	74.67	84.90	87.00	83.43	61.40	882,763 (13.32%)
$F1_{avg}$	67.95	76.43	76.77	77.54	66.40	6,628,478 (100%)

Equation 6.4 calculates the $F1_{avg}$ value over 60 products for each class where $F1_{p_k}$ is the $F1$ value of the particular class k within the product p . N_p is the number of points of the class within the product p , T is the total number of points of the class for all products, and $p \in (1, 60)$ is the number of products.

$$F1_{avg} = \sum_{p=1}^{60} \frac{(F1_{p_k} \times N_p)}{T} \quad \text{with} \quad T = \sum_{p=1}^{60} N_p \quad (6.4)$$

When compared to the Sen2Cor, an ML algorithm achieved an overall improvement of 11%. This study ensures that the high achieved results are due to the learning done by the algorithms, and that the proposed ML models performance is not biased to the test set.

Tree based Model: Tree methods uses information like the Gini index to define the splits of tree (which might be a useful insight to the end-user). The Gini index is a measure

of statistical distribution intended to represent different attribute variables influencing the overall accuracy (Belgiu and Drăguț, 2016). Using the Gini index, we were able to identify that B11 and B12 have a substantial effect on the overall model accuracy.

The random forest and extra tree algorithms produce fast and accurate predictions (with micro-F1 between 83% and 84%). Nonetheless, when using these ensemble methods, the issues of overfitting, and bias/variance tradeoff should not be overlooked.

CNN Model: Regarding the neural network architecture, different CNN-based models were proposed to classify cloud mask and land cover change using different spectral and temporal resolutions satellite imagery (Li et al., 2019; Mohajerani et al., 2018; Zhang et al., 2019). These studies look at different datasets and present different CNN architectures but, to the best of our knowledge, none evaluates the CNN architecture with the dataset used in this work making it impossible to make a comparison of the obtained results.

6.5 Image Scene Classification: Bands or Spectral Indices

It is evident from the preceding subsections that the presented machine learning models outperform Sen2Cor. Although Sen2Cor employs spectral indices to identify some of the classes (refer subsection 2.3.3), this presents a question, “Should spectral indices be utilized instead of 13 bands as a feature during ML learning?” Through experimentation, we attempted to provide a response in this section.

The integrated use of satellite and ground-based observations is widely recognized as the most feasible approach for the measurement and long-term monitoring of terrestrial variables needed by scientific investigators and decision-makers around the world. In particular, Earth observation applications are making use of the unique, synoptic capabilities of an ever-increasing number of satellite remote sensing imaging systems. A key challenge is to ensure that such measurements yield self-consistent and accurate geophysical and biophysical data over time and space, even though the measurements are made with a variety of different sensors under different observational conditions.

According to a study by the International Centre for Integrated Mountain Development (ICIMOD) (Bhandari, 2012), band ratios are used to remove undesirable effects on recorded radiances (e.g. variable illumination) since topographic slope and aspect, shadows or seasonal changes can cause differences in brightness values between identical surface materials. As a result, the interpreter’s ability to correctly identify surface material in an image is hampered. The band ratio transformations can be used to mitigate these effects. Aside from that, the Spectral Indices such as Leaf Area and Cumulative Diversity indices have been widely used to model, predict, and track land change processes (Roy et al., 2015).

In the last two decades (1999 to 2019), Polykretis et al. (2020) examined the impact of various spectral indices in detecting land cover changes on the Greek island of Crete. According to Dixit et al. (2019), the visible, NIR, and SWIR bands are the most commonly used reflectance and absorptive properties for developing snow/ice cover mapping; based on these, they proposed the Snow Water Index (SWI) with an overall accuracy of 93%. Separately, according to Zhai et al. (2018), the majority of existing cloud/shadow detection methods are based on visible and infrared spectral band configurations with working mechanisms relatively complex and computationally complicated; as such, they proposed an unified cloud/shadow detection algorithm based on spectral indices with a cloud detec-

tion accuracy of 98% and a cloud shadow detection accuracy of 84%.

Referring to all the previous work and approaches, this section reports simulation study and encapsulating a result between ‘Classification using surface-specific spectral indices and Sentinel-2 raw bands’.

Spectral indices are functions (usually, ratios) of the pixel values from two or more spectral bands in a multispectral image. Spectral indices are designed to highlight pixels showing the relative abundance or lack of a land-cover type of interest in an image. From the indices presented in the literature, a subset was chosen specifically to identify each of the specific six classes. These are enlisted in Table 6.7 and Appendix A.1 presents the individual spectral indices formulas.

Table 6.7: Classes and Used Spectral Indices.

Class	Spectral Indices
Water	Normalized Difference Water Index (NDWI) Sentinel-2 Water Index (SWI)
Shadow	Shadow Enhancement Index (SEI) Saturation Value Different Index (SVDI)
Cloud	Cloud Index (CI) Brightness Index (BI)
Cirrus	Sentinel-2 Band 10
Snow	Normalized Difference Snow Index (NDSI) Normalized Difference Snow Ice Index (NDSII) S3 Snow Water Index (SWI)
Other	Bare Soil Index (BSI)

We use the same experimental setup and matrix from Section 6.2 while conducting experiments. We also used micro-F1 as an evaluation metric; 50 products for training and 10 for testing; and spectral indices presented in Table 6.7 as features rather than the 13 bands.

Table 6.8 presents the classification results using only the spectral indices and together with the 13 bands. All the 3 ML models give similar results, having higher F1 values for Water (90%) and lower F1 values for Shadow (75%). Adding the 13 bands values to the spectral indices does not seem to improve the results. Moreover, by comparing these results with the ones obtained using the 13 Bands, it is possible to conclude that the use of indices does not improve the classifier.

The highest difference in model performance (in percentage) across any two classes is 15, 13, and 21 for RF, ET, and DT, respectively, when looking at individual F1 values (from 13 Bands + Spectral Indices findings). For instance, in RF, Water has a ‘best’ micro-F1 of 89%, while Cirrus has a ‘worst’ micro-F1 of 74%, resulting in a maximum model performance differential of 15%.

By analyzing the indices presented in Table 6.7, one notices that the spectral indices only use information from 10 bands (not included bands: 6/7/8A) of the available 13 bands of

Table 6.8: micro-F1 with Spectral Indices and 13 Bands.

Class	Spectral Indices			13 Bands + Spectral Indices		
	RF	ET	DT	RF	ET	DT
Water	90	90	82	89	89	81
Shadow	75	75	62	76	76	66
Cloud	81	82	70	80	81	71
Cirrus	78	79	63	74	76	62
Snow	87	87	81	88	87	83
Other	79	80	66	81	81	71
micro-F1	81	82	70	81	81	72

Sentinel-2. Having this in mind, classifiers were built using raw information from those 10 bands only. The obtained results are presented in Table 6.9 and show that there is no significant difference on classifiers performance (1% more for Random Forest and Extra Tress when compared to the results of Table 6.8). Thus, we can definitively conclude that there is no need to calculate and use spectral indices instead of raw bands for Sentinel-2 Image Scene Classification (at least for studied six classes: Cloud, Cirrus, Shadow, Snow, Water, and Other.)

The highest difference in model performance (in percentage) across any two classes is 15, 15, and 20 for RF, ET, and DT, respectively, when looking at individual F1 values from Table 6.8 findings. Additionally, for a single class ‘Water and Cirrus’, both approaches, raw bands and Spectral Indices exhibits similar performance with a maximum F1 score of approximately 89-90 and a minimum value of around 62-63.

Table 6.9: micro-F1 using 10 Bands (not included bands - 6/7/8A).

Class	RF	ET	DT
Water	89	89	81
Shadow	76	76	66
Cloud	80	81	71
Cirrus	74	76	62
Snow	88	87	83
Other	81	81	71
micro-F1	81	81	72

Through our experiments, we were able to provide a experimental study that shows that raw bands of Sentinel-2 can be used as features instead of using different Spectral Indices. This can be verified from the results presented on Tables 6.8 and 6.9. Moreover, when ‘13 bands + spectral indices’ are used together no improvement is verified (Table 6.8) compared to results obtained using only 13 bands (refer micro-F1 results presented in Table 6.5 of Section 6.3).

6.6 Atmospheric Disturbance Identification

We have demonstrated, using an image scene dataset, that the ML model performs better than Sen2Cor for Sentinel-2 Image Scene Classification when 13 bands are used as features. However, it would be reassuring if the same outcomes were found when the proposed ML model is applied to a real-world issue, such as ‘‘Atmospheric Disturbance Identification’’. Atmospheric disturbance can be defined as ‘no disturbance’, image with clear-sky or ‘with disturbance’, image with cloud, shadow, snow, and water coverage.

Agroinsider (Agroinsider, 2022), a firm that offers solutions for agricultural and environmental sustainability, served as our key source of inspiration for undertaking this assignment. They use the NDVI value to determine whether an atmospheric disturbance is present, however, we suggested using the provided ML model to get better outcomes.

To experimentally prove, with the help of Agroinsider, we acquired 170 (5 days apart) Sentinel-2 images from 05-01-2017 to 03-08-2019 of ten corn parcels from Alentejo region, Portugal. Figure 6.10 shows the corresponding 2D image of the ten corn parcels (referred as parcel-1 to parcel-10 onwards).

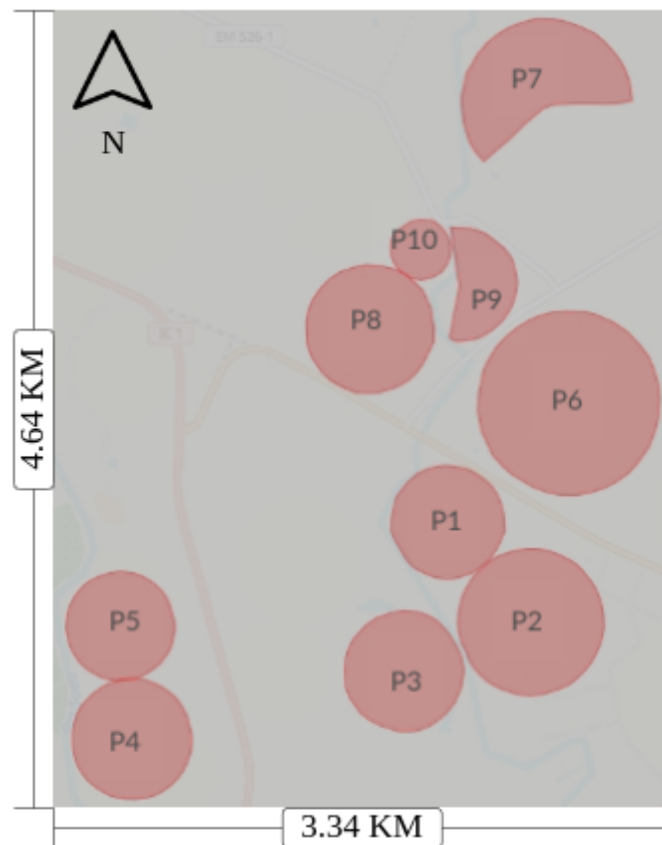


Figure 6.10: Ten corn parcels from Alentejo Region, Portugal between $(37^{\circ}56'29.13''$ N, $8^{\circ}22'21.95''$ W) and $(37^{\circ}55'32.44''$ N, $8^{\circ}21'02.23''$ W) coordinates.

Figure 6.11 shows the mean NDVI value from 05-01-2017 to 03-08-2019 for parcel-1². In it, the presence of atmospheric disturbance can be observed as sudden dips in the NDVI

²The same can be replicated to rest of parcels.

values, supported by the fact that it is not possible to lose crop growth and regain it within a range of 5 days (the observation cycle time).

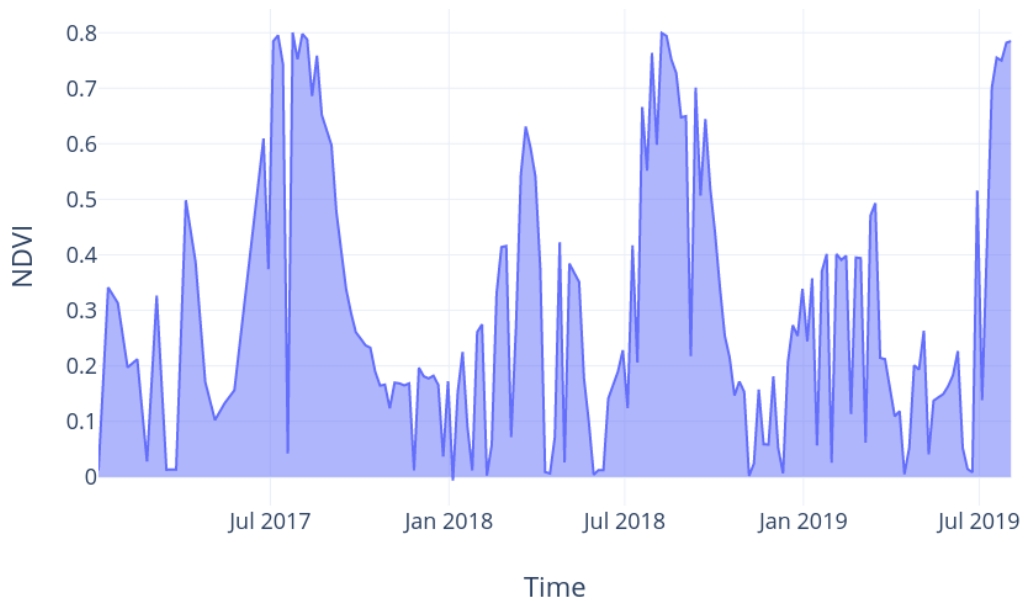


Figure 6.11: Mean NDVI value for parcel-1 from 05-01-2017 to 03-08-2019.

Now, using the developed Extra Tree model (refer Section 6.1), the new, unseen optical images (with 13 bands) of the ten parcels were classified as ‘no atmospheric disturbance image’, clear-sky or ‘image with disturbance’, cloud, shadow, snow, and water coverage. Here, each point within the parcel was classified as either 0 if point was classified as clear sky and 1 when it was classified as atmospheric disturbance. Figure 6.12 presents the calculated disturbance over dates 14-06-2017 to 01-12-2017, with red line for the ET model and blue line mean NDVI. These results sync with sudden dips of the NDVI values supporting the claim of the presence of atmospheric disturbance in the optical image.

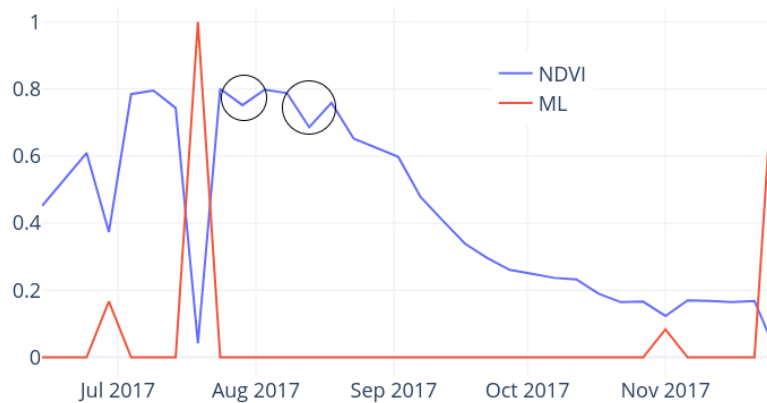


Figure 6.12: Parcel-1: Mean NDVI and atmospheric disturbance identification by ML (over dates 14-06-2017 to 01-12-2017).

Figure 6.12 demonstrates that although the mean NDVI ranged from 0.78 to 0.68 (a dip, show in black circles) to 0.76 on 08, 13, and 18 Aug’17, the value of atmospheric disturbance remained at 0.0. Leading us to make a statement that the NDVI value is not sufficient enough to locate disturbance. In conclusion, the ML model provided in this chapter may

be applied with success to resolve real-world problems like “Atmospheric Disturbance Identification.”

6.7 Summary

A summary of the Chapter *Image Scene Classification: Modeling and Results* is provided below:

Presents modeling of four different ML algorithms namely, Random Forest, Extra Tree, K-Nearest Neighbors, and Convolutional Neural Network for Sentinel-2 image scene classification; the ML model benchmarking was performed against the existing Sen2Cor package, officially developed by ESA for calibrating and classifying Sentinel-2 imagery; the ML model sensitivity, biasness, and generalization ability were tested over geographically independent images; experimental proof showing that during image scene classification, spectral indices doesn't out weight the raw 13 bands; experimental proof showing that presented ML model can be used to resolve real-world problems like identification of Atmospheric Disturbance.

6.7.1 Limitation

Being composed of several modules, each of them with a high level of complexity, it is certain that our approach does face below limitations:

The proposed ML model is data dependent, and a large amount of labeled data is required to classify more classes; even though a large amount of geometric independent data was used during training, ML models did reach a performance bottleneck; because ML models were trained using 20m resolution images, input images should always be rescaled to 20m; in such cases, band information regarding a pixel may change; in terms of practical application, because ML models are black box models, unlike Sen2Cor, learning and classification cannot be explained.

Chapter 7

Misclassification Detection: Modeling and Results

“In the end you should only measure and look at the numbers that drive action, meaning that the data tells you what you should do next. The goal is to turn data into information, and information into insight.”

— Alexander Peiniger and Carly Fiorina

Consider a multi-classification problem of determining whether an image scene is a Cloud, Cirrus, Shadow, Snow, Water, or Other. Given a new input without true class, is it feasible to determine, whether the new input is classified properly or misclassified? We are not discussing the related confidence of the classification here, but rather a binary metric indicating whether a **classification** is **correct** or **incorrect**.

In this chapter, we attempt to identify the misclassification for a Sentinel-2 image scene classification model using EFM; mainly, we are going to use the results presented in Chapter 6 and formulate and apply the generalized EFM presented in Chapter 4. Furthermore, given the importance of misclassification, it is critical to examine the Chapter 6 results in terms of detecting misclassification. The idea of using distance between unseen observations and the train set and the identification of prediction uncertainty is explored and developed further.

When we talk about classification results from Chapter 6, misclassification can be observed as for each class total number of misclassified points out of support points. Table 7.1 details, for each class and classifier, the percentage of misclassification and support. For example, in the ‘Cloud’ class entry, 16.70%, 9.65%, and 11.69% out of 134315 samples were misclassified by KNN, ET, and CNN, respectively. In this chapter, we attempted to detect these misclassified samples using EFM.

The remainder of the chapter is organized as: Section 7.1 talks about EFM modeling in accordance with previous Chapters 4 and 6 and a general understanding of how misclassification can be detected for image scene classification problem; Section 7.2 details the experimental setup and evaluation matrix used; Section 7.3 shows the results achieved; and finally, Section 7.4 exhibit in-depth discussion leading to summary presented in Section 7.5.

Table 7.1: Sentinel-2 image scene classification misclassified points (in %).

Class	Classifier			Support
	KNN	ET	CNN	
Cloud	16.70	9.65	11.69	134315
Cirrus	44.24	32.06	25.53	175988
Shadow	33.21	22.09	23.63	155715
Snow	13.39	14.95	18.07	154751
Water	16.83	14.35	13.50	117010
Other	30.78	9.36	20.85	174369

7.1 EFM Modeling

Before modeling the EFM to image scene classification, let us review how the EFM model works. EFM is framed of four modules: Train set Engineering, Model Building, Test set Engineering, and Model Applying. Referring to Figure 4.3 of EFM, Figure 7.1 shows the modeling steps undertaken in this section

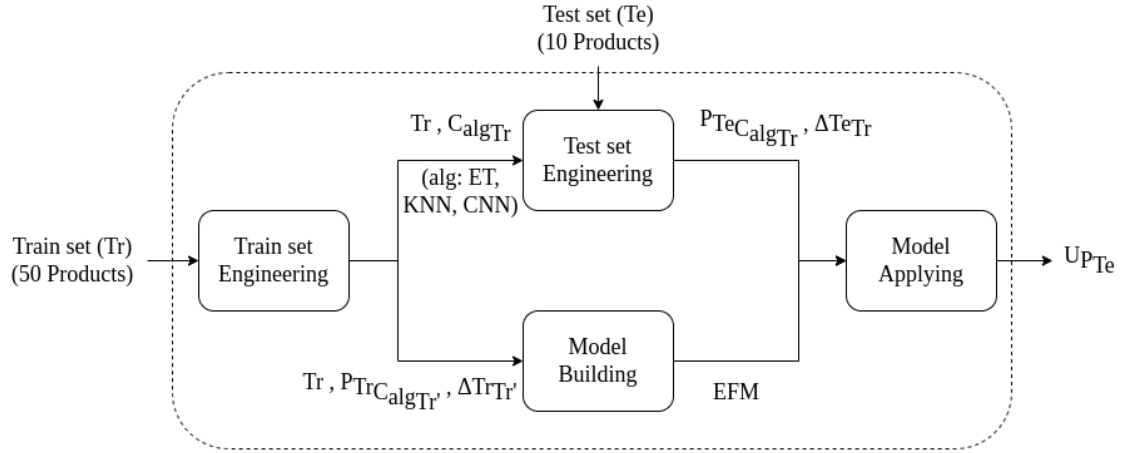


Figure 7.1: Problem-specific EFM modeling: main modules and data-flow.

Following Figure 7.1:

Train set Engineering. As mentioned, this module is responsible for four tasks: dataset split, feature transformation, train a classifier, and make a prediction. Therefore, we have broken down module into 4 steps.

Step 1: Given 50 products, Tr , the first step divides it into two subsets \mathbf{A} , a smaller, 1 product and \mathbf{B} , a larger, remaining 49 products. Therefore, referred to as a dataset split task. **Note:** $A \cap B = \phi$ and $A, B \subset Tr$, and Tr consist of 13 bands as features F and six classes as True_class C .

Step 2: Given A and B from step 1, step 2 divides B into six class-wise subsets, $K = 6$, one for each class $c \in (\text{Cloud, Cirrus, Shadow, Snow, Water, Other})$, resulting B as a $\bigcup_{c=1}^K B_c$.

Distance ΔA_{B_c} is calculated using Equation 7.1 for each point p in A . Where, N is a total

number of points in A , μ_{B_c} is a mean of B_c , and Σ_{B_c} is a covariance of B_c .

$$\Delta A_{B_c} = \sqrt{(Ap - \mu_{B_c})\Sigma_{B_c}^{-1}(Ap - \mu_{B_c})^\top} \text{ where, } \forall p \in N \quad (7.1)$$

At the end, for each point p in A , $\Delta \mathbf{A}_B = \{\Delta A_{B_1}, \Delta A_{B_2}, \dots, \Delta A_{B_K}\}$. Therefore, referred to as a feature transformation task.

As a Result, **six Mahalanobis distances**, $\Delta A_{B_1 \dots K}$ for each point in A , 1 product, from B , 49 products, using B_c mean and covariance is generated, where $c \in C$ and $K = 6$.

Step 3: Given B from step 1, any ML algorithm (for example, distance-based, tree-based, or neural network-based) can be used to train the classifier C_{alg_B} . Therefore, referred to as a train classifier task.

We used K-Nearest Neighbor (KNN), Extra Trees (ET), and Convolutional Neural Network (CNN) classifiers to train over B and the resultant models were C_{KNN_B} , C_{ET_B} , and C_{CNN_B} . **Note:** to train these classifiers over B , we used the same experimental settings mentioned in the Section 6.2 of Chapter 6.

Step 4: Given A from step 1, for each point p in A , a prediction P_A is made using a classifier C_{alg_B} from step 3. In short, $C_{alg_B}(A) \rightarrow P_A$ is calculated. Referred as $\mathbf{P}_{A_{C_{alg_B}}}$. Therefore, referred to as a make predictions task.

In here, as we trained three classifiers algorithms KNN, ET, CNN, for each point in A , thus, we will have **three predictions**, $P_{AC_{KNN_B}}$, $P_{AC_{ET_B}}$, and $P_{AC_{CNN_B}}$.

At the end of step 4, for each point in A , 1 product, we have: *six Mahalanobis distances*, and *three predictions*.

The step 1 splits the train set, 50 products, into two subsets A and B , 1 and 49 products, can be done 50 times (i.e. 50 different combinations). Thus, step 1 to step 4 process is repeated for each of the 50 products, resulting in a dataset consisting of 50 products and for each point, *six Mahalanobis distances*, referred as $\Delta \mathbf{T}r_{T_r'}$ and *three predictions*, referred as $\mathbf{P}_{T_r C_{alg_{T_r'}}}$, where algorithms are KNN, ET, CNN.

At the end of *Train set Engineering* module, outputs Tr , $P_{T_r C_{alg_{T_r'}}}$ and $\Delta Tr_{T_r'}$ are passed to *Model Building* module; and outputs Tr and $C_{alg_{T_r}}$ are passed to *Test set Engineering* module. **Note:** $C_{alg_{T_r}}$ is trained over all the 50 products which is passed to *Test set Engineering* module.

Model Building. The second module is responsible for training a EFM model using the inputs Tr , $P_{T_r C_{alg_{T_r'}}}$ and $\Delta Tr_{T_r'}$ from previous module.

The training of the EFM can be done using any ML algorithm which takes two features as input: Mahalanobis Distance $\Delta Tr_{T_r'}$ and Prediction $P_{T_r C_{alg_{T_r'}}}$ made over Tr using a classifier $C_{alg_{T_r'}}$. Meaning, for EFM training, feature set F as $(\Delta Tr_{T_r'}, P_{T_r C_{alg_{T_r'}}})$ and True Class Label C will come from Tr . Here, algorithms are KNN, ET, CNN.

Model Building module result **EMF** is passed to *Model Applying* module.

For any new observation, the trained Evidence Function Model takes two inputs: the Mahalanobis distance from 50 products distribution mean using Equation 7.2 and a prediction of the new observation made by the classifier trained over 50 products using some ML algorithm, in short, $C_{alg_{Tr}}$.

Test set Engineering. This module is responsible for two tasks, feature transformation and make a prediction. It uses as inputs Tr , $C_{alg_{Tr}}$ from *Train set Engineering* module and test set Te . Therefore, we have broken down module into 2 steps process. **Note:** step 1 is similar to as *Train set Engineering* module step 2 process; here, instead of A , we are using here Te .

Step 1: Given Tr from step 1 and Te , the next step is to divide Tr into six class-wise subsets, $K = 6$, one for each class $c \in (\text{Cloud, Cirrus, Shadow, Snow, Water, Other})$, resulting Tr as a $\bigcup_{c=1}^K Tr_c$.

Distance ΔTe_{Tr_c} is calculated using Equation 7.2 for each point p in Te . Where, N is a total number of points in Te , μ_{Tr_c} is a mean of Tr_c , and Σ_{Tr_c} is a covariance of Tr_c .

$$\Delta Te_{B_c} = \sqrt{(Te_p - \mu_{Tr_c}) \Sigma_{Tr_c}^{-1} (Te_p - \mu_{Tr_c})^\top} \text{ where, } \forall p \in N \quad (7.2)$$

At the end, for each point p in Te , $\Delta Te_{Tr} = \{\Delta Te_{Tr_1}, \Delta Te_{Tr_2}, \dots, \Delta Te_{Tr_K}\}$. Therefore, referred to as a feature transformation task.

Step 2: Given Te , for each point p in Te , a prediction P_{Te} is made using a classifier $C_{alg_{Tr}}$ from step 3 of *Train set Engineering* module. In short, $C_{alg_{Tr}}(Te) \rightarrow P_{Te}$ is calculated. Referred as $P_{Te_{C_{alg_{Tr}}}}$. Therefore, referred to as a make predictions task.

At the end of *Test set Engineering* module, results, *six Mahalanobis distances*, ΔTe_{Tr} and *three predictions*, $P_{Te_{C_{alg_{Tr}}}}$ for each point in Te , 10 product, are passed to *Model Applying* module.

Model Applying. Finally, this module is responsible for one task: produce the Uncertainty $U_{P_{Te}}$ (value: 0 and 1). This module uses the outputs of modules *Model Building* and *Test set Engineering* as inputs.

Given inputs, $P_{Te_{C_{alg_{Tr}}}}$, ΔTe_B , from *Test set Engineering* module, and the trained EFM model from *Model Building* module, it generates $U_{P_{Te}}$ using the statistical distance relation between the train set, Tr , and test set, Te . Therefore, referred to as an uncertainty produce task.

When there is a mismatch between the input P_{Te} and P'_{Te} calculated by EFM, it is referred as ‘classification prediction error’ or ‘misclassification detection’, making EFM a binary model. When EFM predicts 1, the EFM predicted a different class based upon the feature data space representation compared to existing feature value based predictor.

If the above steps were not adequate for explaining EFM modeling, as a supplementary, the (**extremely complex** yet **very easy** to understand version of) overall modeling process is illustrated in Figure 7.2 (Raiyani et al., 2022b). Refer to the process flow from 1 to final result in the Figure.

Processes **1** to **3** are equal to *Train set Engineering* module; processes **4** and **5** are equal to *Test set Engineering* module; process **6** is equivalent to *Model Building* module; and process **7** is equivalent to *Model Applying* module.

Apart from that, $\Delta 1$ to $\Delta 6$ represent the six Mahalanobis distances, one from each class distribution to a point. Class represents value belonging to a true label, i.e. Cloud, Cirrus, Shadow, Snow, Water, Other. Original Prediction can be referred as prediction, P_{Te} , made on test set using trained classifier, $C_{alg_{Tr}}$, over 50 products, using algorithms KNN, ET, CNN. In the end, when there is a mismatch between *Updated Prediction*, P'_{Te} , and *Original Prediction*, P_{Te} , is referred to as ‘classification prediction error’ or ‘misclassification detection’.

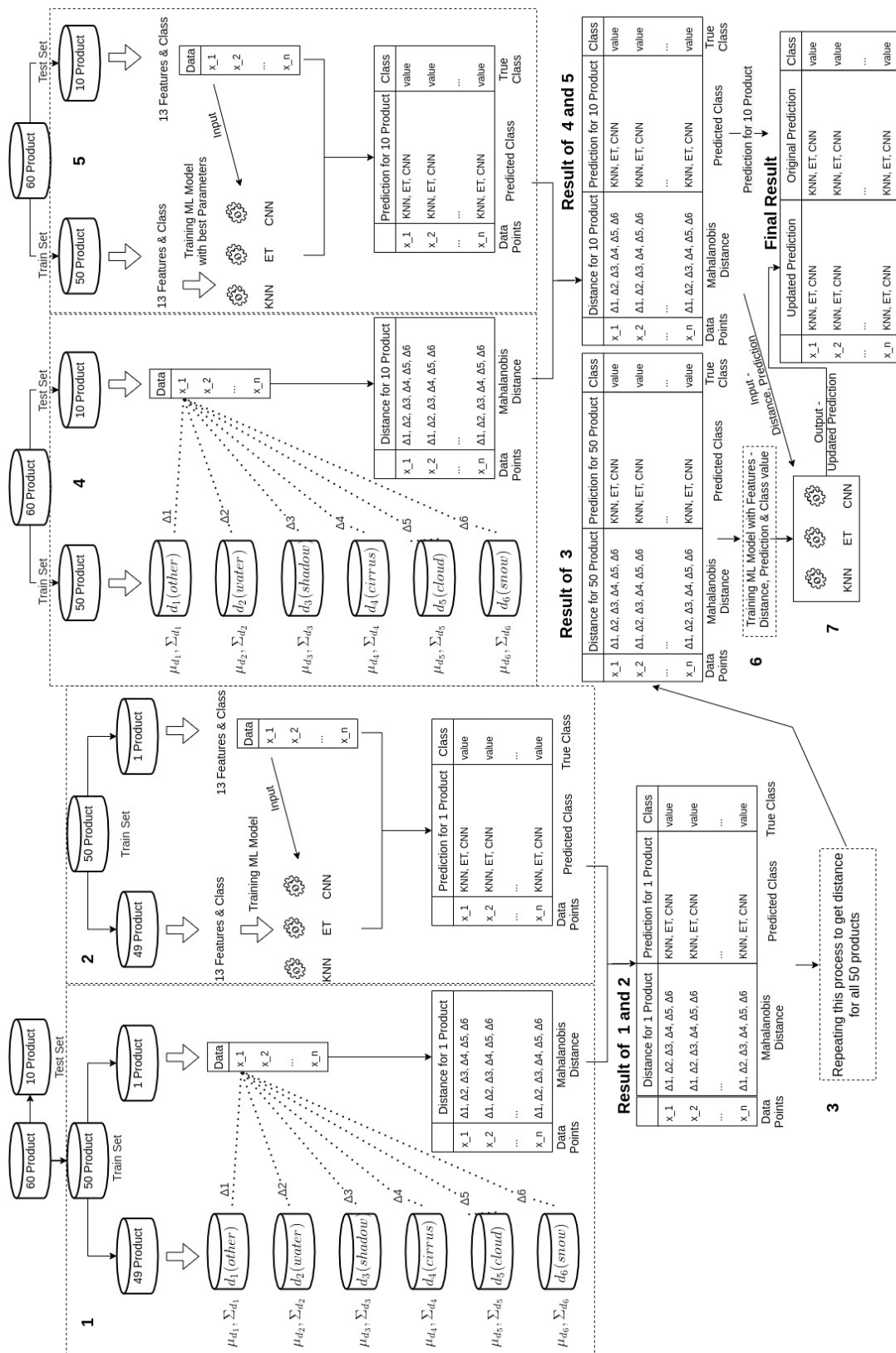


Figure 7.2: Process illustration: generation of Evidence Function Model using Mahalanobis distances between train and test sets.

7.2 Experimental Setup

The training of the Evidence Function Model was done using the Extra Trees (ET) algorithm fine-tuned using a 5 folds cross-validation procedure with micro-F1 measure as assessment. Out of 1000 cross-validation fits, the best model reached a micro-F1 of 76.44%, being this, the proposed Evidence Function Model.

Table 7.2 shows the ET parameter values of EFM model. (`np.linspace`¹)

Table 7.2: Evidence Function Model: Fine-tune Parameter values for Extra Trees (ET) Algorithm.

Parameter	Value	Search Space
criterion	gini	[gini, entropy, log_loss]
n_estimators	177	np.linspace(start = 100, stop = 300, num = 50)
min_samples_split	20	[2, 5, 10, 20, 50]
min_samples_leaf	1	[1]
max_features	sqrt	[auto, sqrt, log2]
max_depth	24	np.linspace(start = 20, stop = 100, num = 20)
bootstrap	True	[True, False]

The information about the experimental setup used to build the proposed solution and the training and test times of the proposed method are presented in Table 7.3.

Table 7.3: Experimental Setup and Time Specifications.

Attribute	Value
Language and Library	Python and Scikit-learn
System Specification	Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz
Train set Engineering	37 minutes
Model Building	4 minutes
Test set Engineering	4 minutes
Model Applying	5 minutes

Precision, recall and F1 score are performance measures that can be used to evaluate ML models. Precision is defined as the ratio between the number of correct positive and all positive results whereas, in recall all relevant samples (all samples that should have been identified as positive) are considered instead of all positive results; F1 is the harmonic mean of Precision and Recall. These measures are calculated per class (considering one class as positive and all the other classes as negative) using Equations 7.3, and average of F1 for all classes, micro-F1 is used.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{F1} = 2 \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7.3)$$

In our experiments, we have defined True Positive, False Positive, and False Negative as

¹numpy.linspace - returns evenly spaced numbers (num) over a specified (start,stop)) interval. <https://numpy.org/doc/stable/reference/generated/numpy.linspace.html>

per Equation 7.4, where $i \in (\text{KNN}, \text{ET}, \text{CNN})$, $k \in (\text{Cloud}, \text{Cirrus}, \text{Shadow}, \text{Snow}, \text{Water}, \text{Other})$.

$$\begin{aligned} \text{TP} &= \text{misclassification_detection}_{(i,k)} - \text{misclassification_detection_error}_{(i,k)} \\ \text{FP} &= \text{misclassification_detection_error}_{(i,k)} \\ \text{FN} &= \text{misclassification}_{(i,k)} - \text{TP} \end{aligned} \tag{7.4}$$

Using these formulas, the micro-F1 performance of EFM in detecting misclassification for different ML models is calculated.

7.3 EFM Results

This section presents the evaluation of EFM over different datasets: the image scene (test set), waterbody, and unlabeled Sentinel-2.

7.3.1 Image scene dataset results

EFM is used to assess three KNN, ET, and CNN Sentinel-2 image scene classifiers, as discussed in this chapter. Thus, Tables 7.4, 7.5, and 7.6 present the misclassification vs. misclassification detection of KNN, ET, and CNN models, respectively.

Each table is made up of columns: **Misclassification**, as seen for each class, as a percentage of total misclassified points; **Overall Detection**, calculated as a percentage of the total number of misclassified points discovered; **Error in Detection**, seen as a proportion of the total number of misclassified points incorrectly identified; **Final Detection**, seen as for each class, percentage of misclassified points correctly recognized out of a total number of misclassified points; **Undetected**, seen as for each class, percentage of misclassified points remained undetected; **Support**, for each class, a total number of test set points.

Table 7.4: Misclassification vs. Misclassification Detection of KNN model.

Class	Misclassification	Overall Detection	Error in Detection	Final Detection	Undetected	Support
Cloud	16.70	12.63	1.17	11.46	5.24	134315
Cirrus	44.24	30.87	3.65	27.22	17.02	175988
Shadow	33.21	24.60	6.15	18.45	14.76	155715
Snow	13.39	12.76	7.03	5.73	7.66	154751
Water	16.83	9.50	2.06	7.44	9.39	117010
Other	30.78	28.56	1.11	27.45	3.33	174369

Table 7.4 present the misclassification vs. misclassification detection of the KNN model. To better understand the results, consider the example of the ‘Cloud’ class entry out of 134315 points: 16.70% (22430 points) were misclassified; 12.63% (16964 points) were detected misclassified; 1.17% (1571 points as a False Positive) were wrongly detected as misclassified; 11.46% (15392 points as a True Positive) were correctly detected as misclassified; 5.24%

(7038 points as a False Negative) were remained undetected. Resulting in 68.62% Recall, 90.74% Precision, and 78.14% F1 in misclassified detection. Figure 7.3 shows a visual representation of the true positive, false positive, and false negative for each class.

Table 7.4 shows that undetected misclassification ranges 13.69% from a minimum of 3.33% for ‘Other’ to a maximum of 17.02% for ‘Cirrus’; detection error ranges 5.92% from a minimum of 1.11% for ‘Other’ to a maximum of 7.03% for ‘Snow’; final detection ranges 21.72% from a minimum of 5.73% for ‘Snow’ to a maximum of 27.45% for ‘Other’.

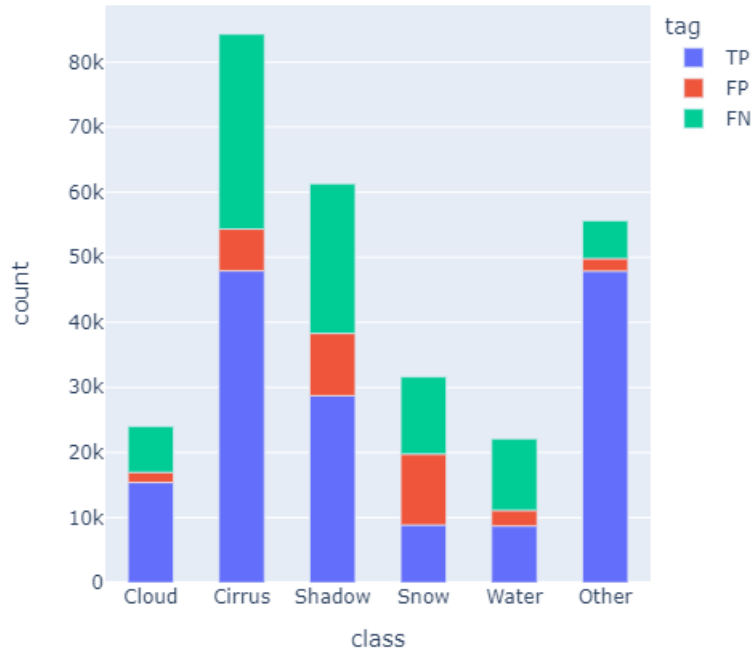


Figure 7.3: Misclassification Detection of KNN model.

Table 7.5: Misclassification vs. Misclassification Detection of ET model.

Class	Misclassification	Overall Detection	Error in Detection	Final Detection	Undetected	Support
Cloud	9.65	1.77	0.26	1.51	8.14	134315
Cirrus	32.06	16.98	1.47	15.51	16.55	175988
Shadow	22.09	6.37	2.80	3.57	18.52	155715
Snow	14.95	4.81	1.53	3.28	11.67	154751
Water	14.35	2.62	1.02	1.60	12.75	117010
Other	9.36	5.41	0.32	5.09	4.27	174369

Table 7.5 present the misclassification vs. misclassification detection of the ET model. To better understand the results, consider the example of the ‘Cloud’ class entry out of 134315 points: 9.65% (12961 points) were misclassified; 1.77% (2377 points) were detected misclassified; 0.26% (349 points as a False Positive) were wrongly detected as misclassified; 1.51% (2028 points as a True Positive) were correctly detected as misclassified; 8.14%

(10933 points as a False Negative) were remained undetected. Resulting in 15.65% Recall, 85.31% Precision, and 26.45% F1 in misclassified detection. Figure 7.4 shows a visual representation of the true positive, false positive, and false negative for each class.

Table 7.5 shows that undetected misclassification ranges 14.25% from a minimum of 4.27% for ‘Other’ to a maximum of 18.52% for ‘Shadow’; detection error ranges 2.54% from a minimum of 0.26% for ‘Cloud’ to a maximum of 2.80% for ‘Shadow’; final detection ranges 14.00% from a minimum of 1.51% for ‘Cloud’ to a maximum of 15.51% for ‘Cirrus’.

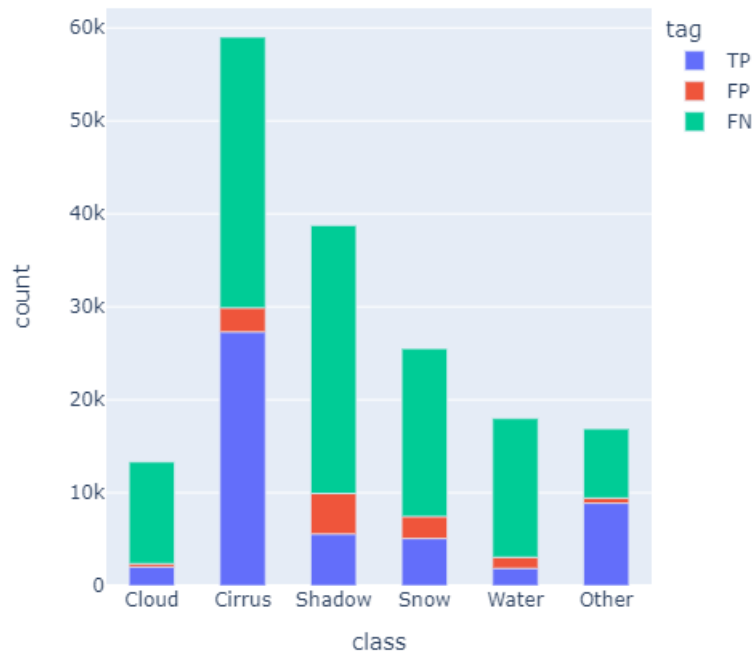


Figure 7.4: Misclassification Detection of ET model.

Table 7.6: Misclassification vs. Misclassification Detection of CNN model.

Class	Misclassification	Overall Detection	Error in Detection	Final Detection	Undetected	Support
Cloud	11.69	4.03	0.01	4.02	7.67	134315
Cirrus	25.53	23.19	6.12	17.07	8.46	175988
Shadow	23.63	7.93	4.24	3.69	19.94	155715
Snow	18.07	2.22	0.50	1.72	16.35	154751
Water	13.50	5.81	4.36	1.45	12.05	117010
Other	20.85	8.31	0.56	7.75	13.10	174369

Table 7.6 present the misclassification vs. misclassification detection of the CNN model. To better understand the results, consider the example of the ‘Cloud’ class entry out of 134315 points: 11.69% (15701 points) were misclassified; 4.03% (5413 points) were detected misclassified; 0.01% (13 points as a False Positive) were wrongly detected as misclassified; 4.02% (5400 points as a True Positive) were correctly detected as misclassified; 7.67%

(10301 points as a False Negative) were remained undetected. Resulting in 34.39% Recall, 99.75% Precision, and 51.15% F1 in misclassified detection. Figure 7.5 shows a visual representation of the true positive, false positive, and false negative for each class.

Table 7.6 shows that undetected misclassification ranges 12.27% from a minimum of 7.67% for ‘Cloud’ to a maximum of 19.94% for ‘Shadow’; detection error ranges 6.11% from a minimum of 0.01% for ‘Cloud’ to a maximum of 6.12% for ‘Cirrus’; final detection ranges 15.62% from a minimum of 1.45% for ‘Water’ to a maximum of 17.07% for ‘Cirrus’.



Figure 7.5: Misclassification Detection of CNN model.

Using Equations 7.3 and 7.4, Precision, Recall, and F1 performance of the Evidence Function Model in-detecting the misclassification of KNN, ET, and CNN model were calculated respectively, and the results are presented in Tables 7.7 and 7.8.

Table 7.7: Precision and Recall of EFM in-detecting the misclassification of KNN, ET, and CNN models.

Class	Precision			Recall			Support
	KNN	ET	CNN	KNN	ET	CNN	
Cloud	90.74	85.31	99.75	68.62	15.65	34.39	134315
Cirrus	88.18	91.34	73.61	61.53	48.38	66.86	175988
Shadow	75.00	56.04	46.53	55.56	16.16	15.62	155715
Snow	44.91	68.19	77.48	42.79	21.94	9.52	154751
Water	78.32	61.07	24.96	44.21	11.15	10.74	117010
Other	96.11	94.09	93.26	89.18	54.38	37.17	174369

Table 7.8: F1 of EFM in-detecting the misclassification of KNN, ET, and CNN models.

Class	KNN	ET	CNN	Support
Cloud	78.14	26.45	51.15	134315
Cirrus	72.48	63.26	70.07	175988
Shadow	63.83	25.09	23.39	155715
Snow	43.82	33.20	16.96	154751
Water	56.52	18.86	15.02	117010
Other	92.52	68.92	53.15	174369

After analysing Tables 7.7 and 7.8, the following observations can be made:

In terms of Recall, for KNN, ranges 46.39% from a minimum of 42.79% for ‘Snow’ to a maximum of 89.18% for ‘Other’; for ET, ranges 43.23% from a minimum of 11.15% for ‘Water’ to a maximum of 54.38% for ‘Other’; for CNN, ranges 46.39% from a minimum of 57.34% for ‘Snow’ to a maximum of 66.86% for ‘Cirrus’. Overall, for all the classes, KNN has better recall compared to ET and CNN as the mean recall is 60.32%, 27.94%, and 29.05%, respectively.

In terms of Precision, for KNN, ranges 51.20% from a minimum of 44.91% for ‘Snow’ to a maximum of 96.11% for ‘Other’; for ET, ranges 38.02% from a minimum of 56.04% for ‘Shadow’ to a maximum of 94.09% for ‘Other’; for CNN, ranges 74.79% from a minimum of 24.96% for ‘Water’ to a maximum of 99.75% for ‘Cloud’. Overall, for all the classes, KNN has better precision compared to ET and CNN as the mean precision is 78.88%, 76.01%, and 69.27%, respectively.

In terms of F1, for KNN, ranges 48.70% from a minimum of 43.82% for ‘Snow’ to a maximum of 92.52% for ‘Other’; for ET, ranges 50.06% from a minimum of 18.86% for ‘Water’ to a maximum of 68.92% for ‘Other’; for CNN, ranges 55.05% from a minimum of 15.02% for ‘Water’ to a maximum of 70.07% for ‘Cirrus’. Overall, for all the classes, KNN has better micro-F1 compared to ET and CNN as the mean F1 is 67.89%, 39.30%, and 38.29%, respectively.

Low recall and high precision, EFM detects very few misclassification, but most of detected points are correct when classified points are feed. The Evidence Function Model employed for the KNN model performs 29.60% better than ET and CNN in recognizing misclassified points.

7.3.2 Waterbody dataset results

As mentioned previously, waterbody dataset consists only 2355498 points. Over this single ‘Water’ class dataset, Table 7.9 present the misclassification vs. misclassification detection of KNN, ET, and CNN models, respectively. **Note:** the same classifier, trained in previous chapter, is used to classify these dataset points, i.e. C_{algTr} .

To better understand Table 7.9 results, consider the example of KNN model, out of 2355498 points: 39.51% (930657 points) were misclassified; 11.21% (264051 points) were detected misclassified; 2.41% (56767 points as a False Positive) were wrongly detected as misclassified; 8.80% (207283 points as a True Positive) were correctly detected as misclassified;

Table 7.9: Misclassification vs. Misclassification Detection of KNN, ET, and CNN models.

Classifier	Misclassification	Overall Detection	Error in Detection	Final Detection	Undetected
KNN	39.51	11.21	2.41	8.80	30.71
ET	40.04	17.11	0.26	16.85	23.19
CNN	45.29	12.56	0.03	12.53	32.76

30.71% (723373 points as a False Negative) were remained undetected. Resulting in 22.27% Recall, 78.50% Precision, and 34.70% micro-F1 in misclassified detection.

Similarly, for ET model, out of 2355498 points: 40.04% (943141 points) were misclassified; 17.11% (403025 points) were detected misclassified; 0.26% (6124 points as a False Positive) were wrongly detected as misclassified; 16.85% (396901 points as a True Positive) were correctly detected as misclassified; 23.19% (546239 points as a False Negative) were remained undetected. Resulting in 42.08% Recall, 98.48% Precision, and 58.96% micro-F1 in misclassified detection.

Similarly, for CNN model, out of 2355498 points: 45.29% (1066805 points) were misclassified; 12.56% (295850 points) were detected misclassified; 0.03% (706 points as a False Positive) were wrongly detected as misclassified; 12.53% (295144 points as a True Positive) were correctly detected as misclassified; 23.19% (771661 points as a False Negative) were remained undetected. Resulting in 27.67% Recall, 99.76% Precision, and 43.32% micro-F1 in misclassified detection.

Using Equations 7.3 and 7.4, Precision, Recall, and F1 performance of the Evidence Function Model in-detecting the misclassification of KNN, ET, and CNN model were calculated respectively, and the results are presented in Table 7.10.

Table 7.10: Precision, Recall, and micro-F1 of EFM in-detecting the misclassification of KNN, ET, and CNN models.

Classifier	Precision	Recall	micro-F1
KNN	78.50	22.27	34.70
ET	98.48	42.08	58.96
CNN	99.76	27.67	43.32

Analysing Table 7.9 and 7.10:

The undetected misclassification ranges 9.57% from a minimum of 23.19% for ET to a maximum of 32.76% for CNN; detection error ranges 2.38% from a minimum of 0.03% for CNN to a maximum of 2.41% for KNN; final detection ranges 8.05% from a minimum of 8.80% for KNN to a maximum of 16.85% for ET.

In terms of Recall, ranges 19.81% from a minimum of 22.27% for KNN to a maximum of 42.08% for ET; overall, ET has better recall compared to KNN and CNN.

In terms of Precision, ranges 21.26% from a minimum 78.50% for KNN to a maximum of 99.76% for CNN; overall, CNN has better precision compared to KNN and ET.

In terms of micro-F1, ranges 24.26% from a minimum 34.70% for KNN to a maximum of 58.96% for ET; overall, ET has better micro-F1 compared to KNN and ET.

Further, to visually analyze the water-body classification vs misclassification detection, randomly 16/49 water-bodies RGB images are shown in Appendix C Figure C.1.

7.3.3 Unlabeled dataset results

As mentioned previously, two unlabeled Sentinel-2 images are chosen to have a better grasp of the differences between the ML model classification and misclassification detection.

Note: the same classifier, trained in previous chapter, is used to classify these images, i.e. $C_{alg_{Tr}}$.

Figure 7.6 shows the Fiji Sentinel-2 Level-1C RGB image, as well as the results of the classification and misclassification detection. From figure, below observations are made:

- For KNN classified image, KNN has misclassified ‘Water’ as ‘Cloud’ and ‘Other’ as ‘Snow’. Out of those misclassified points, most of them are detected by EFM.
- For ET classified image, ET has misclassified ‘Other’ as ‘Cloud and Snow’. Out of misclassified points, most of them are detected by EFM.
- For CNN classified image, CNN has misclassified ‘Water’ and ‘Other’ as ‘Shadow, Cloud, and Snow’. Out of misclassified points, for ‘Shadow’, very few, and, for ‘Cloud and Snow’, most of them are detected by EFM.
- In terms of the overall classification, CNN performs the worst while ET performs the best; however, in terms of overall misclassification detection, CNN performs the worst while KNN and ET performs the best.

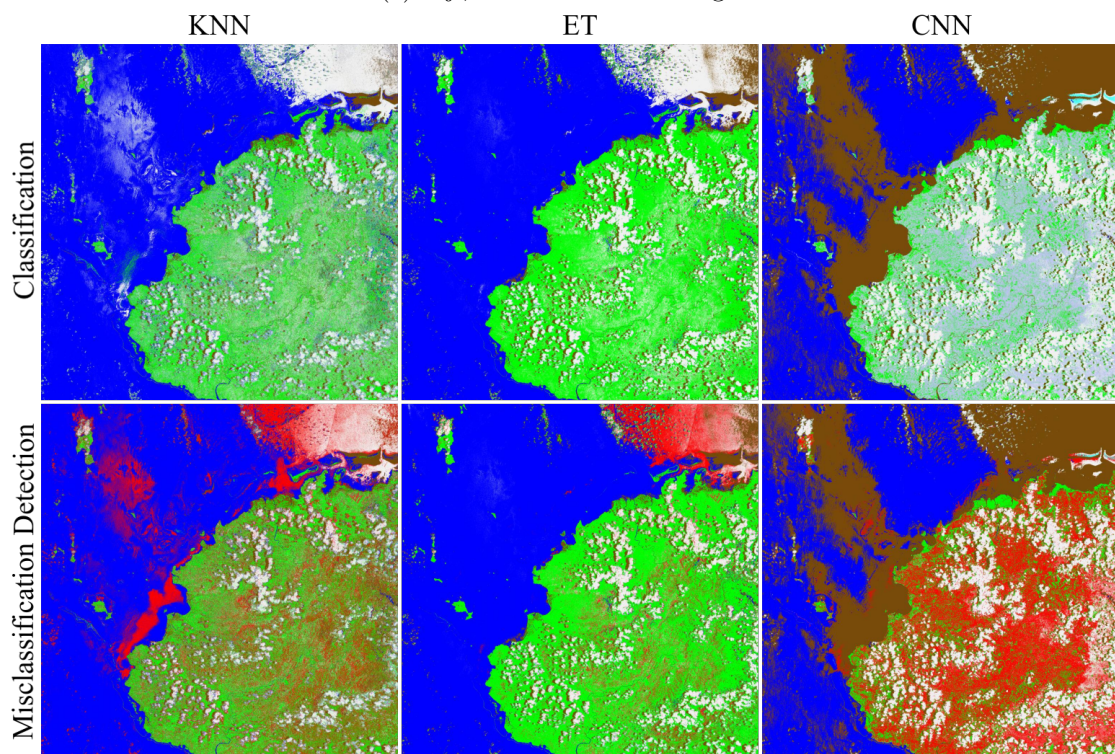
Figure 7.7 shows the Portugal Sentinel-2 Level-1C RGB image, as well as the results of the classification and misclassification detection.

From Figure 7.7, below observations are made:

- For KNN classified image, KNN has misclassified ‘Water, Shadow, and Other’ as ‘Cloud, Cirrus, Snow and Other’. Out of those misclassified points, most of them are detected by EFM.

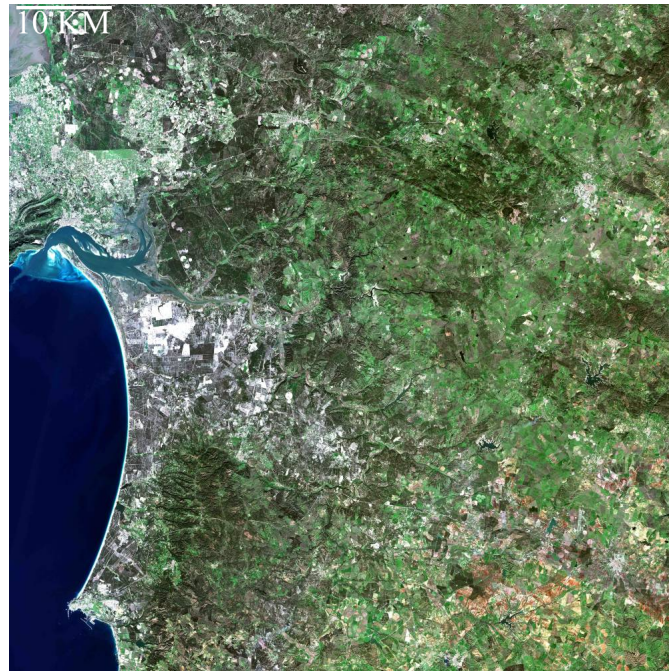


(a) Fiji, Sentinel-2 RGB image.

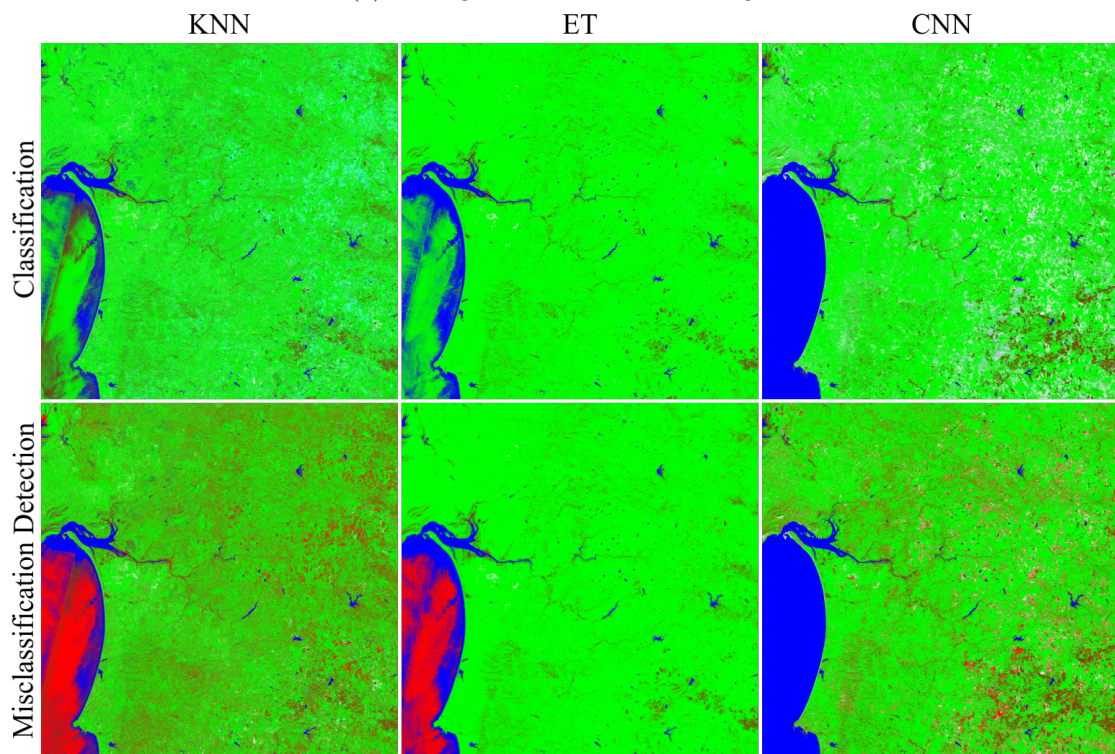


(b) Classification vs. Misclassification Detection of KNN, ET, and CNN models.

Figure 7.6: Classification represent the output from the ML classifier (KNN, ET and CNN), and Misclassification Detection shows the error detected over classified images. Color labels: Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan, Other as Green, Error as Red.



(a) Portugal, Sentinel-2 RGB image.



(b) Classification vs. Misclassification Detection of KNN, ET, and CNN models.

Figure 7.7: Classification represent the output from the ML classifier (KNN, ET and CNN), and Misclassification Detection shows the error detected over classified images. Color labels: Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan, Other as Green, Error as Red.

- For ET classified image, ET has misclassified ‘Water, Shadow, and Other’ as ‘Cloud, Cirrus, Snow and Other’. Out of misclassified points, most of them are detected by EFM.
- For CNN classified image, CNN has misclassified ‘Shadow’ and ‘Other’ as ‘Cloud, Cirrus, and Snow’. Out of misclassified points, most of them are detected by EFM.
- In terms of the overall classification, KNN and CNN perform the worst while ET performs the best; however, in terms of overall misclassification detection, all three perform equally best.

7.4 Discussion

This section discusses specific elements of the EFM in terms of its modeling and working principle and its output with respect to the classifiers of Chapter 6.

EFM Results: Based on the results in Tables 7.8 and 7.9, it is clear that the KNN classifier outperforms the ET and CNN in terms of misclassification detection. Although the proposed function for measuring classification prediction error is independent of the classification algorithm, the proposed model used over a distance-based classifier outperformed tree-based and neural network-based classifiers. This leads to a claim that a ‘distance-based classifier has a high correlation between true prediction vs false prediction in terms of Mahalanobis distance measurement.’ This claim is supported by the Table 7.8 showing that the Evidence Function Model with KNN performs 29.60% better for identifying misclassified.

Error in EFM Results: Based on the findings in the preceding section, we believe that quantifying the error made by the Evidence Function Model (as False Positive) is an important step toward determining a model’s suitability. It also explains the experimental data-space distribution in terms of the Mahalanobis distance. The model’s suitability is demonstrated in Tables 7.4 to 7.6, where the total mean error in misclassification detection is 3.53%, 1.23%, and 2.63% for the KNN, ET, and CNN models, respectively; for the waterbody dataset, the error in misclassification detection is 2.41%, 0.26%, and 0.03% for the KNN, ET, and CNN models, respectively.

EFM Modeling: Following EMF modeling, *Train set Engineering* module steps 1 and 4, task Data Split and Prediction, input data was divided into A , 1 product and B , 49 products. During the experimental process, when different data splits were made, the question ‘why is the classification of 1 product done using 49 products?’ was raised. Reasoning, to take advantage of the relationship inform-of statistical-distance/data-space from one product to 49 products for true vs predicted class. As a result, each product in the training set generated sufficient evidence/explanation for the predicted class, allowing us to train a machine learning EFM on the generated data.

Consider Figure 7.8 for a Parallel Coordinates (Inselberg, 1985) visualization of the six Mahalanobis Distances from a test set to a train set for true classification vs. misclassification to gain a better understanding of the relationship between statistical distance and data-space among train test sets. (**Note:** in this case, classifiers were trained with 50 products and tested with 10 products.)

Different plots in Figure 7.8 depict the statistical pattern recognition (Iwamura et al., 2000) between correct/incorrect prediction and train samples. In the KNN true classification vs. misclassification plot for class ‘Snow’, plot six, for example, the Mahalanobis distance between train set ‘Water’ distribution and test set points is much greater than 150+ for misclassification compared to true classification. Similarly, the Mahalanobis distance between the ‘Cirrus’ distribution in the train set and the test set points ranges from 100 to 150 for true classification versus misclassification. Thus, Figure 7.8 explains the predicted class and its relationship to a train set in terms of Mahalanobis distance and data space representation.

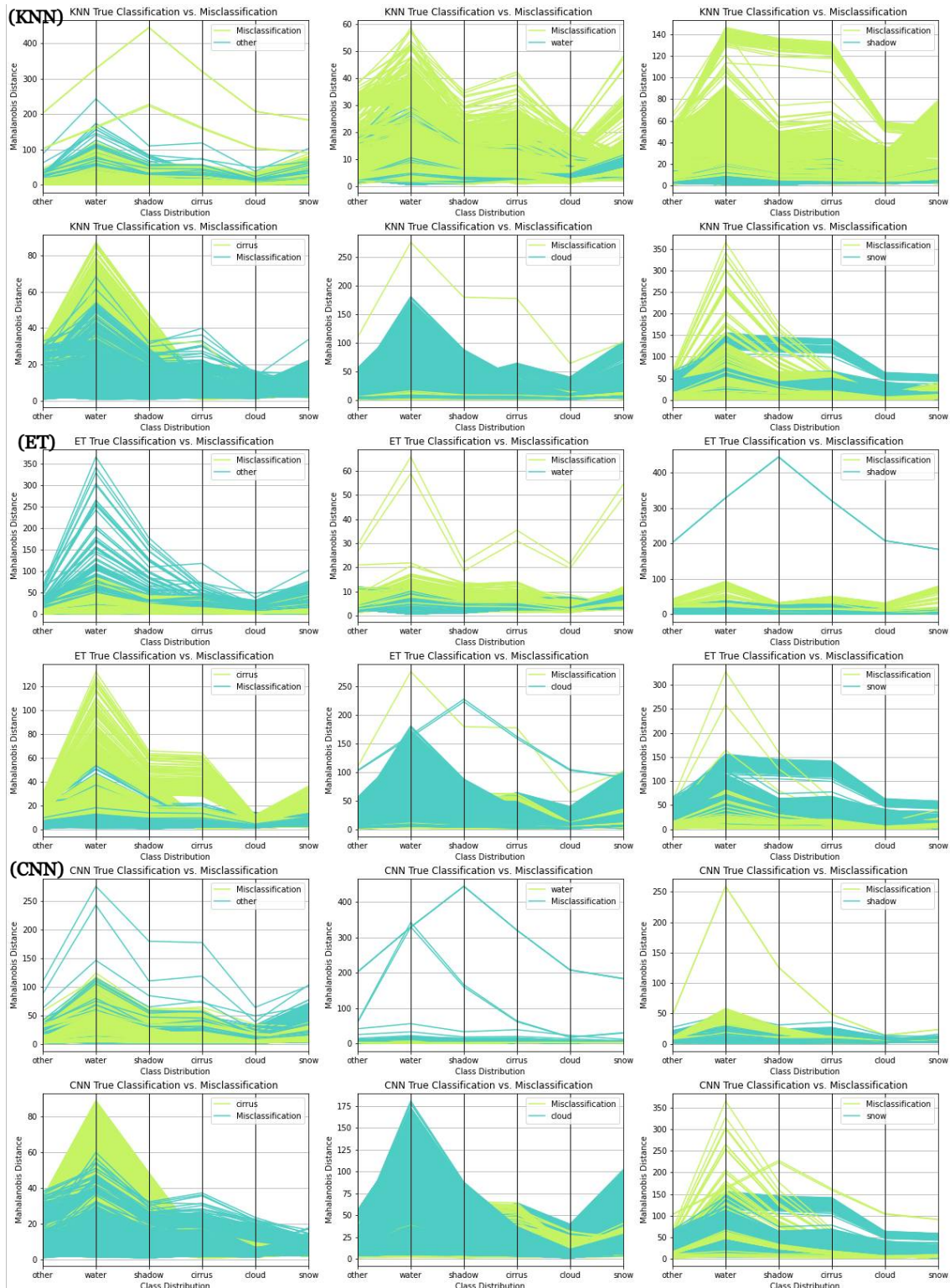


Figure 7.8: Six Mahalanobis distance based Parallel Coordinates visualization of test set from train set: True classification vs. Misclassification for KNN, ET, and CNN classifiers.

Mahalanobis distances as a Feature for Image Scene Classification: The image scene pixel classifier was trained in Chapter 6 using the 13 raw bands as features. It would be interesting to evaluate the performance of a classifier trained with Mahalanobis distances as features rather than raw bands.

Referring to the Train set Engineering module of the EMF modeling section, 13 raw band values were transformed into six Mahalanobis distances for each point in 50 products. Table 7.11 compares the classifier trained with 13 bands to Mahalanobis distances tested over 10 products using modified six Mahalanobis distances as a feature set. The overall micro-F1 score for KNN and ET did not change considerably; for CNN, the micro-F1 declined by 6.00% when the Mahalanobis distances were used as features.

Table 7.11: Image Scene Classification - 13 Bands vs. Mahalanobis distances.

Class	Using 13 Bands			Using 6 Distances ²		
	KNN	ET	CNN	KNN	ET	CNN
Other	68	83	82	67	78	70
Water	84	90	90	77	88	87
Shadow	68	81	83	62	73	70
Cirrus	61	79	76	72	80	83
Cloud	76	86	84	80	88	75
Snow	84	90	91	84	87	89
micro-F1	73	84	84	73	82	78

Aside from our primary goal of identifying misclassification, the Mahalanobis distances could be used as extra features when training a classifier. Following this logic, we trained the classifiers using 13 raw bands + six Mahalanobis distances as feature space and compared the obtained results with only 13 bands, Table 7.12 presents them. From the Table, it is possible to conclude that for all three ML models, using 13 bands + 6 Mahalanobis distances as feature increases the micro-F1 score.

Table 7.12: KNN, ET, and CNN: result comparison between model trained using 13 bands vs. 13 bands + 6 Mahalanobis distance over test set.

Class	Using 13 Bands			Using 13 Bands + 6 Distances ²		
	KNN	ET	CNN	KNN	ET	CNN
Other	68	83	82	69	83	80
Water	84	90	90	78	91	89
Shadow	68	81	83	63	82	80
Cirrus	61	79	76	72	79	80
Cloud	76	86	84	81	86	89
Snow	84	90	91	84	90	88
micro-F1	73	84	84	74	85	84

According to the results in Table 7.11, six Mahalanobis distances as feature classifiers performed marginally worse/less than 13 Bands, whereas having them as additional features,

²KNN and ET are trained using default values whereas CNN using Section 6.2 settings.

performed the same/marginally better, can be seen in Table 7.12.

7.5 Summary

A summary of the Chapter *Misclassification Detection: Modeling and Results* is provided below:

Using the generalized concept of EFM presented in Chapter 4, introduced modeling of EFM for Sentinel-2 image scene classification problem; experimental demonstration proving EFM can identify misclassification for the problem of Sentinel-2 image scene classification; generalization ability of EFM was tested across different datasets; and presented a discussion demonstrating the use of Mahalanobis distances as a feature for image scene classification.

7.5.1 Limitation

Being composed of several modules, each of them with a high level of complexity, it is certain that our approach does face below limitations:

The proposed model must compute the Mahalanobis distance from the train set for each new input, the computation complexity may be considerable if large volumes of images are supplied; for output-sensitive applications where false negatives are critical, our model with a low recall cannot be employed; we only addressed pixel-level scene classification in our trials, thus we cannot comment on EFM's performance on object-level classification tasks; similarly, we cannot comment on the performance of the EFM model on other domain datasets like text or audio data.

Chapter 8

Abbreviating Train Cost: Modeling and Results

“Perhaps we should all take a minute to reflect not only on making AI more intelligent & effective but rather on how it might assist humankind.”

— Stephen Hawking

As mentioned in Chapter 2, a lot of researchers in the field of remote sensing have recently turned their attention to scene classification, or the segmentation of regions into morphological categories such as land, ocean, cloud (Mohajerani et al., 2018).

Reiterating from Chapter 3, as paired-label datasets have been enriched throughout time, supervised machine learning has become an essential stage in every problem-solving process. When there aren't enough manual labels for the massive amount of publicly available data, one can consider an active learning approach.

For the pixel-based image scene classification case, in active learning scenario, an algorithm rates the unlabeled pixels based on predefined criteria and automatically selects those that are deemed to be the most valuable for labeling. The expert then manually labels the selected pixels, and the procedure is repeated. The system generates the minimum best collection of samples, suboptimal set, achieving a predefined classification accuracy.

In Chapter 6, we presented ML models trained using 50 labeled Sentinel-2 images and evaluated over 10 images, and the highest F1-micro attained by Extra Tree (ET) model was 84% for image scene classification.

Now, “Is it possible to use fewer labeled samples S (where $S \lll 5.7$ millions) during the training phase and achieve the same accuracy of 84% over the test set?”

To answer above, in this chapter, we have proposed a *Generalized Sampling Algorithm* and, with experimental results, proved its working principle. Also, the goal of the presented work here is to elaborate more broadly on the usefulness of *Evidence Functions Model as a query selection methodology*, comparing EFM's performance in active learning.

Given the importance of scene classification, this chapter presents an active learning ap-

proach for reducing Sentinel-2 image scene classification training data.

The remainder of the chapter is organized as follows: Section 8.1 proposes and talks about general understanding of proposed sampling algorithm which can work with any query selection methods; Section 8.2 presents two selection methods and its usage with proposed sampling algorithm is showcased in Section 8.3; Section 8.4 details the experimental setup used; Section 8.5 shows the results achieved; and finally, Section 8.6 exhibit in-depth discussion leading to summary presented in Section 8.7.

8.1 Sampling Algorithm

In Chapter 3, we detailed three: membership, stream, and pool based sampling methods. We evaluated the commonly used pool-based sampling strategy in our work, in which the most useful data samples are chosen from a pool of unlabeled data using some selection methods or informativeness measures explained in the next section.

Table 3.2 of Chapter 3 describes different existing selection methods that encompass the sampling methods stated above. Taking them all into account, a *Generalized Sampling Algorithm 1* is formulated. Here, $micro-F1_{ac}$ represents the micro-F1 score obtained using active learning methods. (f1_score library ¹.)

Algorithm 1 Generalized Sampling Algorithm

Input:

Unlabeled dataset U .

Initial train set X . Where, $X \lll U$

Test set T (with data-label pair).

Output:

Optimal dataset S . Where, $X < S < U$

```

1: procedure SAMPLING_STRATEGIES( $U, X, T$ )
2:    $micro-F1_{ac} = 0$ 
3:    $S = X$ 
4:   for  $S < U$  do
5:     Train classifiers  $C_1$  and  $C_2$  over  $S$ .
6:     Using  $C_1$  calculate  $U_{pred}$  and using  $C_2$   $T_{pred}$ .
7:      $micro-F1_{ac} = f1\_score(T_{true}, T_{pred}) \triangleright f1\_score$  is a library from sklearn.metrics
8:     if  $micro-F1_{ac} > F1$  then return  $F$   $\triangleright F1$  is a predefined micro-F1 score over
       test-set
9:     else
10:      select  $E$  samples from  $U$  using  $M_i(S, U, U_{pred})$ .  $\triangleright$  Here, for each class in  $U$ ,
         $E$  samples are selected.  $M_i$  is a sampling strategy
11:       $S = S \cup E$   $\triangleright$  Updating  $S$  with the  $E$  chosen samples
12:       $U = U \setminus E$   $\triangleright$  Removing  $E$  chosen samples from  $U$ 
13:    end if
14:  end for
15: end procedure

```

The algorithm takes three inputs: an unlabeled dataset U , an initial train-set set $X \lll$

¹https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html

U , and a test set T containing a data-label pair and outputs an optimal dataset S . After considering $\text{micro-F1}_{ac} = 0$ with $S = X$, classifiers C_1 and C_2 are trained over S and predictions over U and T are made resulting, U_{pred} and T_{pred} , respectively. T_{true} and T_{pred} are later used to update micro-F1_{ac} . If micro-F1_{ac} achieves a certain level of accuracy, the optimal dataset S is returned; otherwise, E samples are chosen from U for labeling and added to S . This procedure is repeated until the optimal dataset is found. Here, E is chosen not at random but rather according to a problem-oriented heuristic that aims to maximize the classifiers' performance. The E points identified as the most informative are added to S .

8.2 Selection Methods

In our experiments, two selection methods are used: Entropy based, referred to as M_{en} where M stands for method and en for entropy and Mahalanobis distance based, referred to as M_{md} with md referring Mahalanobis distance.

8.2.1 Entropy based method

The conditional Entropy is an information-theoretic measure of a random variable uncertainty (Shannon, 1948b). The entropy measurement, is the most prominent measurement used in active learning pool-based sampling processes.

The conditional Entropy is computed using Equation 8.1, where y is the class, with $y \in Y = y_1, y_2, \dots, y_k$ and $P(y|x)$ is the a posteriori conditional probability. $H(Y|x)$ is the uncertainty measurement function based on the classifier's posterior distribution entropy estimation (Roy and McCallum, 2001a). This measure, also known as traditional uncertainty sampling (Nguyen et al., 2022), which will be the baseline for the experiments.

$$H(Y|x) = - \sum_{y \in Y} P(y|x) \log P(y|x) \quad (8.1)$$

8.2.2 Mahalanobis distance based method

In Chapter 7, we conjecture that, given a train test split and different classifiers built over the train set, it is possible to find an Evidence Function for the prediction using the Mahalanobis distance relation between the train and the test sets.

Knowing the prediction error can be used as an additional feature in the field of 'Disagreement based Active Learning' (Dasgupta and Langford, 2009). Here, idea is to query from dense regions where EFM predictions $P'(X)$ mismatch with classifier predictions $P(X)$. The reasoning is that measuring the prediction error can aid in the generation of labeled datasets, with human input required only for misclassified data (Hanneke, 2014).

The Mahalanobis Distance Δ between a point \mathbf{x}_i and a distribution D with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ is given by Equation 8.2.

$$\Delta = \sqrt{(\mathbf{x}_i - \boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \boldsymbol{\mu})^\top} \quad (8.2)$$

8.3 Modeling Active Learning

Given the image scene dataset, $Tr = 5,716,330$ labeled training points from a set of classes Cloud, Cirrus, Shadow, Snow, Water, Other described by a set of attributes 13 Bands and $Te = 912,148$ testing points, the formulation of active learning can be done as:

Looking at the Algorithm 1, the idea is to start by using $S \subset Tr$ points for training, and predict the rest of the U points where, $U = Tr \setminus S$ and Te is the set of test points. Over the Te points, the trained model will achieve some performance. Subsequently, among the U predicted samples, find the E points with highest uncertainty using Entropy and EFM methods for labeling and added to S . This process will continue until the trained model reaches a performance higher or equal to 84% resulting in optimal $S \lll Tr$ samples.

Entropy is computed for each prediction using Equation 8.1 in step-10 of the generalized Algorithm 1, provided with U and U_{pred} . Following that, a total of E samples are selected for the highest entropy value.

The Mahalanobis distance for each point in S to U is computed using Equation 8.2 in step-10 of the generalized Algorithm 1, provided with three inputs S , U , and U_{pred} . Following that, a total of E samples are selected using their probabilistic measure based on the discovered misclassification points over U . In other words, the top E samples with high EFM confidence are chosen.

8.4 Experimental Setup

Table 8.1 presents the experimental setup used to build the proposed solution. Here, we have used the same EFM setup mentioned in Section 7.1; **Batch Size** refers to every iteration per class, the number of samples to be added in S ; **Max Iteration** refers to how many maximum iterations of adding labeled samples should continue; **Max. No. of Samples Labeled** refers to the maximum number of labeled samples that should be added during the active learning process. **Note:** Because the active learning process is computationally expensive, we have established a maximum limit of 20000 labeled samples to be inserted per class in S throughout our experiments, making the maximum limit of $S = 20000 * 6$ as we have six classes. Thus, a combination of $Batch_Size * Max_Iteration * 6$ is proposed which never exceeds 120000.

Table 8.1: Experimental setup.

Attribute	Value
Language and Library	Python and Scikit-learn
System Specification	Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz
Evidence Function Setup	Same setup as Section 7.1
[Batch Size, Max Iteration] [E , N]	[10, 2000], [50, 400], [100, 200], [200, 100], and [500, 40]
Max. No. of Samples Labeled	120,000 (initially $E * 6$ up to $20,000 * 6$) i.e. $E * N * 6 \leq 120000$

The fifth step of the Generalized Sampling Algorithm 1 shows the training of classifiers C_1 and C_2 . We employed the Extra Trees technique as a classifier in this case, and Table 8.2 outlines the algorithm settings. Note that the parameters for C_1 are chosen in such a way that we may have a four-digit probability value; for C_2 , the same parameters as specified in Table 6.2 are chosen as we predict the test set.

Table 8.2: Extra Trees (ET) Algorithm Parameters.

Parameter	C_1	C_2
criterion	gini	gini
n_estimators	1000	177
min_samples_split	1	20
min_samples_leaf	1	1
max_features	sqrt	sqrt
max_depth	None	24
bootstrap	True	True

Note that, during the initialization phase (i.e. 0th iteration), for each class, E labels are used; following that, every iteration within Algorithm 1, adds $E * 6$ manual labels (by Oracle) from the U_{pred} class. The number 6 denotes the number of classes in the dataset. **Note:** These added samples do not have to be exactly E samples for each class.

8.5 Training Cost Reduction Results

Table 8.3 shows the total number of labels required to attain $\text{micro-F1}_{ac} > \text{micro-F1}$ for various batch sizes E . Note that the total number of labels added would never exceed $20,000 * 6$. micro-F1_{ac} represents the micro-F1 score obtained using active learning methods. Table also includes the computational time required to achieved micro-F1_{ac} .

Consider the following for a better understanding of the results: for method M_{en} , with batch size $E = 50$, the micro-F1_{ac} reached 82.75% after 400 iterations (i.e. after adding 120000 labels.); with batch size $E = 100$, the micro-F1_{ac} was able to attain, the desired micro-F1 84% after 81 iterations (i.e. after adding 48600 labels).

From Table 8.3 one can say that:

For batch sizes $E = 100$ and 200 , M_{md} outperforms M_{en} by 79 and 9 iterations, 2 and 3 compared to 81 and 12, (i.e. 47,400 and 10,800 labels), respectively. This means that for these batch sizes, by using Mahalanobis distance-based selection method, the required number of training samples are reduced by 99.98% and 99.92% while achieving the same level of accuracy as using the complete train set; for batch size $E = 100$ and 200 , M_{en} was able to reduce training samples by 99.15% and 99.73%;

For a batch size $E = 50$, M_{md} is able to achieve 84% micro-F1 in 10 iterations (i.e. 3000 labels) while M_{en} was unable to achieve it; for batch sizes of $E = 10$, even after 2000 iterations, no strategy was able to achieve the same level of accuracy as the full train set; for a batch size of $E = 500$ as the initial training set, the same degree of accuracy as the whole train set was obtained, hence no further labels were added.

Table 8.3: Total number of labels added to reach micro-F1 for different batch sizes E .

Method	batch size E	iterations N	labels added $E * N * 6$	micro-F1 _{ac}	Computational Time
M_{en}	10	2000	120000	72.18	108 hours
	50	400	120000	82.75	21 hours
	100	81	48600	84.03	4 hours
	200	12	14400	84.01	36 minutes
	500	0	0	85.19	NA
M_{md}	10	2000	120000	80.08	388 hours
	50	10	3000	84.24	3 hours
	100	2	1200	84.19	28 minutes
	200	3	3600	84.65	41 minutes
	500	0	0	85.19	NA

8.6 Discussion

This section discusses specific elements of the modeling sampling algorithm and equivalent assessment of presented selection methods.

Training Label Cost: To visually analyse the results, Figures 8.1 and 8.2 illustrates the performance obtained over the test set on each iteration for M_{en} and M_{md} methods for different batch sizes E . (Y axes: micro-F1 over Test-set and X axes: No. of Training Samples.)

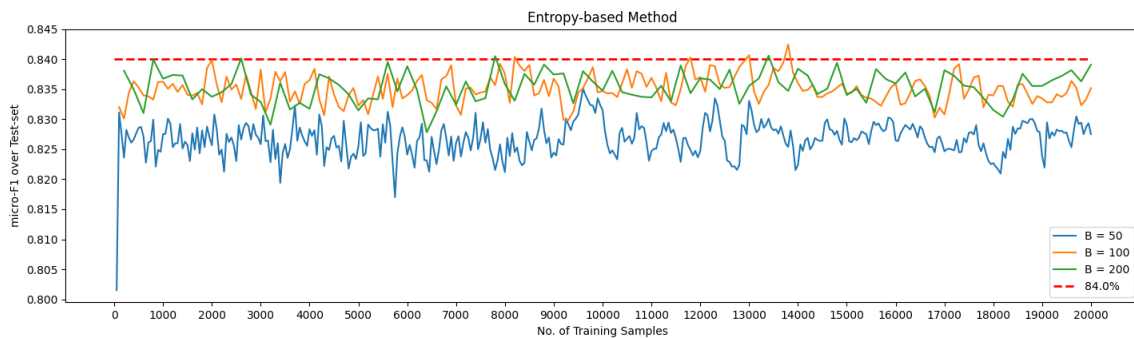


Figure 8.1: A graphical performance of Entropy-based method (M_{en}) based sampling strategy for various batch sizes E .

After analysing Table 8.3, Algorithm 1 and Figures 8.1, 8.2, the following observations and discussions are made:

Statistically (Cohn et al., 1996b), the ‘active learning result curve’ should rise as more informative labels are added, however, this is not the case with Figures 8.1 and 8.2, micro-F1 result curve; Compared to the M_{md} , the M_{en} approach has less variance between the beginning and end of the iterations; A fair explanation would be that for method M_{md} , added points have a greater heterogeneity to test set, also there might be a presence of numerous clusters containing similar information among test set data points; When M_{md} adds a new point to the train set S , which is a heterogeneous point to the test set point, all the related points within test set are misclassified as well, resulting in a greater variance

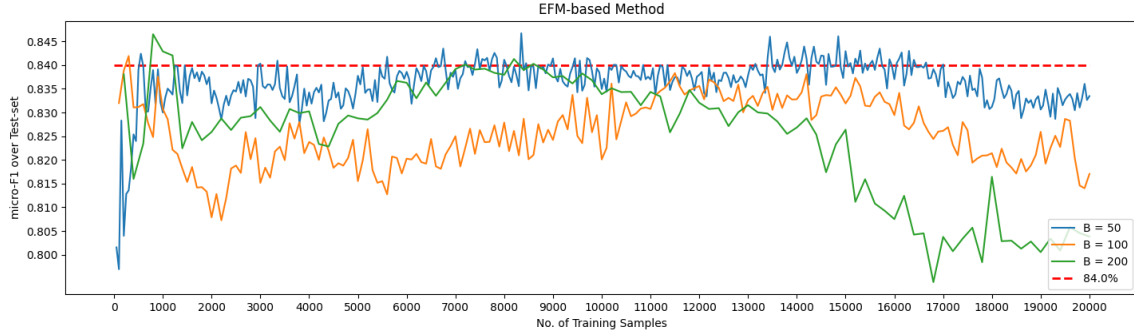


Figure 8.2: A graphical performance of Mahalanobis distance method(M_{md}) based sampling strategy for various batch sizes E .

between the beginning and end of the iteration or dropping of the active learning result curve.

Initial Training Sample Selection: In our experiments, the initial training samples are selected randomly using the `train_test_split` function² of sklearn; we kept `shuffle = True` and a fixed random seed for reproducibility. Also, to observe the effect of initial training sample selection, we kept `Shuffle = False`, meaning the selection is not random, and the grouped E samples from each image are considered for selection. Figure 8.3 shows the performance of both randomly and grouped selection methods for different numbers of initial samples. As can be seen, the random sample selection approach outperforms the grouped one. Furthermore, for random selection, using 250+ samples or more the performance obtained is always above 84%, implying that the initial training sample size does not need to exceed 250+ using random selection for this dataset.

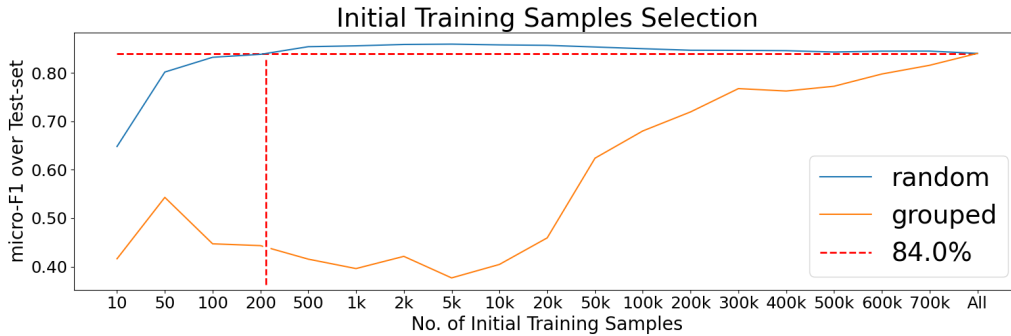


Figure 8.3: Initial training samples selected (random vs grouped).

Myopic vs. Batch Mode Active Learning: In our experiments, we did not evaluate whether the picked points were information redundant or not; irrespective of the method (i.e. M_{en} or M_{md}), the generalized sampling algorithm 1, selects the top E samples without considering their informative relation. The drawback of choosing the top E samples is that some of the selected samples $\hat{E} \subset E$ might provide enough information to the learner regarding remaining samples (i.e. $E \setminus \hat{E}$), leading to redundancy among the selected samples and generating extra labeling. Some of the recent research (Citovsky et al., 2021) uses a clustering step after selecting the top E to diversify and select only \hat{E} samples.

²https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

Batch Size E: According to the Shao et al. (2019), when more samples are chosen at the beginning of the training process, fewer samples may be used in later phases to exploit data recommendations. If more samples were allocated to later iterations, the model would have higher variation in the early iterations but a better chance of biasing samples for active learning in the later rounds. Lourentzou et al. (2018), on the other hand, states that the optimal batch size is determined by the dataset and machine learning application to be addressed.

In our experiments, we kept, in each iteration, a batch size equal to the initial training sample size. The ‘‘Adaptive Batch Mode Active Learning’’ (Chakraborty et al., 2015) was not explored.

Computational vs. Training Label Cost: Table 8.3 show that the M_{en} method has a lower training label cost than M_{md} . In comparison to M_{en} , M_{md} has a lower computational cost. Depending on the nature of the dataset and actual application, one approach may be favored over the other based on batch size E and the trade-off between computational and training label cost.

Reasoning for High Results: In remote sensing, image classification differs from traditional classification problems as the labeled dataset often consist of dense regions. This means that for any given class say ‘Water’, labeled images would have ample similar (high dense) data points representing ‘Water’ pixels. To verify our claim, Figure 8.4 presents the class-wise surface reflectance value distribution of the image scene dataset using the violin plot.

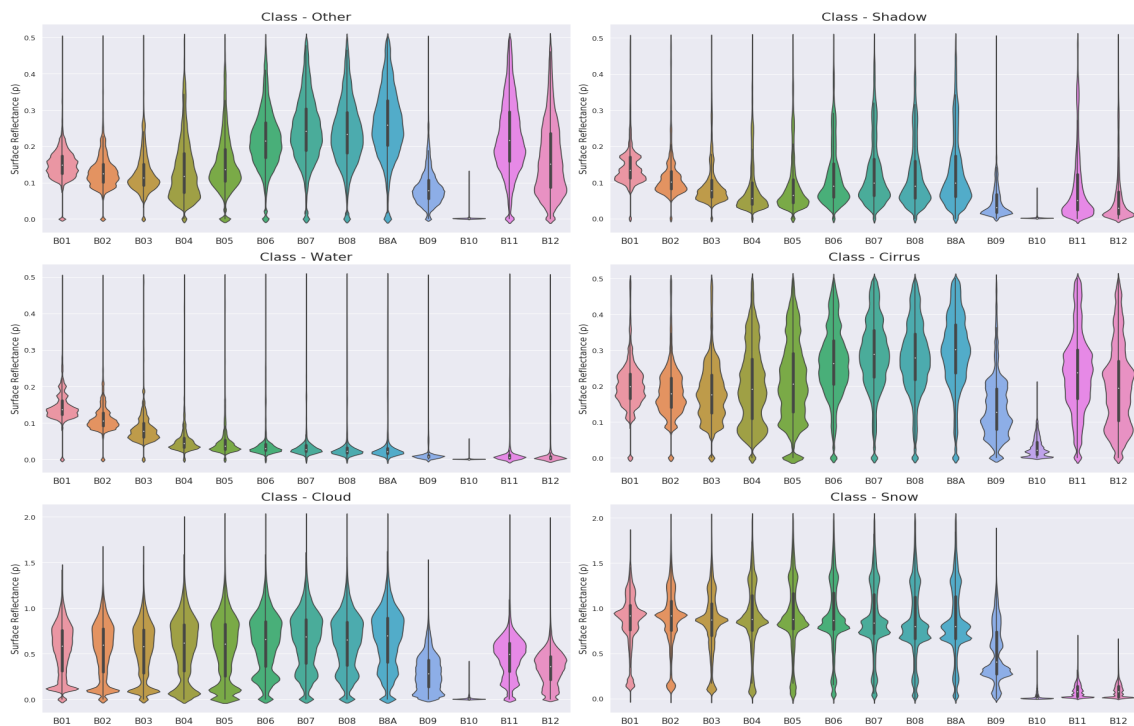


Figure 8.4: Image scene dataset: class-wise surface reflectance value distribution over 13 Bands (Figure 5.3 from Section 5.1).

Here we can observe that for each class and feature, nearly 5.7 million points, the data points are highly dense i.e. the surface reflectance values range with 0.1 difference. Thus, by only selecting a few S samples from the dense region, the proposed learner was able

to achieve the same level of performance as the complete train set. Although, depending upon different datasets and their distribution, this might not be always true.

8.7 Summary

A summary of the Chapter *Abbreviating Train Cost: Modeling and Results* is provided below:

Presents a Generalized Sampling Algorithm 1 for abbreviating training label cost; presents and models an EFM as an query selection method; experimental findings demonstrates that by adopting the proposed methodology, 0.02% of total training samples are required for Sentinel-2 Image Scene Classification while still reaching the same level of accuracy reached by complete train set; highlights the advantages of the proposed method by a comparison with the state-of-the-art entropy based query selection method in active learning.

There are several factors to consider in active learning, including the initial training sample, the batch mode and size, the training label and computational costs, the problem statement and dataset. From this research, one can conclude that using active learning does reduce the overall training label cost, especially when the dataset comprises high-density regions such as multiple pixels in satellite images. Furthermore, the results provides an in-depth comparison of two approaches on the above-mentioned factors, notably the initial training sample selection and the batch size.

Moreover, this chapter expands the usage and proves the utility of Evidence Function Model for query selection and compares its performance in active learning. Finally, for image scene dataset and scene classification problem statement, the Evidence Function Model outperforms the Entropy-based selection approach on reducing training label cost.

8.7.1 Limitation

Being composed of several modules, each of them with a high level of complexity, it is certain that our approach does face below limitations:

At every iteration of the active learning, the EFM must compute the Mahalanobis distance from the train set to remaining data points, in this case, the computation complexity may be considerable high if a large unlabeled dataset is supplied; during experiments, as the initial train set was chosen randomly, information redundancy within the train set might be present; we only addressed pixel-level scene classification in our trials, thus we cannot comment on EFM's performance on object-level classification training cost reduction tasks.

Chapter 9

Conclusions and Future Work

“It seems that perfection is reached not when there is nothing left to add, but when there is nothing left to take away.”

— Antoine de Saint Exupéry

This chapter highlights the key findings in relation to the goals and research questions, as well as assesses their significance and contributions. It also examines the research limitations and suggests topics for further research.

9.1 Conclusions

The overall focus of the doctoral study is on the domains of Active Learning, Uncertainty Prediction, and Earth Observation and their interconnection in the task of image scene classification, misclassification detection, and training label cost reduction.

Over the last decade, substantial advancements in remote sensing technology have enabled us to conduct intelligent Earth Observation such as scene classification using satellite images. The absence of publicly available “big and diverse labeled datasets” of remote sensing images greatly restricts the development of new technologies, particularly using supervised learning methods.

We begin with a detailed analysis of recent developments in the field of remote sensing image scene classification, including Sen2Cor and existing datasets in Chapter 2. By assessing the limitations of these datasets, Chapter 5, presents a surface reflectance-based *image scene* and *waterbody* datasets. Then, using the micro-F1 metric and the introduced image scene datasets, Chapter 6 assesses three ML representative algorithms (Random Forest, Extra Tree, and Convolution Neural Network) for the task of scene classification reaching a performance of 84%, which is significantly higher than Sen2Cor’s 59% value.

The findings in Table 6.5 corroborate our claim that the built ML model may be used as a tool for Sentinel-2 image scene classification. Furthermore, whereas the ML model captures ‘Cloud Shadows’, Sen2Cor misses the majority of them as seen in Figures 6.4 and 6.5. Additionally, supporting our claim, we tested the model’s sensitivity (Figures 6.7 to 6.9)

and biasness (Table 6.6) across multiple LIC images. These results answers RQ1, “Can we provide an ML model that can scene classify any new image, regardless of region, using Sentinel-2 images?”

On the other hand, distances and divergences have recently seen a surge in use in scientific domains such as machine learning. There is, however, a lack of publicly available methods that may use the correlation between the train and test sets to provide insight into the uncertainty of predictions. In light of this, Chapter 4 proposes the generalized Evidence Function Model (EFM) after studying the notion of evidence, the research literature on confidence estimation, and the relationship between the statistical distance and the classifier’s prediction uncertainty. The use of EFM for misclassification detection and a general understanding of how misclassification could be identified for image scene classification is illustrated in Chapter 7: a detailed assessment of the EFM model across multiple datasets is done to quantify classification prediction errors produced by different ML models over Sentinel-2 image scene classification.

For the image scene dataset (Tables 7.7 and 7.8) the overall detection of misclassification was 62.99%, 29.80%, and 31.51% for KNN, ET and CNN models, respectively, leading to a mean micro-F1 of 67.89%, 39.30%, and 38.29% in classifying six classes; for the waterbody dataset (Tables 7.9 and 7.10), the detection of misclassification was 22.27%, 42.08%, and 27.67%, leading to a micro-F1 of 34.70%, 58.96%, and 43.32%, for KNN, ET and CNN models, respectively. Further, over the unseen Sentinel-2 images, the EFM approach was able to identify most of the prediction errors. These findings corroborate our hypothesis that the proposed EFM model may be used to detect misclassification and answer RQ2, “Can we provide an AI model that can detect misclassification for any new data, regardless of the classification algorithm used, without knowledge about new data?”

Chapter 3 states the factors such as the initial training sample, batch mode and size, training label and computation costs, problem objective and dataset that must be taken into account in active learning. According to Chapter 8 findings, active learning reduces the overall cost of labeling particularly when the dataset comprises high-density regions like many pixels in satellite images.

The results of Table 8.3 provide a detailed comparison of two approaches: Entropy-based and Mahalanobis distance-based EFM, with regard to the aforementioned elements, particularly the initial training sample selection and the importance of batch size. The EFM outperforms the Entropy-based approach for image scene classification (and the experimental dataset) in terms of reducing training label cost. Moreover, EFM can be used to identify data points that need to be labeled, answering RQ3, “Can we provide an AL model that can reduce the data required for training classifiers and assist in the generation of new labeled data?”

The stated findings led to the following publications:

1. *Sentinel-2 Image Scene Classification: A Comparison between Sen2Cor and a Machine Learning Approach*, published in 2021 in Remote Sensing 13, no. 2: 300. (DOI: 10.3390/rs13020300). This article presents an image scene dataset made up of 60 Sentinel-2 images where each pixel have been classified into six classes, (Section 5.1) and assesses the built Machine Learning models for image scene classification (Section 6.1).

2. *Mahalanobis distance based accuracy prediction models for Sentinel-2 Image Scene Classification*, published in 2022 in International Journal of Remote Sensing (DOI: 10.1080/01431161.2021.2013575). This article introduces the Waterbody dataset (Section 5.2), the Evidence Function Model (Chapter 4) and the prediction uncertainty identification results for Sentinel-2 image scene classification (Chapter 7).
3. *Sentinel-2 Image Scene Classification over Alentejo Region Farmland*, presented in RECPAD 2020, the 26th Portuguese Conference on Pattern Recognition (pages 43-44). This paper presents a practical use of the ML Sentinel-2 image scene classifier in the detection of ‘Atmospheric Disturbance’ over Alentejo region farmland (Section 6.6).
4. *Sentinel 2 Image Scene Classification: A Comparison Between Bands and Spectral Indices*, presented in RECPAD 2021, the 27th Portuguese Conference on Pattern Recognition (pages 47-48). This paper presents a comparative study between the use of Bands or Spectral Indices information for image scene classification (Section 6.5).
5. *Abbreviating Labelling Cost for Sentinel-2 Image Scene Classification Through Active Learning*, presented in IbPRIA 2022, the 10th Iberian Conference on Pattern Recognition and Image Analysis and published in Lecture Notes in Computer Science, vol 13256, Springer (DOI: 10.1007/978-3-031-04881-4_24). This paper presents how to abbreviate training cost in general, specially for Sentinel-2 image scene classifiers, and compares Entropy-based and EFM-based methods (Chapter 8).
6. *A ML approach for scene classification using Sentinel-2 images*, an oral presentation made in 2022 in the 1st Copernicus National Conference¹. This publication presents the need to have an ML approach for scene classification (RQ1) and compares ML vs Sen2Cor results (Section 6.3).

9.2 Future Work

The present research is made up of multiple, highly intricate modules; undoubtedly, our approach could be improved and the overall performance might be increased. In addition to strengthening the various modules, further improvements are possible, such as:

Concerning generated datasets (Chapter 5), add Sentinel-1 image bands to enhance the observations and connect the findings as well as their effects on the detection of Water, Shadow, Cirrus, Cloud, and Snow classes; detect and delete near-duplicates from the datasets that do not contribute/have an influence on the classifier; add new training scenarios using current training data and image augmentation, also known as elastic transformation [Gabrani and Tretiak \(1996\)](#).

Concerning the proposed Mahalanobis distance based Evidence Function Model (Chapter 4), calculate the distance between the nearest and furthest point in the distribution and compare the findings to the mean distance, as well as the influence on parallel coordinates visualization; apply different weights to Mahalanobis distances based on the actual and predicted classes.

¹<https://www.copernicus.eu/en/events/events/portugal-copernicus-national-conference>

Concerning the findings of the EFM model in identifying the prediction uncertainty for Sentinel-2 image scene classification (Chapter 7), according to the results in Tables 7.8 and 7.10, one can still improve the EFM results by lowering the False Positive values. To do so, a potential argument was raised that ‘clustering the distribution into smaller subsets while creating different data splits could further increase the knowledge acquired by the distance in terms of true prediction vs misclassification can help in lowering the False Positive errors’. Further clustering and increasing the number of Mahalanobis distances might also assist to decrease model bias by having multiple reference points as cluster centroids.

Concerning the active learning baseline specifications (Chapter 3) and their impact on modeling training cost reduction model (Chapter 8), investigate appropriate initial training set size and sample selection strategy; achieve low information redundancy within a batch of selected examples on active learning iterations; consider fixed vs. variable batches while looking for the appropriate batch size; investigate the usage of a mix of Entropy-based and Mahalanobis distance-based approaches using the Generalized Sampling Algorithm 1; broaden the study and apply the trials to new datasets.

Appendix A

Supporting Material

A.1 Spectral Indices

Chlorophyll Index (CI)
(GREEN: 520-600 NIR: 760-900)

$$CI = \frac{NIR}{GREEN} - 1 \quad (A.1)$$

Effective Leaf Area Index (ELAI)
(RED: 610-680 NIR: 780-890)

$$ELAI = -0.441 + 0.285 \frac{NIR}{RED} \quad (A.2)$$

Green Normalised Difference Vegetation Index (GNDVI)
(GREEN: 557-582 NIR: 720-920)

$$GNDVI = \frac{NIR - GREEN}{NIR + GREEN} \quad (A.3)$$

Modified Soil Adjusted Vegetation Index (MSAVI2)
(RED: 630-690 NIR: 760-860)

$$MSAVI2 = \frac{2NIR + 1 - \sqrt{(2NIR + 1)^2 - 8(NIR - RED)}}{2} \quad (A.4)$$

Normalised Difference Infrared Index (NDII)
(NIR: 845-885 SWIR: 1650-170)

$$NDII = \frac{NIR - SWIR}{NIR + SWIR} \quad (A.5)$$

Normalised Difference Water Index (NDWI)

(NIR: 841-876 SWIR: 1230-1250)

$$NDWI = \frac{NIR - SWIR}{NIR + SWIR} \quad (A.6)$$

Normalised Pigment Chlorophyll Ratio Index (NPCI)

(BLUE: 460 RED: 660)

$$NPCI = \frac{RED - BLUE}{RED + BLUE} \quad (A.7)$$

Relative Reflectance Index (RRI)

(VIS: 400-700 NIR: 740-820)

$$RRI = \frac{NIR/VIS}{NIR/VIS} \quad (A.8)$$

Short wave Infrared Water Stress Index (SIWSI)

(NIR: 841-876 SWIR: 1628-1652)

$$SIWSI = \frac{SWIR - NIR}{SWIR + NIR} \quad (A.9)$$

Triangular Greenness Index (TGI)

(BLUE: 450-520 GREEN: 520-600)

$$TGI = -0.5[(RED - BLUE)(RED - GREEN) - (RED - GREEN)(RED - BLUE)] \quad (A.10)$$

Normalized Difference Vegetation Index (NDVI)

(NIR: 841-876 RED: 660)

$$NDVI = \frac{NIR - RED}{NIR + RED} \quad (A.11)$$

Reflectance ratio (R)

(NIR: 841-876 GREEN: 520-600)

$$R = \frac{NIR}{GREEN} \quad (A.12)$$

Reflectance ratio ($R1$)

(BLUE: 460 SWIR: 1230-1250)

$$R1 = \frac{BLUE}{SWIR} \quad (A.13)$$

Sentinel-2 Water Index (SWI)

(VNIR: 705 SWIR: 1610)

$$SWI = \frac{VNIR - SWIR}{VNIR + SWIR} \quad (A.14)$$

Normalized Difference Snow Index (NDSI)
(Green: 560 SWIR: 1610)

$$NDSI = \frac{Green - SWIR}{Green + SWIR} \quad (A.15)$$

Normalized Difference Snow Index 2 (NDSII)
(Red: 665 SWIR: 1610)

$$NDSII = \frac{Red - SWIR}{Red + SWIR} \quad (A.16)$$

S3
(Red: 665 VNIR: 842 SWIR: 1610)

$$S3 = \frac{VNIR \star (Red - SWIR)}{(Red + VNIR) \star (VNIR + SWIR)} \quad (A.17)$$

Snow Water Index (SWI)
(Green: 560 VNIR: 842 SWIR: 1610)

$$SWI = \frac{Green \star (VNIR - SWIR)}{(Green + VNIR) \star (VNIR + SWIR)} \quad (A.18)$$

Shadow Enhancement Index (SEI)
(Ultra Blue: 443 Green: 560 VNIR: 842 SWIR: 940)

$$SEI = \frac{(UltraBlue + SWIR) - (Green - VNIR)}{(UltraBlue + SWIR) + (Green - VNIR)} \quad (A.19)$$

Normalized Saturation Value Different Index (NSVDI),
Where, $V = \max(Red, VNIR, SWIR)$ and $S = (V - \min(Red, VNIR, SWIR))/V$.
(Red: 665 VNIR: 842 SWIR: 2190)

$$NSVDI = \frac{S - V}{S + V} \quad (A.20)$$

Cloud Index (CI)
(Blue: 490 Green: 560 Red: 665 VNIR: 842 SWIR: 1610)

$$CI = \frac{VNIR + (2 \star SWIR)}{(Blue + Green + Red)} \quad (A.21)$$

Brightness Index (BI)

(Blue: 490 Green: 560 Red: 665 VNIR: 842 SWIR: 1375 SWIR: 2190)

$$BI = 0.30 \star Blue + 0.27 \star Green + 0.47 \star Red \\ + 0.55 \star VNIR + 0.50 \star SWIR + 0.18 \star SWIR \quad (\text{A.22})$$

Bare Soil Index (BSI)

(Blue: 490 Red: 665 VNIR: 842 SWIR: 1610)

$$BSI = \frac{(Red + SWIR) - (Blue + VNIR)}{(Red + SWIR) + (Blue + VNIR)} \quad (\text{A.23})$$

A.2 Lemma

Lemma A.2.1. *The Mahalanobis Distance can be used to measure the distance from a point to a multivariable distribution specified by its mean vector and covariance matrix (Danielsson, 1980; Darmochwał, 1991; Barhen and Daudin, 1995; McLachlan, 1999; De Maesschalck et al., 2000).*

Proof. The Mahalanobis Distance between point \mathbf{x}_i and a distribution with mean $\boldsymbol{\mu}$ and covariance matrix Σ is given by Equation (A.24).

$$\Delta = \sqrt{(\mathbf{x}_i - \boldsymbol{\mu})\Sigma^{-1}(\mathbf{x}_i - \boldsymbol{\mu})^\top} \quad (\text{A.24})$$

To compute the MD, you must first generate the variance-covariance matrix Σ_x :

$$\Sigma_x = \frac{1}{(n-1)}(X_c)^T(X_c) \quad (\text{A.25})$$

where X is the data matrix with n objects in the rows measured for p variables. The column-centred data matrix $(X - \bar{X})$ is denoted by X_c . The variance-covariance matrix for two variables, x_1 and x_2 , is

$$\Sigma_x = \begin{bmatrix} \sigma_1^2 & \rho_{12} \star \sigma_1 \star \sigma_2 \\ \rho_{12} \star \sigma_1 \star \sigma_2 & \sigma_2^2 \end{bmatrix} \quad (\text{A.26})$$

where σ_1^2 and σ_2^2 are the variances of the values of, respectively, the first and second variable; $\rho_{12} \star \sigma_1 \star \sigma_2$ is the covariance between the two variables.

Within Equation A.24, Σ_x^{-1} can be substituted for an object x_i as:

$$\Sigma_x^{-1} = \begin{bmatrix} \sigma_1^2/\det(\Sigma_x) & -(\rho_{12} \star \sigma_1 \star \sigma_2)/\det(\Sigma_x) \\ -(\rho_{12} \star \sigma_1 \star \sigma_2)/\det(\Sigma_x) & \sigma_2^2/\det(\Sigma_x) \end{bmatrix} \quad (\text{A.27})$$

where $\det(\Sigma_x) = \sigma_1^2 * \sigma_2^2 * (1 - \rho_{12}^2)$ is the determinant of the variancecovariance matrix.

For an object x_i measured in two variables, x_1 and x_2 , Equation A.24 and A.27 can be rewritten, since

$$\begin{aligned}
& [(x_1 - \bar{x}) * (x_2 - \bar{x})] \Sigma_x^{-1} = \\
& \left[\frac{(\sigma_2^2 * (x_1 - \bar{x})) - ((x_2 - \bar{x}) * (\rho_{12} * \sigma_1 * \sigma_2))}{\det(\Sigma_x)} \quad \frac{(\sigma_1^2 * (x_2 - \bar{x})) - ((x_1 - \bar{x}) * (\rho_{12} * \sigma_1 * \sigma_2))}{\det(\Sigma_x)} \right] \\
& [(x_1 - \bar{x}) * (x_2 - \bar{x})] \Sigma_x^{-1} \begin{bmatrix} (x_1 - \bar{x}) \\ (x_2 - \bar{x}) \end{bmatrix} \\
= & \left[\frac{(\sigma_2^2 * (x_1 - \bar{x})^2) - ((x_1 - \bar{x}) * (x_2 - \bar{x}) * (\rho_{12} * \sigma_1 * \sigma_2))}{\det(\Sigma_x)} + \frac{(\sigma_1^2 * (x_2 - \bar{x})^2) - ((x_1 - \bar{x}) * (x_2 - \bar{x}) * (\rho_{12} * \sigma_1 * \sigma_2))}{\det(\Sigma_x)} \right] \\
= & \left[\frac{(\sigma_2^2 * (x_1 - \bar{x})^2 * (1 - \rho_{12}^2)) + (\sigma_1^2 * (x_2 - \bar{x})^2) - 2((x_1 - \bar{x}) * (x_2 - \bar{x}) * (\rho_{12} * \sigma_1 * \sigma_2)) + (\sigma_2^2 * (x_1 - \bar{x}) \rho_{12}^2)}{\sigma_1 * \sigma_2 (1 - \rho_{12}^2)} \right] \\
= & \left[\frac{(x_1 - \bar{x})^2}{\sigma_1^2} + \frac{(x_2 - \bar{x})^2}{\sigma_2^2 (1 - \rho_{12}^2)} - 2 \frac{(x_1 - \bar{x})(x_2 - \bar{x}) \rho_{12}}{\sigma_1 \sigma_2 (1 - \rho_{12}^2)} + \frac{\rho_{12}^2 (x_1 - \bar{x})^2}{\sigma_1^2 (1 - \rho_{12}^2)} \right] \\
= & \frac{(x_1 - \bar{x})^2}{\sigma_1^2} + \left(\frac{(x_2 - \bar{x})}{\sigma_2 \sqrt{1 - \rho_{12}^2}} - \frac{\rho_{12} (x_1 - \bar{x})}{\sigma_1 \sqrt{1 - \rho_{12}^2}} \right)^2
\end{aligned} \tag{A.28}$$

Leading to MD_i be

$$MD_i = \sqrt{\left(\frac{x_{i1} - \bar{x}_1}{\sigma_1} \right)^2 + \left[\left\{ \left(\frac{x_{i2} - \bar{x}_2}{\sigma_2} \right) - \rho_{12} \left(\frac{x_{i1} - \bar{x}_1}{\sigma_1} \right) \right\} \frac{1}{\sqrt{1 - \rho_{12}^2}} \right]^2} \tag{A.29}$$

The component of the second variable that is already explained by the first variable is deducted in this formulation. To put it another way, the MD rectifies for data correlation by using ρ_{12} . Thus, Equation A.29 is simplified to the formula for the Euclidean Distance (ED) when the independent quantities are uncorrelated (i.e. $\rho_{12} = 0$) resulting to Equation A.30.

$$ED_i = \sqrt{(x_{i1} - \bar{x}_1)^2 + (x_{i2} - \bar{x}_2)^2} \tag{A.30}$$

□

A.3 Sensor

Table A.1 presents the satellite sensors and their reference.

Table A.1: Sensor and Reference.

Sensor	Reference
Sentinel 1	European Space Agency (2022c)
Sentinel 2	European Space Agency (2022d)
Sentinel 3	European Space Agency (2022e)
Sentinel 4	European Space Agency (2022f)
Sentinel 5	European Space Agency (2022g)
Sentinel 6	European Space Agency (2022h)
QuickBird	Satellite Imaging Corporation (2022b)
CIMEL 313	Cuevas et al. (2019)
Terra ASTER	NASA Terra, The EOS Flagship (2022)
MODIS	NASA Modis (2022)
LI-190s and LI-220S	Li-Cor (2022)
Airborne AVIRIS	Johnson and Green (1995)
Landsat TM	NASA Landsat Science (2013)
Landsat-8 OLI	NASA Landsat Missions (2022)
EnMAP	European Space Agency (2022b)
ROSIS-03	Kunkel et al. (1988)
Hyper-spectral Electro-optic	Corning (2022)
SIPPER II	Luo et al. (2005)
LIDAR	Survey (2022)
MapSwipe	MapSwipe (2022)
IRS-1A	Department of Space (2022)
IKONOS	Satellite Imaging Corporation (2022a)
AISA Eagle	Aisa Systems (2022)

A.4 Image-wise Class Value Distribution

Table A.2 details Train set (50) and Table A.3 details Test set (10) image wise class value Distribution.

Table A.2: Train set (50) Products Wise Class Value Distribution.

Product ID	Cloud	Cirrus	Shadow	Snow	Water	Other	Total
R069_V20151204T171502_20151204T171502	0	0	1369	21163	20667	8184	51383
R022_V20151211T102944_20151211T102944	41292	0	34033	0	17744	34140	127209
R010_V20160109T143825_20160109T143825	3836	0	11129	173217	54486	25447	268115
R093_V20151206T093115_20151206T093115	19740	0	35984	4668	48889	27266	136547
R108_V20160624T103721_20160624T103721	8891	0	0	9918	23773	30334	72916
R065_V20161108T102232_20161108T102232	30839	0	2975	1873	1572	30092	67351
R090_V20151206T043239_20151206T043239	7922	0	41461	42298	0	8790	100471
R105_V20151207T054131_20151207T054131	8047	5832	32057	67291	0	16544	129771
R135_V20151209T080737_20151209T080737	0	0	18318	11113	12418	15223	50722
R116_V20151228T002843_20151228T002843	22307	0	20814	0	14948	32196	90265
R013_V20160109T191435_20160109T191435	2000	6708	11863	23371	26763	13190	83895
R094_V20151216T111216_20151216T111216	15385	58309	29970	0	0	31259	134923
R110_V20151227T142837_20151227T142837	10075	0	15176	41289	29485	23575	119600
R038_V20160210T130341_20160210T130341	6563	9115	29431	77843	33476	1389	157817
R046_V20160112T025031_20160112T025031	56396	42625	35356	0	29995	34224	198596
R137_V20151209T112253_20151209T112253	11769	37276	18106	0	10482	15630	93263
R135_V20160815T074942_20160815T081315	215	0	232	0	0	52783	53230
R079_V20160831T095032_20160831T095217	12374	19592	1293	0	2783	8477	44519
R092_V20151206T080705_20151206T080705	110291	2011	18611	0	5973	30716	167602
R065_V20161029T102132_20161029T102132	681	8447	0	8371	4998	41986	64483
R122_V20160327T100012_20160327T100012	0	11988	0	1839	0	4607	18434
R127_V20151218T183704_20151218T183704	0	16406	12111	45919	0	11699	86135
R122_V20151208T101125_20151208T101125	7109	62855	37488	0	13204	32501	153157
R021_V20151211T084342_20151211T084342	13089	8880	40619	0	11901	32041	106530
R022_V20160419T101026_20160419T101026	1993	0	0	1038	2124	40659	45814
R065_V20151224T103329_20151224T103329	4051	31403	17304	0	34878	38120	125756
R053_V20160122T144141_20160122T144141	0	0	5197	42458	16383	26318	90356
R092_V20151226T080933_20151226T080933	79093	72124	23259	0	40203	35347	250026
R135_V20151209T080737_20151209T080737	0	105076	64809	0	81956	164017	415858
R137_V20160417T111159_20160417T111159	6388	135066	0	626	0	19744	161824
R044_V20160220T230557_20160220T230557	0	8761	9826	38722	19384	0	76693
R054_V20160102T161125_20160102T161125	54049	1851	31241	0	118705	78638	284484
R065_V20160422T102025_20160422T102025	7552	26263	0	2903	9099	54720	100537
R103_V20160126T023520_20160126T023520	24653	0	23285	0	22524	24878	95340
R027_V20161115T183632_20161115T183632	0	0	0	0	0	38859	38859
R103_V20160116T023225_20160116T023225	33542	23636	24978	0	14888	28985	126029
R135_V20160217T075949_20160217T075949	38719	0	16652	0	17548	28607	101526
R065_V20160323T102143_20160323T102143	17977	0	0	4686	21838	86062	130563
R137_V20151209T112253_20151209T112253	36185	40153	67931	0	70179	46701	261149
R092_V20160204T075210_20160204T075210	32877	5390	14624	1127	4139	45573	103730
R135_V20151229T081422_20151229T081422	45199	0	11514	0	23554	28435	108702
R027_V20151231T184606_20151231T184606	0	0	6633	0	26864	10815	44312
R053_V20151223T144214_20151223T144214	25493	12514	11810	25732	0	60942	136491
R030_V20160121T000856_20160121T000856	15870	9961	29524	0	33103	25148	113606
R035_V20160210T080716_20160210T080716	3333	2969	6488	49935	4861	2987	70573
R092_V20160921T073612_20160921T080338	0	0	0	0	802	21602	22404
R135_V20160207T081608_20160207T081608	75883	6530	16714	0	23600	46415	169142
R108_V20160205T103556_20160205T103556	3640	5783	3712	27263	4227	4220	48845
R065_V20161108T102232_20161108T102232	901	0	1781	1663	0	0	4345
R022_V20160310T101207_20160310T101207	1285	3111	0	1686	0	0	6082
total	897504	780635	835678	728012	954416	1520085	5716330

Table A.3: Test set (10) Products Wise Class Value Distribution.

Product ID	Cloud	Cirrus	Shadow	Snow	Water	Other	Total
R096_V20160105T144841_20160105T144841	31474	99669	36053	0	0	20142	187338
R022_V20160817T101032_20160817T101559	2893	8277	2092	0	1263	8414	22939
R122_V20160506T100226_20160506T100226	0	21277	0	0	2102	1345	24724
R137_V20160217T111843_20160217T111843	3671	1048	33292	68929	42214	59632	208786
R078_V20151225T083056_20151225T083056	37024	14838	13731	0	5767	22812	94172
R051_V20151203T110846_20151203T110846	8657	1841	21142	0	21545	18897	72082
R112_V20160215T171407_20160215T171407	0	7873	5874	41049	8989	0	63785
R105_V20151207T054131_20151207T054131	7884	17597	4750	5460	8531	11417	55639
R135_V20151219T080616_20151219T080616	1775	3568	3740	39313	6194	5937	60527
R072_V20160123T223141_20160123T223141	40937	0	35041	0	20405	25773	122156
	134315	175988	155715	154751	117010	174369	912148

Appendix B

Sentinel-2 Image Scene Classification Package

Through this doctoral study, the following resource is made publicly available ([Raiyani et al., 2021](#)): a ready to use Python package (scripts) with a trained ML model to classify Sentinel-2 L1C image. The Python package takes the L1C product path and produces an RGB image with six classes (Cloud, Cirrus, Shadow, Snow, Water, and Other) at 20m resolution. The working example of the developed Sentinel-2 L1C image scene classification package is discussed further.

Figure [B.1](#) shows the processing steps of the developed package. The path to Sentinel-2 L1C product is passed as input, and a RGB image with six colors (each identifying one class) at 20m resolution is produced as output. The GDAL library ([GDAL/OGR contributors, 2020](#)) was used to read and rescale images, and during post-processing, neighbour pixels are checked to minimize the classification error.

Figure [B.2](#) shows the working example of the developed package where, L1C product is classified into six classes. Figure [B.2a,b](#) respectively present the corresponding RGB image of L1C product and classified image. Using our package the average time to produce a scene classified RGB image is 4 min; using Sen2Cor v2.5.5 takes 18 min over system specification detailed in [Table 7.3](#) (it is worth mentioning that Sen2Cor performs many other operations apart from scene classification). For the sole purpose of scene classification, our model is 4 times faster than Sen2Cor when classifying Sentinel-2 L1C images into six classes (Cloud, Cirrus, Shadow, Snow, Water, and Other).

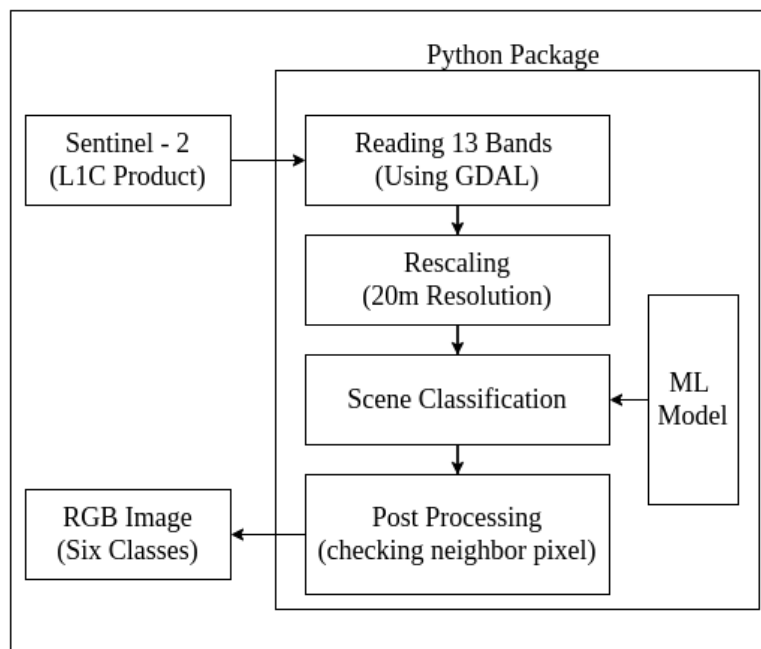
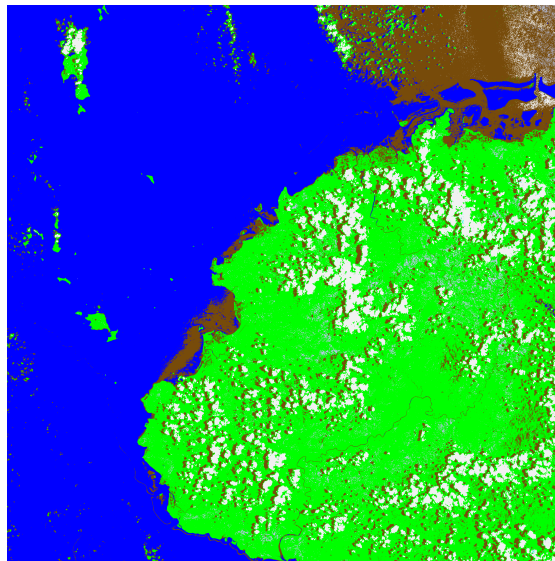


Figure B.1: Package Processing Steps: Classifying Sentinel-2 L1C Product.



(a)



(b)

Figure B.2: (a) L1C product (b) RGB Scene classified image using developed package. Labels—Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan and Other as Green.

Appendix C

EFM Waterbody dataset results

Further, to visually analyze the waterbody misclassification vs misclassification detection, randomly 16/49 water-bodies RGB images are shown in Figure C.1.

From the Figure, we can observe:

1. For KNN, images 1, 9, and 10 had complete misclassifications detected, images 2, 5, and 15 had partial misclassifications detected, image 14 had no misclassification detected, making it a 100% false negative, and image 4 had no misclassification but was identified as having a 100% (as a false positive) classification error. Images 3, 7, 8, 11, 12, 13, and 16 were perfectly identified with no false positives.
2. For ET, images 1, 4, 5, 9, and 10 had complete misclassifications detected, image 15 had partial misclassifications detected, images 2, 6 and 14 had no misclassification detected, making it a 100% false negative. Images 3, 7, 8, 11, 12, 13, and 16 were perfectly identified with no false positives.
3. For CNN, images 4, 11, and 14 had complete misclassifications detected, images 1, 2, and 5 had partial misclassifications detected, images 6 and 15 had no misclassification detected, making it a 100% false negative. Images 3, 7, 8, 9, 10, 12, 13, and 16 were perfectly identified with no false positives.

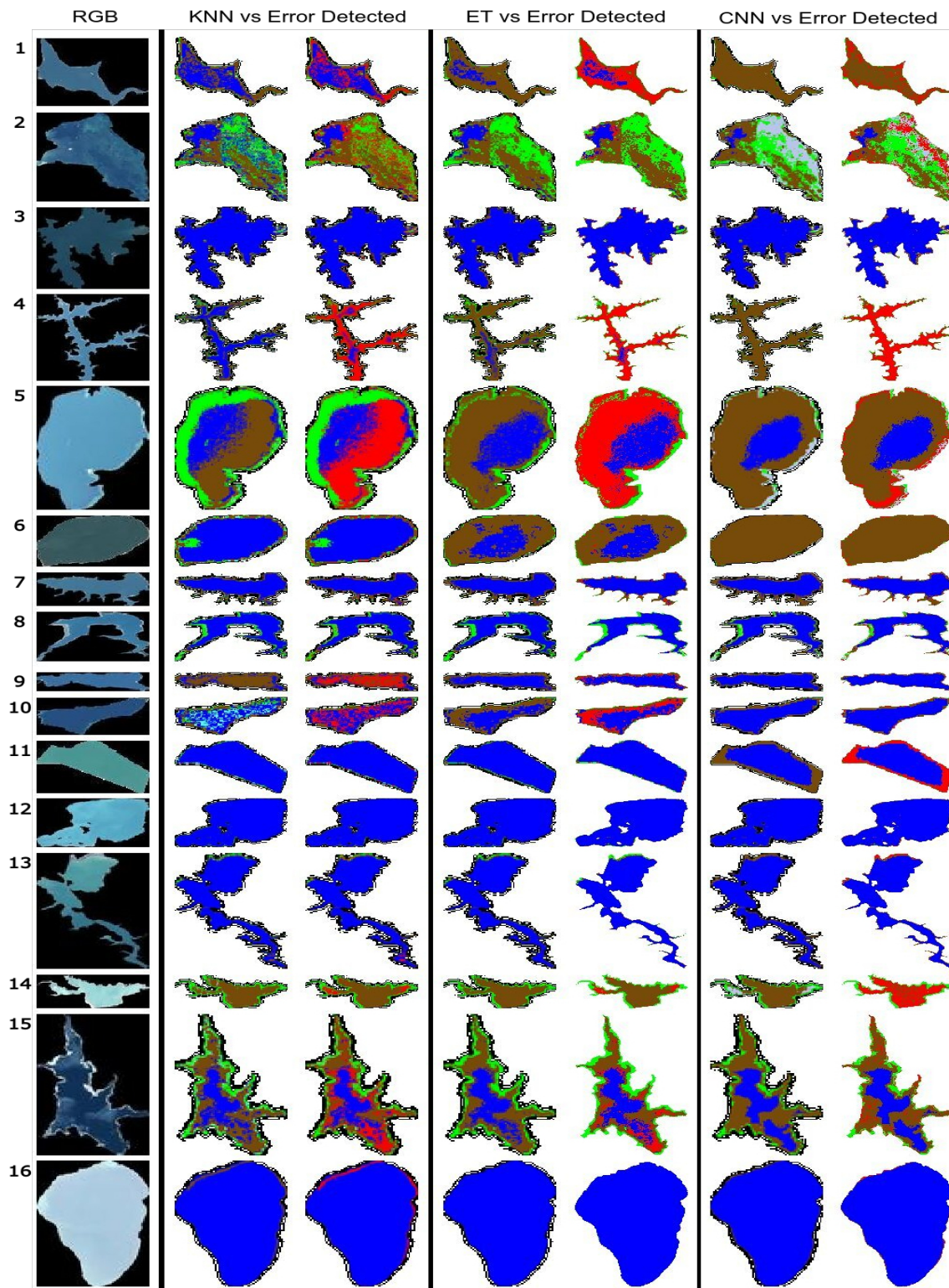


Figure C.1: Water Body RGB image followed by Classified and Error Detected image for KNN, ET and, CNN. Color Labels—Other as Green, Water as Blue, Shadow as Brown, Cirrus as light Purple, Cloud as White, Snow as Cyan and, Error as Red.

Bibliography

- Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U. R., et al. (2021). A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297.
- Abdel-Hamid, O., Deng, L., and Yu, D. (2013). Exploring convolutional neural network structures and optimization techniques for speech recognition. In *Interspeech*, volume 11, pages 73–5.
- Abe, N., Mamitsuka, H., and Nakamura, A. (1998). Empirical comparison of competing query learning methods. In *International Conference on Discovery Science*, pages 387–388. Springer.
- Adam, E., Mutanga, O., and Rugege, D. (2010). Multispectral and hyperspectral remote sensing for identification and mapping of wetland vegetation: a review. *Wetlands Ecology and Management*, 18(3):281–296.
- Agroinsider (2022). Environmental and agricultural sustainability. <https://agroinsider.com/home>. Accessed: 27 06 2022.
- Ahmad, M., Protasov, S., Khan, A. M., Hussain, R., Khattak, A. M., and Khan, W. A. (2018). Fuzziness-based active learning framework to enhance hyperspectral image classification performance for discriminative and generative classifiers. *PloS one*, 13(1):e0188996.
- Aisa Systems (2022). Aisa eagle hyperspectral sensor. <https://www.adept.net.au/cameras/specim/systems/pdf/AisaEAGLE.pdf>. Accessed: 06 04 2022.
- Akar, Ö. and Güngör, O. (2012). Classification of multispectral images using random forest algorithm. *Journal of Geodesy and Geoinformation*, 1(2):105–112.
- Al-Obeidat, F., Al-Taani, A. T., Belacel, N., Feltrin, L., and Banerjee, N. (2015). A fuzzy decision tree for processing satellite images and landsat data. *Procedia Computer Science*, 52:1192–1197.
- Albert, A., Kaur, J., and Gonzalez, M. C. (2017). Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1357–1366.
- Alzubi, J., Nayyar, A., and Kumar, A. (2018). Machine learning from theory to algorithms: An overview. *Journal of Physics: Conference Series*, 1142:012012.

- Angluin, D. (1988). Queries and concept learning. *Machine Learning*, 2(4):319–342.
- Anwar, S. (2022). Introducing pearl ai accelerated land cover mapping platform. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html. Accessed: 06 04 2022.
- Anyamba, A., Tucker, C. J., and Eastman, J. R. (2001). Ndvi anomaly patterns over africa during the 1997/98 enso warm event. *International Journal of Remote Sensing*, 22(10):1847–1860.
- Atlas, L., Cohn, D., and Ladner, R. (1989). Training connectionist networks with queries and selective sampling. *Advances in neural information processing systems*, 2.
- Attenberg, J., Ipeirotis, P., and Provost, F. (2015). Beat the machine: Challenging humans to find a predictive model’s unknown unknowns. *Journal of Data and Information Quality (JDIQ)*, 6(1):1–17.
- Atzberger, C. (2013). Advances in remote sensing of agriculture: Context description, existing operational monitoring systems and major information needs. *Remote sensing*, 5(2):949–981.
- Azpiroz, I., Oses, N., Quartulli, M., Olaizola, I. G., Guidotti, D., and Marchi, S. (2021). Comparison of climate reanalysis and remote-sensing data for predicting olive phenology through machine-learning methods. *Remote Sensing*, 13(6):1224.
- Babaeian, E., Sadeghi, M., Jones, S. B., Montzka, C., Vereecken, H., and Tuller, M. (2019). Ground, proximal, and satellite remote sensing of soil moisture. *Reviews of Geophysics*, 57(2):530–616.
- Bach, N. and Badaskar, S. (2007). A review of relation extraction. *Literature review for Language and Statistics II*, 2:1–15.
- Baetens, L., Desjardins, C., and Hagolle, O. (2019). Validation of copernicus sentinel-2 cloud masks obtained from maja, sen2cor, and fmask processors using reference cloud masks generated with a supervised active learning procedure. *Remote Sensing*, 11(4).
- Baldrige, J. and Osborne, M. (2004). Active learning and the total cost of annotation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 9–16.
- Bandara, K. M. P. S. (2003). Monitoring irrigation performance in sri lanka with high-frequency satellite measurements during the dry season. *Agricultural water management*, 58(2):159–170.
- Baram, Y., Yaniv, R. E., and Luz, K. (2004). Online choice of active learning algorithms. *Journal of Machine Learning Research*, 5(Mar):255–291.
- Barhen, A. and Daudin, J. (1995). Generalization of the mahalanobis distance in the mixed case. *Journal of Multivariate Analysis*, 53(2):332–342.
- Barrett, E. C. (2013). *Introduction to environmental remote sensing*. Routledge.
- Batra, N. et al. (2006). Estimation and comparison of evapotranspiration from modis and avhrr sensors for clear sky days over the southern great plains. *Remote Sensing of Environment*, 103(1):1–15.

- Bausch, W. C. and Khosla, R. (2010). Quickbird satellite versus ground-based multi-spectral data for estimating nitrogen status of irrigated maize. *Precision Agriculture*, 11(3):274–290.
- Bayarjargal, Y., Karnieli, A., Bayasgalan, M., Khudulmur, S., Gandush, C., and Tucker, C. (2006). A comparative study of noaa-avhrr derived drought indices using change vector analysis. *Remote Sensing of Environment*, 105(1):9–22.
- Belgiu, M. and Drăguț, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, 114:24–31.
- Bell, R. J. (1923). *An elementary treatise on coordinate geometry of three dimensions*. Macmillan.
- Bernard, J., Hutter, M., Zeppelzauer, M., Fellner, D., and Sedlmair, M. (2017). Comparing visual-interactive labeling with active learning: An experimental study. *IEEE transactions on visualization and computer graphics*, 24(1):298–308.
- Bernard, S., Heutte, L., and Adam, S. (2009). On the selection of decision trees in random forests. In *2009 International Joint Conference on Neural Networks*, pages 302–307. IEEE.
- Bezdek, J. C. (2013). *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media.
- Bhandari, M. (2012). International centre for integrated mountain development. *The Wiley-Blackwell Encyclopedia of Globalization*.
- Bhardwaj, A., Sam, L., Bhardwaj, A., and Martín-Torres, F. J. (2016). Lidar remote sensing of the cryosphere: Present applications and future prospects. *Remote Sensing of Environment*, 177:125–143.
- Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern recognition and machine learning*, volume 4. Springer.
- Bisht, P., Kumar, P., Yadav, M., Rawat, J., Sharma, M., and Hooda, R. (2014). Spatial dynamics for relative contribution of cropping pattern analysis on environment by integrating remote sensing and gis. *International Journal of Plant Production*, 8(1):1–17.
- Blackmore, S., Godwin, R. J., and Fountas, S. (2003). The analysis of spatial and temporal trends in yield map data over six years. *Biosystems engineering*, 84(4):455–466.
- Blaschke, T., Burnett, C., and Pekkarinen, A. (2004). Image segmentation methods for object-based analysis and classification. In *Remote sensing image analysis: Including the spatial domain*, pages 211–236. Springer.
- Blaschke, T. and Strobl, J. (2001). Whats wrong with pixels? some recent developments interfacing remote sensing and gis. *Zeitschrift für Geoinformationssysteme*, pages 12–17.
- Bowker, A. H. (1948). A test for symmetry in contingency tables. *Journal of the american statistical association*, 43(244):572–574.
- Bramley, R. G. V. (2009). Lessons from nearly 20 years of precision agriculture research, development, and adoption as a guide to its appropriate application. *Crop and Pasture Science*, 60(3):197–217.

- Bredemeier, C. and Schmidhalter, U. (2005). Laser-induced chlorophyll fluorescence sensing to determine biomass and nitrogen uptake of winter wheat under controlled environment and field conditions. *Precision Agriculture* :, pages 273–280.
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.
- Brinker, K. (2003). Incorporating diversity in active learning with support vector machines. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 59–66.
- Brownlee, J. (2018). How to develop 1d convolutional neural network models for human activity recognition. URL: <https://machinelearningmastery.com/cnn-models-for-human-activityrecognition-time-series-classification/>.(accessed: 02.03. 2020).
- Budd, S., Robinson, E. C., and Kainz, B. (2021). A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical Image Analysis*, 71:102062.
- Burbidge, R., Rowland, J. J., and King, R. D. (2007). Active learning for regression based on query by committee. In *International conference on intelligent data engineering and automated learning*, pages 209–218. Springer.
- Calvaio, T. and Palmeirim, J. M. (2004). Mapping mediterranean scrub with satellite imagery: biomass estimation and spectral behaviour. *International Journal of Remote Sensing*, 25(16):3113–3126.
- Camagni, R. and Capello, R. (2017). Regional innovation patterns and the eu regional policy reform: towards smart innovation policies. In *Seminal studies in regional and urban economics*, pages 313–343. Springer.
- Campbell, C., Cristianini, N., Smola, A., et al. (2000). Query learning with large margin classifiers. In *ICML*, volume 20, page 0.
- Campbell, J. B. and Wynne, R. H. (2011). *Introduction to remote sensing*. Guilford Press.
- Carbonneau, M.-A., Granger, E., and Gagnon, G. (2018). Bag-level aggregation for multiple-instance active learning in instance classification problems. *IEEE transactions on neural networks and learning systems*, 30(5):1441–1451.
- Carneiro, G., Chan, A. B., Moreno, P. J., and Vasconcelos, N. (2007). Supervised learning of semantic classes for image annotation and retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 29(3):394–410.
- Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168.
- Casa, R., Cavalieri, A., and Cascio, B. L. (2011). Nitrogen fertilisation management in precision agriculture: a preliminary application example on maize. *Italian Journal of Agronomy*, 6(1):e5–e5.
- Castelluccio, M., Poggi, G., Sansone, C., and Verdoliva, L. (2015). Land use classification in remote sensing images by convolutional neural networks. *arXiv preprint arXiv:1508.00092*.

- Cebron, N. and Berthold, M. R. (2007). An adaptive multi objective selection strategy for active learning. Technical Report 235, Konstanzer Schriften in Mathematik und Informatik.
- Cesa-Bianchi, N., Gentile, C., Zaniboni, L., and Warmuth, M. (2006). Worst-case analysis of selective sampling for linear classification. *Journal of Machine Learning Research*, 7(7).
- Cesa-Bianchi, N., Lugosi, G., and Stoltz, G. (2005). Minimizing regret with label efficient prediction. *IEEE Transactions on Information Theory*, 51(6):2152–2162.
- Chakraborty, S., Balasubramanian, V., and Panchanathan, S. (2014). Adaptive batch mode active learning. *IEEE transactions on neural networks and learning systems*, 26(8):1747–1760.
- Chakraborty, S., Balasubramanian, V., and Panchanathan, S. (2015). Adaptive batch mode active learning. *IEEE Transactions on Neural Networks and Learning Systems*, 26(8):1747–1760.
- Chang, T. K. H., Bandiera, S. M., and Chen, J. (2003). Constitutive androstane receptor and pregnane x receptor gene expression in human liver: interindividual variability and correlation with cyp2b6 mrna levels. *Drug Metabolism and Disposition*, 31(1):7–10.
- Charbuty, B. and Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, 2(01):20–28.
- Chen, C. H. and Ho, P.-G. P. (2008). Statistical pattern recognition in remote sensing. *Pattern Recognition*, 41(9):2731–2741.
- Chen, G., Wang, T.-j., Gong, L.-y., Boyer, H., et al. (2010). Multi-class support vector machine active learning for music annotation. *International journal of innovative computing and applications*. 2010; 6 (3): 921-30.
- Chen, J. and Zipf, A. (2017). Deepvgi: Deep learning with volunteered geographic information. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 771–772.
- Cheng, T. et al. (2013). Detection of diurnal variation in orchard canopy water content using modis/aster airborne simulator (master) data. *Remote Sensing of Environment*, 132:1–12.
- Cheyer, A. and Martin, D. (2001). The open agent architecture. *Autonomous Agents and Multi-Agent Systems*, 4(1):143–148.
- Chu, W., Zinkevich, M., Li, L., Thomas, A., and Tseng, B. (2011). Unbiased online active learning in data streams. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 195–203.
- Citovsky, G., DeSalvo, G., Gentile, C., Karydas, L., Rajagopalan, A., Rostamizadeh, A., and Kumar, S. (2021). Batch active learning at scale. *Advances in Neural Information Processing Systems*, 34.
- Clark, M. L. (2017). Comparison of simulated hyperspectral hyperspectral and multispectral landsat 8 and sentinel-2 imagery for multi-seasonal, regional land-cover mapping. *Remote Sensing of Environment*, 200:311–325.

- Clark, R. and McGuckin, R. (1996). Variable rate application technology: An overview. In *Proceedings of the Third International Conference on Precision Agriculture*, pages 855–862. Wiley Online Library.
- Cohn, D. A., Ghahramani, Z., and Jordan, M. I. (1996a). Active learning with statistical models. *CoRR*, cs.AI/9603104.
- Cohn, D. A., Ghahramani, Z., and Jordan, M. I. (1996b). Active learning with statistical models. *CoRR*, cs.AI/9603104.
- Coleman, D., Georgiadou, Y., and Labonte, J. (2009). Volunteered geographic information: The nature and motivation of producers. *International journal of spatial data infrastructures research*, 4(4):332–358.
- Colkesen, I. and Kavzoglu, T. (2017). Ensemble-based canonical correlation forest (ccf) for land use and land cover classification using sentinel-2 and landsat oli imagery. *Remote Sensing Letters*, 8(11):1082–1091.
- Collins, J. B. and Woodcock, C. E. (1996). An assessment of several linear change detection techniques for mapping forest mortality using multitemporal landsat tm data. *Remote sensing of Environment*, 56(1):66–77.
- Conservancy, C. (2022). Land cover data project 2013/2014. <https://www.chesapeakeconservancy.org/conservation-innovation-center/high-resolution-data/land-cover-data-project/>. Accessed: 06 04 2022.
- Copernicus (2018). Cool facts for your next copernicus small talk. <https://www.copernicus.eu/en/news/news/observer-cool-facts-your-next-copernicus-small-talk>. Accessed: 06 06 2022.
- Copernicus (2022). Copernicus open access hub. <https://scihub.copernicus.eu/dhus/#/home>. Accessed: 21 03 2022.
- Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., and Lambin, E. (2004). Review articulated digital change detection methods in ecosystem monitoring: a review. *International journal of remote sensing*, 25(9):1565–1596.
- Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.
- Corning (2022). Electro-optical infrared (eoir) systems. <https://www.corning.com/worldwide/en/products/advanced-optics/product-materials/aerospace-defense/eoir-systems.html>. Accessed: 06 04 2022.
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.
- Cressie, N. (2015). *Statistics for spatial data*. John Wiley & Sons.
- Csillik, O. and Belgiu, M. (2017). Cropland mapping from sentinel-2 time series data using object-based image analysis. In *Proceedings of the 20th AGILE International Conference on Geographic Information Science Societal Geo-Innovation Celebrating, Wageningen, The Netherlands*, pages 9–12.

- Cuevas, E., Romero-Campos, P. M., Kouremeti, N., Kazadzis, S., Räisänen, P., García, R. D., Barreto, A., Guirado-Fuentes, C., Ramos, R., Toledano, C., et al. (2019). Aerosol optical depth comparison between gaw-pfr and aeronet-cimel radiometers from long-term (2005–2015) 1 min synchronous measurements. *Atmospheric Measurement Techniques*, 12(8):4309–4337.
- Culotta, A. and McCallum, A. (2005). Reducing labeling effort for structured prediction tasks. In *AAAI*, volume 5, pages 746–751.
- Dagan, I. and Engelson, S. P. (1995). Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier.
- Danielsson, P.-E. (1980). Euclidean distance mapping. *Computer Graphics and image processing*, 14(3):227–248.
- Dao, P. D. and Liou, Y.-A. (2015). Object-based flood mapping and affected rice field estimation with landsat 8 oli and modis data. *Remote Sensing*, 7(5):5077–5097.
- Darmochwał, A. (1991). The euclidean space. *Formalized Mathematics*, 2(4):599–603.
- Darwish, A., Leukert, K., and Reinhardt, W. (2003). Image segmentation for the purpose of object-based classification. In *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477)*, volume 3, pages 2039–2041. Citeseer.
- Das, D. K. and Singh, G. (1989). Estimation of evapotranspiration and scheduling irrigation using remote sensing techniques. In *Proc. Summer Inst*, pages 113–117, on agricultural remote sensing in monitoring crop growth and productivity, IARI, New Delhi.
- Das, S., Wong, W.-K., Dietterich, T., Fern, A., and Emmott, A. (2016). Incorporating expert feedback into active anomaly discovery. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 853–858. IEEE.
- Dasgupta, S. and Hsu, D. (2008). Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215.
- Dasgupta, S., Kalai, A. T., and Tauman, A. (2009). Analysis of perceptron-based active learning. *Journal of Machine Learning Research*, 10(2).
- Dasgupta, S. and Langford, J. (2009). Active learning tutorial, icml 2009.
- De Beurs, K. M. and Townsend, P. A. (2008). Estimating the effect of gypsy moth defoliation using modis. *Remote Sensing of Environment*, 112(10):3983–3990.
- De Maesschalck, R., Jouan-Rimbaud, D., and Massart, D. L. (2000). The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18.
- Dean, A. and Smith, G. (2003). An evaluation of per-parcel land cover mapping using maximum likelihood class probabilities. *International Journal of Remote Sensing*, 24(14):2905–2920.
- Degerickx, J., Roberts, D. A., and Somers, B. (2019). Enhancing the performance of multiple endmember spectral mixture analysis (mesma) for urban land cover mapping using airborne lidar data and band selection. *Remote sensing of environment*, 221:260–273.

- Delalay, M., Tiwari, V., Ziegler, A. D., Gopal, V., and Passy, P. (2019). Land-use and land-cover classification using sentinel-2 data and machine-learning algorithms: operational method and its implementation for a mountainous area of nepal. *Journal of Applied Remote Sensing*, 13(1):014530.
- Delin, S. and Berglund, K. (2005). Management zones classified with respect to drought and waterlogging. *Precision Agriculture*, 6(4):321–340.
- Demir, B., Persello, C., and Bruzzone, L. (2010). Batch-mode active-learning methods for the interactive classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3):1014–1031.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE.
- Deng, J., Wang, K., Deng, Y., and Qi, G. (2008). Pca-based land-use change detection and analysis using multitemporal and multisensor satellite data. *International Journal of Remote Sensing*, 29(16):4823–4838.
- Denize, J., Hubert-Moy, L., Corgne, S., Betbeder, J., and Pottier, E. (2018). Identification of winter land use in temperate agricultural landscapes based on sentinel-1 and 2 times-series. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 8271–8274. IEEE.
- Department of Space, I. S. R. O. (2022). Irs-1a. <https://www.isro.gov.in/Spacecraft/irs-1a>. Accessed: 06 04 2022.
- Derksen, D., Inglada, J., and Michel, J. (2018). Spatially precise contextual features based on superpixel neighborhoods for land cover mapping with high resolution satellite image time series. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 200–203. IEEE.
- Development Seed (2022). Geospatial solutions and global insights for a complex and changing planet. <https://developmentseed.org/>. Accessed: 06 04 2022.
- Devonport, A., Saoud, A., and Arcak, M. (2021). Symbolic abstractions from data: A pac learning approach. *arXiv preprint arXiv:2104.13901*.
- Dima, C. and Hebert, M. (2005). Active learning for outdoor obstacle detection. In *Robotics: Science and Systems*, pages 9–16. Citeseer.
- Dixit, A., Goswami, A., and Jain, S. (2019). Development and evaluation of a new snow water index (swi) for accurate snow cover delineation. *Remote Sensing*, 11(23).
- Dong, Q., Chen, X., Chen, J., Zhang, C., Liu, L., Cao, X., Zang, Y., Zhu, X., and Cui, X. (2020). Mapping winter wheat in north china using sentinel 2a/b data: A method based on phenology-time weighted dynamic time warping. *Remote Sensing*, 12(8):1274.
- Donmez, P. and Carbonell, J. (2008a). Paired-sampling in density-sensitive active learning. *10th International Symposium on Artificial Intelligence and Mathematics, ISAIM 2008*.
- Donmez, P. and Carbonell, J. G. (2008b). Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 619–628.

- Donmez, P., Carbonell, J. G., and Bennett, P. N. (2007). Dual strategy active learning. In *European Conference on Machine Learning*, pages 116–127. Springer.
- Doraiswamy, P. C. et al. (2005). Application of modis derived parameters for regional crop yield assessment. *Remote sensing of environment*, 97(2):192–202.
- Drusch, M., Del Bello, U., Carlier, S., Colin, O., Fernandez, V., Gascon, F., Hoersch, B., Isola, C., Laberinti, P., Martimort, P., et al. (2012). Sentinel-2: Esa’s optical high-resolution mission for gmes operational services. *Remote sensing of Environment*, 120:25–36.
- Dubovik, O., Schuster, G. L., Xu, F., Hu, Y., Bösch, H., Landgraf, J., and Li, Z. (2021). Grand challenges in satellite remote sensing. *Frontiers in Remote Sensing*, page 1.
- Dutta, D. et al. (2015). Assessment of agricultural drought in rajasthan (india) using remote sensing derived vegetation condition index (vci) and standardized precipitation index (spi). *The Egyptian Journal of Remote Sensing and Space Science*, 18(1):53–63.
- Dwivedi, Y. K., Hughes, L., Ismagilova, E., Aarts, G., Coombs, C., Crick, T., Duan, Y., Dwivedi, R., Edwards, J., Eirug, A., et al. (2021). Artificial intelligence (ai): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *International Journal of Information Management*, 57:101994.
- Efron, B. (2004). The estimation of prediction error: covariance penalties and cross-validation. *Journal of the American Statistical Association*, 99(467):619–632.
- Elith, J., Leathwick, J. R., and Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, 77(4):802–813.
- Escobar, F. (2020). 'tracking-water-levels-in-satellite-images'. Accessed: 21 01 2021.
- European Space Agency (2020). Sentinel-2 msi - level 2a products algorithm theoretical basis document. https://earth.esa.int/c/document_library/get_file?folderId=349490&name=DLFE-4518.pdf. Accessed: 04 02 2020.
- European Space Agency (2022a). Copernicus sentinel expansion missions. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Copernicus_Sentinel_Expansion_missions. Accessed: 25 05 2022.
- European Space Agency (2022b). Environmental monitoring and analysis program. <https://earth.esa.int/web/eoportal/satellite-missions/e/enmap>. Accessed: 06 04 2022.
- European Space Agency (2022c). Sentinel-1 radar vision for copernicus. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-1. Accessed: 25 05 2022.
- European Space Agency (2022d). Sentinel-2 colour vision for copernicus. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-2. Accessed: 25 05 2022.
- European Space Agency (2022e). Sentinel-3 a bigger picture for copernicus. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-3. Accessed: 25 05 2022.

- European Space Agency (2022f). Sentinel-4. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-4_-5_and_-5P. Accessed: 25 05 2022.
- European Space Agency (2022g). Sentinel-5p global air monitoring for copernicus. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-5P. Accessed: 25 05 2022.
- European Space Agency (2022h). Sentinel-6 charting sea level for copernicus. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-6. Accessed: 25 05 2022.
- European Space Agency (2022i). The sentinel missions. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/The_Sentinel_missions. Accessed: 25 05 2022.
- European Space Agency (2022j). Sentinelsat 1.1.1 documentation. https://sentinelsat.readthedocs.io/en/latest/api_reference.html. Accessed: 21 03 2022.
- Fang, Y. T., Gundersen, P., Mo, J. M., and Zhu, W. (2008). Input and output of dissolved organic and inorganic nitrogen in subtropical forests of south china under high air pollution. *Biogeosciences*, 5(2):339–352.
- Fedorov, V. V. (2013). *Theory of optimal experiments*. Elsevier.
- Felderer, M. and Ramler, R. (2021). Quality assurance for ai-based systems: Overview and challenges (introduction to interactive session). In *International Conference on Software Quality*, pages 33–42. Springer.
- Feldman, J. A. and Yakimovsky, Y. (1974). Decision theory and artificial intelligence: I. a semantics-based region analyzer. *Artificial Intelligence*, 5(4):349–371.
- Fensholt, R. and Sandholt, I. (2003). Derivation of a shortwave infrared water stress index from modis near-and shortwave infrared data in a semiarid environment. *Remote Sensing of Environment*, 87(1):111–121.
- Ferecatu, M. and Boujemaa, N. (2007). Interactive remote-sensing image retrieval using active relevance feedback. *IEEE Transactions on Geoscience and Remote Sensing*, 45(4):818–826.
- Ferencz, C. et al. (2004). Crop yield estimation by satellite remote sensing. *International Journal of Remote Sensing*, 25(20):4113–4149.
- Ferrettini, G., Aligon, J., and Soulé-Dupuy, C. (2020). Explaining single predictions: A faster method. In Chatzigeorgiou, A., Dondi, R., Herodotou, H., Kapoutsis, C., Manolopoulos, Y., Papadopoulos, G. A., and Sikora, F., editors, *SOFSEM 2020: Theory and Practice of Computer Science*, pages 313–324, Cham. Springer International Publishing.
- Fisher, P., Arnot, C., Wadsworth, R., and Wellens, J. (2006). Detecting change in vague interpretations of landscapes. *Ecological Informatics*, 1(2):163–178.
- Forkuor, G., Dimobe, K., Serme, I., and Tondoh, J. E. (2018). Landsat-8 vs. sentinel-2: examining the added value of sentinel-2s red-edge bands to land-use and land-cover mapping in burkina faso. *GIScience & remote sensing*, 55(3):331–354.

- Franklin, S. E. (2001). *Remote sensing for sustainable forest management*. CRC press.
- Frantz, D., Haß, E., Uhl, A., Stoffels, J., and Hill, J. (2018). Improvement of the fmask algorithm for sentinel-2 images: Separating clouds from bright surfaces based on parallax effects. *Remote sensing of environment*, 215:471–481.
- Freund, Y. and Schapire, R. E. (1997a). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139.
- Freund, Y. and Schapire, R. E. (1997b). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.
- Freund, Y., Seung, H. S., Shamir, E., and Tishby, N. (1992). Information, prediction, and query by committee. *Advances in neural information processing systems*, 5.
- Freund, Y., Seung, H. S., Shamir, E., and Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2):133–168.
- Fu, L.-L., Lee, T., Liu, W. T., and Kwok, R. (2019). 50 years of satellite remote sensing of the ocean. *Meteorological Monographs*, 59:5–1.
- Fu, Y., Li, B., Zhu, X., and Zhang, C. (2014). Active learning without knowing individual instance labels: A pairwise label homogeneity query approach. *IEEE Transactions on Knowledge and Data Engineering*, 26(4):808–822.
- Fu, Y., Zhu, X., and Li, B. (2013). A survey on instance selection for active learning. *Knowledge and information systems*, 35(2):249–283.
- Fujii, A., Inui, K., Tokunaga, T., and Tanaka, H. (1999). Selective sampling for example-based word sense disambiguation. *arXiv preprint cs/9910020*.
- Gabrani, M. and Tretiak, O. J. (1996). Elastic transformations. In *Conference Record of The Thirtieth Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 501–505. IEEE.
- Gamba, P., Dell’Acqua, F., and Lisini, G. (2006). Change detection of multitemporal sar data in urban areas combining feature-based and pixel-based techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 44(10):2820–2827.
- Gammerman, A. and Vovk, V. (2002). Prediction algorithms and confidence measures based on algorithmic randomness theory. *Theoretical Computer Science*, 287(1):209–217. Natural Computing.
- Gammerman, A. and Vovk, V. (2007). Hedging predictions in machine learning. *The Computer Journal*, 50(2):151–163.
- Gao, Y., Skutsch, M., Paneque-Gálvez, J., and Ghilardi, A. (2020). Remote sensing of forest degradation: a review. *Environmental Research Letters*, 15(10):103001.
- Gaston, K. J. (2000). Global patterns in biodiversity. *Nature*, 405(6783):220–227.
- GDAL/OGR contributors (2020). *GDAL/OGR Geospatial Data Abstraction software Library*. Open Source Geospatial Foundation.

- Gebbers, R. and Adamchuk, V. I. (2010). Precision agriculture and food security. *Science*, 327(5967):828–831.
- Geiger, F., Bengtsson, J., Berendse, F., Weisser, W. W., Emmerson, M., Morales, M. B., Ceryngier, P., Liira, J., Tschardtke, T., Winqvist, C., Eggers, S., Bommarco, R., Pärt, T., Bretagnolle, V., Plantegenest, M., Clement, L. W., Dennis, C., Palmer, C., Oñate, J. J., Guerrero, I., Hawro, V., Aavik, T., Thies, C., Flohre, A., Hänke, S., Fischer, C., Goedhart, P. W., and Inchausti, P. (2010). Persistent negative effects of pesticides on biodiversity and biological control potential on european farmland. *Basic and Applied Ecology*, 11(2):97–105.
- Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomized trees. *Machine learning*, 63(1):3–42.
- Gibbons, G. (2000). Turning a farm art into science-an overview of precision farming. URL: <http://www.precisionfarming.com>.
- Glinskis, E. A. and Gutiérrez-Vélez, V. H. (2019). Quantifying and understanding land cover changes by large and small oil palm expansion regimes in the peruvian amazon. *Land Use Policy*, 80:95–106.
- Goetz, A. F., Rock, B. N., and Rowan, L. C. (1983). Remote sensing for exploration; an overview. *Economic Geology*, 78(4):573–590.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Gu, Y., Jin, Z., and Chiu, S. C. (2014). Active learning with maximum density and minimum redundancy. In Loo, C. K., Yap, K. S., Wong, K. W., Teoh, A., and Huang, K., editors, *Neural Information Processing*, pages 103–110, Cham. Springer International Publishing.
- Guanter, L., Kaufmann, H., Segl, K., Foerster, S., Rogass, C., Chabrillat, S., Kuester, T., Hollstein, A., Rossner, G., Chlebek, C., et al. (2015). The enmap spaceborne imaging spectroscopy mission for earth observation. *Remote Sensing*, 7(7):8830–8857.
- Gui, X., Lu, X., and Yu, G. (2021). Cost-effective batch-mode multi-label active learning. *Neurocomputing*, 463:355–367.
- Guo, Y. and Greiner, R. (2007). Optimistic active-learning using mutual information. In *IJCAI*, volume 7, pages 823–829.
- Guo, Y. and Schuurmans, D. (2007). Discriminative batch mode active learning. *Advances in neural information processing systems*, 20.
- Gutierrez, F., Gil, A., Reis, E., Lobo, A., Neto, C., Calado, H., and Costa, J. C. (2011). *Acacia saligna* (labill.) h. wendl in the sesimbra county: Invaded habitats and potential distribution modeling. *Journal of Coastal Research*, pages 403–407.
- Hadria, R. et al. (2006). Monitoring of irrigated wheat in a semiarid climate using crop modelling and remote sensing data: Impact of satellite revisit time frequency. *International Journal of Remote Sensing*, 27(6):1093–1117.

- Haertel, R. A., Seppi, K. D., Ringger, E. K., and Carroll, J. L. (2008). Return on investment for active learning. In *Proceedings of the NIPS workshop on cost-sensitive learning*, volume 72.
- Hagolle, O., Huc, M., Pascual, D. V., and Dedieu, G. (2010). A multi-temporal method for cloud detection, applied to formosat-2, venus, landsat and sentinel-2 images. *Remote Sensing of Environment*, 114(8):1747–1755.
- Han, Q. et al. (2012). Polyfunctional responses by human t cells result from sequential release of cytokines. *Proceedings of the National Academy of Sciences*, 109(5):1607–1612.
- Han, Y., Jiao, J., and Weissman, T. (2015). Minimax estimation of discrete distributions. In *2015 IEEE International Symposium on Information Theory (ISIT)*, pages 2291–2295. IEEE.
- Hanid, M. (2014). *Design science research as an approach to develop conceptual solutions for improving cost management in construction*. University of Salford (United Kingdom).
- Hanneke, S. (2007). A bound on the label complexity of agnostic active learning. In *Proceedings of the 24th international conference on Machine learning*, pages 353–360.
- Hanneke, S. (2014). Theory of disagreement-based active learning. *Foundations and Trends in Machine Learning*, 7(2-3):131–309.
- Hansen, M. C., Potapov, P. V., Moore, R., Hancher, M., Turubanova, S. A., Tyukavina, A., Thau, D., Stehman, S., Goetz, S. J., Loveland, T. R., et al. (2013). High-resolution global maps of 21st-century forest cover change. *science*, 342(6160):850–853.
- Haralick, R. M. and Shapiro, L. G. (1992). *Computer and robot vision*, volume 1. Addison-wesley Reading.
- Hashem, N. and Balakrishnan, P. (2015). Change analysis of land use/land cover and modelling urban growth in greater doha, qatar. *Annals of GIS*, 21(3):233–247.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.
- Hatfield, J. L. and Prueger, J. H. (2010). Value of using different vegetative indices to quantify agricultural crop characteristics at different growth stages under varying management practices. *Remote Sensing*, 2(2):562–578.
- Hauptmann, A. G., Lin, W.-H., Yan, R., Yang, J., and Chen, M.-Y. (2006). Extreme video retrieval: joint maximization of human and computer performance. In *Proceedings of the 14th ACM international conference on Multimedia*, pages 385–394.
- Helber, P., Bischke, B., Dengel, A., and Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226.
- Helmhold, D. and Panizza, S. (1997). Some label efficient learning results. In *Proceedings of the tenth annual conference on Computational learning theory*, pages 218–230.
- Herfort, B. (2018). *Understanding MapSwipe: Analysing Data Quality of Crowdsourced Classifications on Human Settlements*. PhD thesis, Fakultät für Chemie und Geowissenschaften, Institute of Geography.

- Heryadi, Y. and Miranda, E. (2019). Land cover classification based on sentinel-2 satellite imagery using convolutional neural network model: A case study in semarang area, indonesia. In *Asian Conference on Intelligent Information and Database Systems*, pages 191–206. Springer.
- Hoi, S. C., Jin, R., and Lyu, M. R. (2006a). Large-scale text categorization by batch mode active learning. In *Proceedings of the 15th international conference on World Wide Web*, pages 633–642.
- Hoi, S. C., Jin, R., Zhu, J., and Lyu, M. R. (2006b). Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd international conference on Machine learning*, pages 417–424.
- Hoi, S. C., Jin, R., Zhu, J., and Lyu, M. R. (2009). Semisupervised svm batch mode active learning with applications to image retrieval. *ACM Transactions on Information Systems (TOIS)*, 27(3):1–29.
- Hollstein, A., Segl, K., Guanter, L., Brell, M., and Enesco, M. (2016). Ready-to-use methods for the detection of clouds, cirrus, snow, shadow, water and clear sky pixels in sentinel-2 msi images. *Remote Sensing*, 8(8).
- Howarth, P. J. and Wickware, G. M. (1981). Procedures for change detection using landsat digital data. *International Journal of Remote Sensing*, 2(3):277–291.
- Hu, F., Xia, G.-S., Hu, J., and Zhang, L. (2015a). Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, 7(11):14680–14707.
- Hu, Y., Gao, S., Janowicz, K., Yu, B., Li, W., and Prasad, S. (2015b). Extracting and understanding urban areas of interest using geotagged photos. *Computers, Environment and Urban Systems*, 54:240–254.
- Huang, C., Song, K., Kim, S., Townshend, J. R., Davis, P., Masek, J. G., and Goward, S. N. (2008). Use of a dark object concept and support vector machines to automate forest cover change analysis. *Remote sensing of environment*, 112(3):970–985.
- Huang, L., Matwin, S., de Carvalho, E. J., and Minghim, R. (2017). Active learning with visualization for text data. In *Proceedings of the 2017 ACM Workshop on Exploratory Search and Interactive Data Analytics*, pages 69–74.
- Huang, S.-J., Jin, R., and Zhou, Z.-H. (2010). Active learning by querying informative and representative examples. *Advances in neural information processing systems*, 23.
- Huete, A. R. (1988). A soil-adjusted vegetation index (savi). *Remote sensing of environment*, 25(3):295–309.
- Huijser, M. and van Gemert, J. C. (2017). Active decision boundary annotation with deep generative models. In *Proceedings of the IEEE international conference on computer vision*, pages 5286–5295.
- Hüllermeier, E. and Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3):457–506.
- Hunter, D. R., Goodreau, S. M., and Handcock, M. S. (2008). Goodness of fit of social network models. *Journal of the american statistical association*, 103(481):248–258.

- Hussain, M., Chen, D., Cheng, A., Wei, H., and Stanley, D. (2013). Change detection from remotely sensed images: From pixel-based to object-based approaches. *ISPRS Journal of photogrammetry and remote sensing*, 80:91–106.
- Hwa, R. (2004). Sample selection for statistical parsing. *Computational linguistics*, 30(3):253–276.
- İlsever, M. and Ünsalan, C. (2012). *Two-dimensional change detection methods: remote sensing applications*. Springer Science & Business Media.
- Im, J. and Jensen, J. R. (2005). A change detection model based on neighborhood correlation image analysis and decision tree classification. *Remote Sensing of Environment*, 99(3):326–340.
- Inselberg, A. (1985). The plane with parallel coordinates. *The visual computer*, 1(2):69–91.
- Iwamura, M., Omachi, S., and Aso, H. (2000). A modification of eigenvalues to compensate estimation errors of eigenvectors. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 2, pages 378–381. IEEE.
- Iwamura, M., Omachi, S., and Aso, H. (2004). Estimation of true mahalanobis distance from eigenvectors of sample covariance matrix. *Systems and Computers in Japan*, 35(9):30–38.
- Jackson, R. D. (1986). Remote sensing of biotic and abiotic plant stress. *Annual review of Phytopathology*, 24(1):265–287.
- Jackson, R. D. et al. (1981). Canopy temperature as a crop water stress indicator. *Water resources research*, 17(4):1133–1138.
- Japan Association (2022). Remote sensing notes. http://sar.kangwon.ac.kr/etc/rs_note/rsnote/cp9/cp9-1.htm. Accessed: 25 05 2022.
- Jarvis, R. A. (1983). A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:122–139.
- Jeyaseelan, A. (2003). Droughts & floods assessment and monitoring using remote sensing and gis. *Satellite remote sensing and GIS applications in agricultural meteorology*, 291.
- Ji, L. and Peters, A. J. (2003). Assessing vegetation response to drought in the northern great plains using vegetation and drought indices. *Remote Sensing of Environment*, 87(1):85–98.
- Jin, S. and Sader, S. A. (2005). Comparison of time series tasseled cap wetness and the normalized difference moisture index in detecting forest disturbances. *Remote sensing of Environment*, 94(3):364–372.
- Johnson, H. and Green, R. (1995). Aviris user’s guide. *Summaries of the Fifth Annual JPL Airborne Earth Science Workshop*, 1:105–108.
- Joshi, N., Baumann, M., Ehammer, A., Fensholt, R., Grogan, K., Hostert, P., Jepsen, M. R., Kuemmerle, T., Meyfroidt, P., Mitchard, E. T., et al. (2016). A review of the application of optical and radar remote sensing data fusion to land use mapping and monitoring. *Remote Sensing*, 8(1):70.

- Jr, H., Raymond, E., et al. (2013). A visible band index for remote sensing leaf chlorophyll content at the canopy scale. *International Journal of Applied Earth Observation and Geoinformation*, 21:103–112.
- Júnior, A. S., Renso, C., and Matwin, S. (2017). Analytic: An active learning system for trajectory classification. *IEEE computer graphics and applications*, 37(5):28–39.
- Juutilainen, I. and Röning, J. (2007). A method for measuring distance from a training data set. *Communications in Statistics Theory and Methods*, 36(14):2625–2639.
- Kadhim, N., Mourshed, M., and Bray, M. (2016). Advances in remote sensing applications for urban sustainability. *Euro-Mediterranean Journal for Environmental Integration*, 1(1):1–22.
- Kang, J., Ryu, K. R., and Kwon, H.-C. (2004). Using cluster-based sampling to select initial training set for active learning in text classification. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 384–388. Springer.
- Kapoor, A., Horvitz, E., and Basu, S. (2007). Selective supervision: Guiding supervised learning with decision-theoretic active learning. In *IJCAI*, volume 7, pages 877–882.
- Karlos, S., Aridas, C., Kanas, V. G., and Kotsiantis, S. (2021). Classification of acoustical signals by combining active learning strategies with semi-supervised learning schemes. *Neural Computing and Applications*, pages 1–18.
- Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons.
- Kim, S.-R., Lee, W.-K., Kwak, D.-A., Biging, G. S., Gong, P., Lee, J.-H., and Cho, H.-K. (2011). Forest cover classification by optimal segmentation of high resolution satellite imagery. *Sensors*, 11(2):1943–1958.
- Kimura, F., Takashina, K., Tsuruoka, S., and Miyake, Y. (1987). Modified quadratic discriminant functions and the application to chinese character recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(1):149–153.
- King, A. J., He, W., Cuevas, J. A., Freudenberger, M., Ramiarmanana, D., and Graham, I. A. (2009). Potential of *Jatropha curcas* as a source of renewable oil and animal feed. *Journal of Experimental Botany*, 60(10):2897–2905.
- King, D. R., Dalton, D. R., Daily, C. M., and Covin, J. G. (2004a). Meta-analyses of post-acquisition performance: indications of unidentified moderators. *Strategic Management Journal*, 25(2):187–200.
- King, R. D., Whelan, K. E., Jones, F. M., Reiser, P. G., Bryant, C. H., Muggleton, S. H., Kell, D. B., and Oliver, S. G. (2004b). Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427(6971):247–252.
- Kobayashi, Y. (2019). Improved method for correcting sample mahalanobis distance without estimating population eigenvalues or eigenvectors of covariance matrix. *International Journal of Data Science and Analytics*, pages 1–14.
- Kogan, F. N. (1995). Application of vegetation index and brightness temperature for drought detection. *Advances in space research*, 15(11):91–100.

- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69.
- Komissio, E. (2018). The common agricultural policy at a glance. *The common agricultural policy supports farmers and ensures Europe's food security. Noudettu*, 24:2019.
- Komura, D. and Ishikawa, S. (2019). Machine learning approaches for pathologic diagnosis. *Virchows Archiv*, 475(2):131–138.
- Koriat, A., Lichtenstein, S., and Fischhoff, B. (1980). Reasons for confidence. *Journal of Experimental Psychology: Human learning and memory*, 6(2):107.
- Kramer, O. (2013). K-nearest neighbors. In *Dimensionality reduction with unsupervised nearest neighbors*, pages 13–23. Springer.
- Krishnamurthy, V. (2002). Algorithms for optimal scheduling and management of hidden markov model sensors. *IEEE Transactions on Signal Processing*, 50(6):1382–1397.
- Krishnapuram, B., Williams, D., Xue, Y., Carin, L., Figueiredo, M. A., and Hartemink, A. J. (2005). Active learning of features and labels. In *Workshop on learning with multiple views at the 22nd International Conference on Machine Learning (ICML-05)*, pages 43–50.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- Kruglanski, A. W. (1989). The psychology of being "right": The problem of accuracy in social perception and cognition. *Psychological bulletin*, 106(3):395.
- Kuhn, M. and Johnson, K. (2013). Classification trees and rule-based models. In *Applied predictive modeling*, pages 369–413. Springer.
- Kunkel, B., Blechinger, F., Lutz, R., Doerffer, R., Van der Piepen, H., and Schroder, M. (1988). Rosis (reflective optics system imaging spectrometer)-a candidate instrument for polar platform missions. In *Optoelectronic technologies for remote sensing from space*, volume 868, pages 134–141. SPIE.
- Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Mallocci, M., Kolesnikov, A., et al. (2020). The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981.
- Kwok, S. W. and Carter, C. (1990). Multiple decision trees. In *Machine intelligence and pattern recognition*, volume 9, pages 327–335. Elsevier.
- Lancaster, H. O. and Seneta, E. (2005). Chi-square distribution. *Encyclopedia of biostatistics*, 2.
- Lange, A., Atkinson, C., and Tunjic, C. (2020). Simplified view generation in a deep view-based modeling environment. In *International Conference on Systems Modelling and Management*, pages 163–179. Springer.
- Langley, R. B. (1998). The UTM grid system. *GPS world*, 9(2):46–50.
- Laurent, V. C., Schaepman, M. E., Verhoef, W., Weyermann, J., and Chávez, R. O. (2014). Bayesian object-based estimation of lai and chlorophyll from a simulated sentinel-2 top-of-atmosphere radiance image. *Remote Sensing of Environment*, 140:318–329.

- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551.
- Lee, W. S. et al. (2010). Sensing technologies for precision specialty crop production. *Computers and electronics in agriculture*, 74(1):2–33.
- Leng, Y., Xu, X., and Qi, G. (2013). Combining active learning and semi-supervised learning to construct svm classifier. *Knowledge-Based Systems*, 44:121–131.
- Lentile, L. B., Holden, Z. A., Smith, A. M., Falkowski, M. J., Hudak, A. T., Morgan, P., Lewis, S. A., Gessler, P. E., and Benson, N. C. (2006). Remote sensing techniques to assess active fire characteristics and post-fire effects. *International Journal of Wildland Fire*, 15(3):319–345.
- Lewis, D. D. and Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pages 148–156. Elsevier.
- Lewis, D. D. and Gale, W. A. (1994a). A sequential algorithm for training text classifiers. In Croft, B. W. and van Rijsbergen, C. J., editors, *SIGIR '94*, pages 3–12, London. Springer London.
- Lewis, D. D. and Gale, W. A. (1994b). A sequential algorithm for training text classifiers. In *SIGIR94*, pages 3–12. Springer.
- Li, M. and Sethi, I. K. (2006). Confidence-based active learning. *IEEE transactions on pattern analysis and machine intelligence*, 28(8):1251–1261.
- Li, M., Vitányi, P., et al. (2008). *An introduction to Kolmogorov complexity and its applications*, volume 3. Springer.
- Li, T. and Anand, S. S. (2007). Diva: a variance-based clustering approach for multi-type relational data. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 147–156.
- Li, Z., Shen, H., Cheng, Q., Liu, Y., You, S., and He, Z. (2019). Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150:197–212.
- Li-Cor (2022). Quantum sensor. <https://www.licor.com/env/products/light/quantum>. Accessed: 24 05 2022.
- Liere, R. and Tadepalli, P. (1997). Active learning with committees for text categorization. In *AAAI/IAAI*, pages 591–596. Citeseer.
- Lin, M., Chen, Q., and Yan, S. (2013a). Network in network. *arXiv preprint arXiv:1312.4400*.
- Lin, T.-Y., Belongie, S., and Hays, J. (2013b). Cross-view image geolocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898.

- Lin, T.-Y., Cui, Y., Belongie, S., and Hays, J. (2015). Learning deep representations for ground-to-aerial geolocalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5007–5015.
- Lindenbaum, M., Markovitch, S., and Rusakov, D. (2004). Selective sampling for nearest neighbor classifiers. *Machine learning*, 54(2):125–152.
- Lindsay, B. G., Markatou, M., and Ray, S. (2014). Kernels, degrees of freedom, and power properties of quadratic distance goodness-of-fit tests. *Journal of the American Statistical Association*, 109(505):395–410.
- Lindsay, B. G., Markatou, M., Ray, S., Yang, K., Chen, S.-C., et al. (2008). Quadratic distances on probabilities: A unified foundation. *The Annals of Statistics*, 36(2):983–1006.
- Liou, Y.-A., Nguyen, A. K., and Li, M.-H. (2017). Assessing spatiotemporal eco-environmental vulnerability by landsat data. *Ecological Indicators*, 80:52–65.
- Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2(4):285–318.
- Liu, D. and Xia, F. (2010). Assessing object-based classification: advantages and limitations. *Remote sensing letters*, 1(4):187–194.
- Liu, P., Zhang, H., and Eom, K. B. (2016). Active deep learning for classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(2):712–724.
- Liu, Y. (2004). Active learning with support vector machine applied to gene expression data for cancer classification. *Journal of chemical information and computer sciences*, 44(6):1936–1941.
- Liu, Y., Zhong, Y., Fei, F., Zhu, Q., and Qin, Q. (2018). Scene classification based on a deep random-scale stretched convolutional neural network. *Remote Sensing*, 10(3):444.
- Llewellyn-Jones, D., Edwards, M., Mutlow, C., Birks, A., Barton, I., and Tait, H. (2001). Aatsr: Global-change and surface-temperature measurements from envisat. *ESA bulletin*, 105(10-21):25.
- Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.
- Lonjou, V., Desjardins, C., Hagolle, O., Petrucci, B., Tremas, T., Dejus, M., Makarau, A., and Auer, S. (2016). Maccs-atcor joint algorithm (maja). In *Remote Sensing of Clouds and the Atmosphere XXI*, volume 10001, page 1000107. International Society for Optics and Photonics.
- Louis, J., Debaecker, V., Pflug, B., Main-Knorn, M., Bieniarz, J., Mueller-Wilm, U., Cadau, E., and Gascon, F. (2016). Sentinel-2 sen2cor: L2a processor for users. In *Proceedings Living Planet Symposium 2016*, pages 1–8. Spacebooks Online.
- Lourentzou, I., Gruhl, D., and Welch, S. (2018). Exploring the efficiency of batch active learning for human-in-the-loop relation extraction. In *Companion Proceedings of the The Web Conference 2018*, pages 1131–1138.

- Lu, D., Mausel, P., Brondizio, E., and Moran, E. (2004). Change detection techniques. *International journal of remote sensing*, 25(12):2365–2401.
- Lubischew, A. A. (1962). On the use of discriminant functions in taxonomy. *Biometrics*, pages 455–477.
- Ludeke, A. K., Maggio, R. C., and Reid, L. M. (1990). An analysis of anthropogenic deforestation using logistic regression and gis. *Journal of Environmental Management*, 31(3):247–259.
- Lunetta, R. S., Knight, J. F., Ediriwickrema, J., Lyon, J. G., and Worthy, L. D. (2006). Land-cover change detection using multi-temporal modis ndvi data. *Remote sensing of environment*, 105(2):142–154.
- Luo, T., Kramer, K., Goldgof, D. B., Hall, L. O., Samson, S., Remsen, A., Hopkins, T., and Cohn, D. (2005). Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6(4).
- Machado, A., Rocha, F., Gomes, C., Dias, J., Araújo, M., and Gouveia, A. (2005). Mineralogical and geochemical characterisation of surficial sediments from the southwestern iberian continental shelf. *Thalassas*, 21(1):67–76.
- MacKay, D. J. (1992). Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604.
- Mahalanobis, P. C. (1936). On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55.
- Main-Knorn, M., Louis, J., Hagolle, O., Müller-Wilm, U., and Alonso, K. (2018). The sen2cor and maja cloud masks and classification products. In *Proceedings of the 2nd Sentinel-2 Validation Team Meeting, ESA-ESRIN, Frascati, Rome, Italy*, pages 29–31.
- Mans, G. (2011). Developing a geo-data frame using dasymetric mapping principles to facilitate data integration. In *AfriGEO Conference: Developing Geomatics for Africa*.
- MapSwipe (2022). Swiping is just the beginning. explore the data. <https://mapswipe.org/en/data.html>. Accessed: 06 04 2022.
- Margineantu, D. D. (2005). Active cost-sensitive learning. In *IJCAI*, volume 5, pages 1622–1623.
- Marja, R., Kleijn, D., Tschardtke, T., Klein, A.-M., Frank, T., and Batáry, P. (2019). Effectiveness of agri-environmental management on pollinators is moderated more by ecological contrast than by landscape structure or land-use intensity. *Ecology Letters*, 22(9):1493–1500.
- Markatou, M. and Sofikitou, E. M. (2019). Statistical distances and the construction of evidence functions for model adequacy. *Frontiers in Ecology and Evolution*, 7:447.
- Markou, M. and Singh, S. (2003). Novelty detection: a reviewpart 1: statistical approaches. *Signal processing*, 83(12):2481–2497.
- Mather, P. M. and Koch, M. (2011). *Computer processing of remotely-sensed images: an introduction*. John Wiley & Sons.

- Mayer, C. and Timofte, R. (2020). Adversarial sampling for active learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3071–3079.
- McCallumzy, A. K. and Nigamy, K. (1998). Employing em and pool-based active learning for text classification. In *Proc. International Conference on Machine Learning (ICML)*, pages 359–367. Citeseer.
- McDonald, R. P. and Marsh, H. W. (1990). Choosing a multivariate model: Noncentrality and goodness of fit. *Psychological bulletin*, 107(2):247.
- McLachlan, G. J. (1999). Mahalanobis distance. *Resonance*, 4(6):20–26.
- McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.
- Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35.
- Melville, P. and Mooney, R. J. (2004). Diverse ensembles for active learning. In *Proceedings of the twenty-first international conference on Machine learning*, page 74.
- Meneses, B. M. et al. (2018). Modelling land use and land cover changes in portugal: A multi-scale and multi-temporal approach. *Finisterra: Revista Portuguesa de Geografia*, 53:107.
- Menon, A. R. R. (2012). Remote sensing application in agriculture and forestry. *DOI*, 10:13140.
- Microsoft (2022a). Featured ai for earth partners. <https://www.microsoft.com/en-us/ai/ai-for-earth>. Accessed: 06 04 2022.
- Microsoft (2022b). A planetary computer for a sustainable future. <https://planetarycomputer.microsoft.com/>. Accessed: 06 04 2022.
- Miranda, E., Mutiara, A. B., Wibowo, W. C., et al. (2018). Classification of land cover from sentinel-2 imagery using supervised classification technique (preliminary study). In *2018 International Conference on Information Management and Technology (ICIMTech)*, pages 69–74. IEEE.
- Mirik, M. et al. (2013). Remote monitoring of wheat streak mosaic progression using sub-pixel classification of landsat 5 tm imagery for site specific disease management in winter wheat. *Advances in Remote Sensing*, 2.
- Mitchell, T. M. (1982). Generalization as search. *Artificial intelligence*, 18(2):203–226.
- Mitra, P., Murthy, C., and Pal, S. K. (2004a). A probabilistic active support vector learning algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):413–418.
- Mitra, P., Shankar, B. U., and Pal, S. K. (2004b). Segmentation of multispectral remote sensing images using active support vector machines. *Pattern recognition letters*, 25(9):1067–1074.

- Mogensen, C. E. et al. (1996). Prevention of diabetic renal disease with special reference to microalbuminuria. In and, T. K., editor, *and Hypertension in Diabetes Mellitus*, pages 539–549. Springer, Boston, MA.
- Mohajerani, S., Krammer, T. A., and Saeedi, P. (2018). A cloud detection algorithm for remote sensing images using fully convolutional neural networks. In *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–5.
- Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of Machine Learning. Adaptive Computation and Machine Learning*. MIT Press, Cambridge, MA, 2 edition.
- Mongus, D. and Žalik, B. (2018). Segmentation schema for enhancing land cover identification: A case study using sentinel 2 data. *International journal of applied earth observation and geoinformation*, 66:56–68.
- Mooney, P. and Minghini, M. (2017). *A Review of OpenStreetMap Data*, pages 37–60. Ubiquity Press.
- Moran, M. S. et al. (1994). Estimating crop water deficit using the relation between surface-air temperature and spectral vegetation index. *Remote sensing of environment*, 49(3):246–263.
- Moskovitch, R., Nissim, N., Stopel, D., Feher, C., Englert, R., and Elovici, Y. (2007). Improving the detection of unknown computer worms activity using active learning. In *Annual Conference on Artificial Intelligence*, pages 489–493. Springer.
- Motta, R., Andrade Lopes, A. d., and Oliveira, M. C. F. d. (2009). Centrality measures from complex networks in active learning. In *International Conference on Discovery Science*, pages 184–196. Springer.
- Mousivand, A., Verhoef, W., Menenti, M., and Gorte, B. (2015). Modeling top of atmosphere radiance over heterogeneous non-lambertian rugged terrain. *Remote sensing*, 7(6):8019–8044.
- Moustakidis, S., Mallinis, G., Koutsias, N., Theocharis, J. B., and Petridis, V. (2011). Svm-based fuzzy decision trees for classification of high spatial resolution remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 50(1):149–169.
- Munoz-Mari, J., Tuia, D., and Camps-Valls, G. (2012). Semisupervised classification of remote sensing images with active queries. *IEEE transactions on geoscience and remote sensing*, 50(10):3751–3763.
- Muslea, I., Minton, S., and Knoblock, C. A. (2000). Selective sampling with redundant views. In *AAAI/IAAI*, pages 621–626.
- Muslea, I., Minton, S., and Knoblock, C. A. (2006). Active learning with multiple views. *Journal of Artificial Intelligence Research*, 27:203–233.
- Myint, S. W., Gober, P., Brazel, A., Grossman-Clarke, S., and Weng, Q. (2011). Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote sensing of environment*, 115(5):1145–1161.
- NASA Earth Observatory (2022). The earth-sensing legacy. https://earthobservatory.nasa.gov/features/E01/eo1_2.php. Accessed: 24 05 2022.

- NASA EarthData NASA Distributed Active Archive Center (2022). Amsr-e overview. <https://nsidc.org/data/amsre>. Accessed: 24 05 2022.
- NASA EarthData OceanColor Web (2022). Seawifs. <https://oceancolor.gsfc.nasa.gov/data/seawifs/>. Accessed: 24 05 2022.
- NASA Landsat Missions (2022). Landsat 8 data users handbook. <https://www.usgs.gov/landsat-missions/landsat-8-data-users-handbook>. Accessed: 06 04 2022.
- NASA Landsat Science (2013). The thematic mapper. <https://landsat.gsfc.nasa.gov/article/the-thematic-mapper/>. Accessed: 24 05 2022.
- NASA Modis (2022). Moderate resolution imaging spectroradiometer. <https://modis.gsfc.nasa.gov/about/>. Accessed: 24 05 2022.
- NASA Terra, The EOS Flagship (2022). Advanced spaceborne thermal emission and reflection radiometer. <https://terra.nasa.gov/about/terra-instruments/aster>. Accessed: 24 05 2022.
- National Geospatial Intelligence Agency (2022). Wgs 84 data apps. <https://earth-info.nga.mil/>. Accessed: 25 05 2022.
- Neale, C. M. U., Jayanthi, H., and Wright, J. L. (2005). Irrigation water management using high resolution airborne remote sensing. *Irrigation and Drainage Systems*, 19(3-4):321–336.
- Nellis, M. D., Price, K. P., and Rundquist, D. (2009). Remote sensing of cropland agriculture. *The SAGE handbook of remote sensing*, 1:368–380.
- Nguyen, H. T. and Smeulders, A. (2004). Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79.
- Nguyen, K.-A. and Liou, Y.-A. (2019). Mapping global eco-environment vulnerability due to human and nature disturbances. *MethodsX*, 6:862–875.
- Nguyen, V.-L., Shaker, M. H., and Hüllermeier, E. (2022). How to measure uncertainty in uncertainty sampling for active learning. *Machine Learning*, 111(1):89–122.
- Nichols, W. E. and Cuenca, R. H. (1993). Evaluation of the evaporative fraction for parameterization of the surface energy balance. *Water Resources Research*, 29(11):3681–3690.
- Nicholson, S. E. and Farrar, T. J. (1994). The influence of soil type on the relationships between ndvi, rainfall, and soil moisture in semiarid botswana. i. ndvi response to rainfall. *Remote Sensing of Environment*, 50(2):107–120.
- Nouretdinov, I., Costafreda, S. G., Gammerman, A., Chervonenkis, A., Vovk, V., Vapnik, V., and Fu, C. H. (2011). Machine learning classification with confidence: Application of transductive conformal predictors to mri-based diagnostic and prognostic markers in depression. *NeuroImage*, 56(2):809–813. Multivariate Decoding and Brain Reading.
- Ojala, T., Pietikainen, M., and Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987.
- Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175.

- Pal, M. and Mather, P. M. (2003). An assessment of the effectiveness of decision tree methods for land cover classification. *Remote sensing of environment*, 86(4):554–565.
- Papathanasiou, V. (1993). Some characteristic properties of the fisher information matrix via cacoullos-type inequalities. *Journal of Multivariate analysis*, 44(2):256–265.
- Pareeth, S. et al. (2019). Mapping agricultural landuse patterns from time series of landsat 8 using random forest based hierarchial approach. *Remote Sensing*, 11:5.
- Pasolli, E., Melgani, F., Tuia, D., Pacifici, F., and Emery, W. J. (2013). Svm active learning approach for image classification using spatial information. *IEEE Transactions on Geoscience and Remote Sensing*, 52(4):2217–2233.
- Patel, J. H. and Oza, M. P. (2014). Deriving crop calendar using ndvi time-series. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40:8.
- Patel, J. K. and Read, C. B. (1996). *Handbook of the normal distribution*, volume 150. CRC Press.
- Patra, S. and Bruzzone, L. (2010). A fast cluster-assumption based active-learning technique for classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(5):1617–1626.
- Patra, S. and Bruzzone, L. (2011). A batch-mode active learning technique based on multiple uncertainty for svm classifier. *IEEE Geoscience and Remote Sensing Letters*, 9(3):497–501.
- Penatti, O. A., Nogueira, K., and Dos Santos, J. A. (2015). Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 44–51.
- Petrucci, B., Huc, M., Feuvrier, T., Ruffel, C., Hagolle, O., Lonjou, V., and Desjardins, C. (2015). Maccs: Multi-mission atmospheric correction and cloud screening tool for high-frequency revisit data processing. In *Image and Signal Processing for Remote Sensing XXI*, volume 9643, page 964307. International Society for Optics and Photonics.
- Phiri, D., Simwanda, M., Salekin, S., Nyirenda, V. R., Murayama, Y., and Ranagalage, M. (2020). Sentinel-2 data for land cover/use mapping: A review. *Remote Sensing*, 12(14):2291.
- Pichón, F. J. (1997). Colonist land-allocation decisions, land use, and deforestation in the ecuadorian amazon frontier. *Economic Development and Cultural Change*, 45(4):707–744.
- Pierce, F. J. and Nowak, P. (1999). Aspects of precision agriculture. In *Advances in agronomy*, volume 67, pages 1–85. Elsevier.
- Pijanowski, B. C., Brown, D. G., Shellito, B. A., and Manik, G. A. (2002). Using neural networks and gis to forecast land use changes: a land transformation model. *Computers, environment and urban systems*, 26(6):553–575.
- Poggi, M., Tosi, F., and Mattoccia, S. (2017a). Even more confident predictions with deep machine-learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

- Poggi, M., Tosi, F., and Mattochia, S. (2017b). Quantitative evaluation of confidence measures in a machine learning world. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- Polasky, S. et al. (2008). Where to put things? spatial land management to sustain biodiversity and economic returns. *Biological conservation*, 141(6):1505–1524.
- Polykretis, C., Grillakis, M. G., and Alexakis, D. D. (2020). Exploring the impact of various spectral indices on land cover change detection using change vector analysis: A case study of crete island, greece. *Remote Sensing*, 12(2).
- Popescu, A., Faur, D., Vaduva, C., and Datcu, M. (2016). Enhanced classification of land cover through joint analysis of sentinel-1 and sentinel-2 data. In *Proc. ESA Living Planet Symp.*, pages 9–13.
- Pradhan, B., Tehrany, M. S., and Jebur, M. N. (2016). A new semiautomated detection mapping of flood extent from terrasars-x satellite image using rule-based classification and taguchi optimization techniques. *IEEE Transactions on Geoscience and Remote Sensing*, 54(7):4331–4342.
- Prasad, A. K., Chai, L., Singh, R. P., and Kafatos, M. (2006). Crop yield estimation model for iowa using remote sensing and surface parameters. *International Journal of Applied earth observation and geoinformation*, 8(1):26–33.
- Qiu, S., Zhu, Z., and He, B. (2019). Fmask 4.0: Improved cloud and cloud shadow detection in landsats 4–8 and sentinel-2 imagery. *Remote Sensing of Environment*, 231:111205.
- Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys (CSUR)*, 28(1):71–72.
- Rahman, A., Kumar, S., Fazal, S., and Siddiqui, M. A. (2012). Assessment of land use/land cover change in the north-west district of delhi using remote sensing and gis techniques. *Journal of the Indian Society of Remote Sensing*, 40(4):689–697.
- Raiyani, K. (2023). Sentinel 2 image scene classifier. https://github.com/kraiyani/Sentinel_2_image_scene_classifier. Accessed: 24 02 2023.
- Raiyani, K., Gonçalves, T., and Rato, L. (2022a). Abbreviating labelling cost for sentinel-2 image scene classification through active learning. In Pinho, A. J., Georgieva, P., Teixeira, L. F., and Sánchez, J. A., editors, *Pattern Recognition and Image Analysis*, pages 295–308, Cham. Springer International Publishing.
- Raiyani, K., Gonçalves, T., Rato, L., and Barão, M. (2022b). Mahalanobis distance based accuracy prediction models for sentinel-2 image scene classification. *International Journal of Remote Sensing*, 0(0):1–26.
- Raiyani, K., Gonçalves, T., Rato, L., Salgueiro, P., and Marques da Silva, J. R. (2021). Sentinel-2 image scene classification: A comparison between sen2cor and a machine learning approach. *Remote Sensing*, 13(2).
- Rajan, S., Ghosh, J., and Crawford, M. M. (2008). An active learning approach to hyperspectral data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 46(4):1231–1242.

- Read, T. R. and Cressie, N. A. (2012). *Goodness-of-fit statistics for discrete multivariate data*. Springer Science & Business Media.
- Reed, B. C. et al. (1994). Measuring phenological variability from satellite imagery. *Journal of vegetation science*, 5(5):703–714.
- Rees, G. and Rees, W. G. (1999). *The remote sensing data book*. Cambridge university press.
- Richards, J. A. and Jia, X. (2006). Image classification methodologies. *Remote sensing digital image analysis: An introduction*, pages 295–332.
- Riedell, W., Osborne, S., and Hesler, L. (2005). Insect pest and disease detection using remote sensing techniques. In *Proceedings of the 7th International Conference on Precision Agriculture*.
- Rouse, J., Haas, R., Schell, J., and Deering, D. (1974). Monitoring vegetation systems in the great plains with erts. *NASA special publication*, 351:309.
- Rouse Jr, J., Haas, R., Schell, J., and Deering, D. (1973). Paper a 20. In *Third Earth Resources Technology Satellite-1 Symposium: Section AB. Technical presentations*, volume 1, page 309. Scientific and Technical Information Office, National Aeronautics and Space .
- Roy, N. and McCallum, A. (2001a). Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown*, 2:441–448.
- Roy, N. and McCallum, A. (2001b). Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01*, page 441448, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Roy, P. S., Roy, A., Joshi, P. K., Kale, M. P., Srivastava, V. K., Srivastava, S. K., Dwivedi, R. S., Joshi, C., Behera, M. D., Meiyappan, P., et al. (2015). Development of decadal (1985–1995–2005) land use and land cover database for india. *Remote Sensing*, 7(3):2401–2430.
- Royall, R. (2004). The likelihood paradigm for statistical evidence. *The nature of scientific evidence: Statistical, philosophical, and empirical considerations*, pages 119–152.
- Rujoiu-Mare, M.-R., Olariu, B., Mihai, B.-A., Nistor, C., and Săvulescu, I. (2017). Land cover classification in romanian carpathians and subcarpathians using multi-date sentinel-2 remote sensing imagery. *European Journal of Remote Sensing*, 50(1):496–508.
- Russell, S. J. and Norvig, P. (2010). *Artificial intelligence : a modern approach*. Third edition. Upper Saddle River, N.J. : Prentice Hall, [2010] ©2010. Includes bibliographical references (pages 1063-1093) and index.
- Rute, J. (2016). When does randomness come from randomness? *Theoretical Computer Science*, 635:35–50.
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. (2016). Improved techniques for training gans. *Advances in neural information processing systems*, 29.

- San, B. T. (2014). International journal of applied earth observation and geoinformation. *International Journal of Applied Earth Observation and Geoinformation*, 26:399–412.
- Satellite Imaging Corporation (2022a). Ikonos satellite sensor. <https://www.satimagingcorp.com/satellite-sensors/ikonos/>. Accessed: 06 04 2022.
- Satellite Imaging Corporation (2022b). Quickbird satellite sensor. <https://www.satimagingcorp.com/satellite-sensors/quickbird/>. Accessed: 06 04 2022.
- Schohn, G. and Cohn, D. (2000). Less is more: Active learning with support vector machines. In *ICML*, volume 2, page 6. Citeseer.
- Schütze, H., Velipasaoglu, E., and Pedersen, J. O. (2006). Performance thresholding in practical text classification. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 662–671.
- Seifert, C. and Granitzer, M. (2010). User-based active learning. In *2010 IEEE International Conference on Data Mining Workshops*, pages 418–425. IEEE.
- Seiler, R. A., Kogan, F., and Wei, G. (2000a). Monitoring weather impact and crop yield from noaa avhrr data in argentina. *Advances in Space Research*, 26(7):1177–1185.
- Seiler, R. A., Kogan, F., and Wei, G. (2000b). Monitoring weather impact and crop yield from noaa avhrr data in argentina. *Advances in Space Research*, 26(7):1177–1185.
- Sekertekin, A., Marangoz, A., and Akcin, H. (2017). Pixel-based classification analysis of land use land cover using sentinel-2 and landsat-8 data. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, 42:91–93.
- Settles, B. (2009). Active learning literature survey. In *CS Technical Reports*. University of Wisconsin-Madison Department of Computer Sciences.
- Settles, B. (2010). Active learning literature survey. university of wisconsin. *Computer Science Department*.
- Settles, B. (2011). From theories to queries: Active learning in practice. In *Active learning and experimental design workshop in conjunction with AISTATS 2010*, pages 1–18. JMLR Workshop and Conference Proceedings.
- Settles, B. and Craven, M. (2008). An analysis of active learning strategies for sequence labeling tasks. In *proceedings of the 2008 conference on empirical methods in natural language processing*, pages 1070–1079.
- Settles, B., Craven, M., and Friedland, L. (2008). Active learning with real annotation costs. In *Proceedings of the NIPS workshop on cost-sensitive learning*, volume 1. Vancouver, CA:.
- Settles, B., Craven, M., and Ray, S. (2007). Multiple-instance active learning. *Advances in neural information processing systems*, 20.
- Seung, H. S., Opper, M., and Sompolinsky, H. (1992). Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294.
- Shanmugapriya, P., Rathika, S., Ramesh, T., and Janaki, P. (2019). Applications of remote sensing in agriculture—a review. *International Journal of Current Microbiology and Applied Sciences*, 8(1):2270–2283.

- Shannon, C. E. (1948a). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.
- Shannon, C. E. (1948b). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.
- Shao, J., Wang, Q., and Liu, F. (2019). Learning to sample: an active learning framework. *CoRR*, abs/1909.03585.
- Sharma, M. and Bilgic, M. (2017). Evidence-based uncertainty sampling for active learning. *Data Mining and Knowledge Discovery*, 31(1):164–202.
- Siddiqui, A. R. (2003). Regional evaluation of desertification hazards in the arid lands of western rajasthan. *Department of Geography*.
- Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Singh, A. (1989). Review article digital change detection techniques using remotely-sensed data. *International journal of remote sensing*, 10(6):989–1003.
- Smith, S. L., Kindermans, P.-J., Ying, C., and Le, Q. V. (2017). Don’t decay the learning rate, increase the batch size. *arXiv preprint arXiv:1711.00489*.
- Snyder, J. S. et al. (2005). A role for adult neurogenesis in spatial long-term memory. *Neuroscience*, 130(4):843–852.
- Sruthi, S. and Aslam, M. M. (2015). Agricultural drought analysis using the ndvi and land surface temperature data; a case study of raichur district. *Aquatic Procedia*, 4:1258–1264.
- Sugrue, L. P., Corrado, G. S., and Newsome, W. T. (2005). Choosing the greater of two goods: neural currencies for valuation and decision making. *Nature Reviews Neuroscience*, 6(5):363–375.
- Sumbul, G., Charfuelan, M., Demir, B., and Markl, V. (2019). Bigearthnet: A large-scale benchmark archive for remote sensing image understanding. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 5901–5904. IEEE.
- Sun, S. and Hardoon, D. R. (2010). Active learning with extremely sparse labeled examples. *Neurocomputing*, 73(16-18):2980–2988.
- Survey, U. S. G. (2022). What is lidar data and where can i download it? <https://www.usgs.gov/faqs/what-lidar-data-and-where-can-i-download-it>. Accessed: 06 04 2022.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.
- Tavazza, F., DeCost, B., and Choudhary, K. (2021). Uncertainty prediction for machine learning models of material properties. *ACS omega*, 6(48):32431–32440.
- Thompson, C. A., Califf, M. E., and Mooney, R. J. (1999). Active learning for natural language parsing and information extraction. In *ICML*, pages 406–414. Citeseer.
- Tibshirani, R. (1996). *Bias, variance and prediction error for classification rules*. Citeseer.

- Tilman, D., Balzer, C., Hill, J., and Befort, B. L. (2011). Global food demand and the sustainable intensification of agriculture. *Proceedings of the national academy of sciences*, 108(50):20260–20264.
- Tomanek, K. and Olsson, F. (2009). A web survey on the use of active learning to support annotation of text data. In *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pages 45–48.
- Tomanek, K., Wermter, J., and Hahn, U. (2007). An approach to text corpus construction which cuts annotation costs and maintains reusability of annotated data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 486–495.
- Tomowski, D., Ehlers, M., and Klonus, S. (2011). Colour and texture based change detection for urban disaster analysis. In *2011 Joint Urban Remote Sensing Event*, pages 329–332. IEEE.
- Tong, S. and Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66.
- Tracewski, L., Bastin, L., and Fonte, C. C. (2017). Repurposing a deep learning network to filter and classify volunteered photographs for land cover and land use characterization. *Geo-spatial information science*, 20(3):252–268.
- Tuia, D., Volpi, M., Copa, L., Kanevski, M., and Munoz-Mari, J. (2011). A survey of active learning algorithms for supervised remote sensing image classification. *IEEE Journal of Selected Topics in Signal Processing*, 5(3):606–617.
- Tur, G., Hakkani-Tür, D., and Schapire, R. E. (2005). Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45(2):171–186.
- Turner, W., Spector, S., Gardiner, N., Fladeland, M., Sterling, E., and Steininger, M. (2003). Remote sensing for biodiversity science and conservation. *Trends in ecology & evolution*, 18(6):306–314.
- UNESCO (2013). World population prospects the 2012 revision. *United Nations Department of Economic and Social Affairs: New York, NY, USA*.
- Upreti, D., Huang, W., Kong, W., Pascucci, S., Pignatti, S., Zhou, X., Ye, H., and Casa, R. (2019). A comparison of hybrid machine learning algorithms for the retrieval of wheat biophysical variables from sentinel-2. *Remote Sensing*, 11(5):481.
- Van Aken, J. E. (2005). Management research as a design science: Articulating the research products of mode 2 knowledge production in management. *British journal of management*, 16(1):19–36.
- Van der Meer, F. D., Van der Werff, H. M., Van Ruitenbeek, F. J., Hecker, C. A., Bakker, W. H., Noomen, M. F., Van Der Meijde, M., Carranza, E. J. M., De Smeth, J. B., and Woldai, T. (2012). Multi-and hyperspectral geologic remote sensing: A review. *International Journal of Applied Earth Observation and Geoinformation*, 14(1):112–128.
- Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., and Belongie, S. (2018). The inaturalist species classification and detection dataset.

- In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778.
- Vapnik, V. (1998). The support vector method of function estimation. In *Nonlinear modeling*, pages 55–85. Springer.
- Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999.
- Vatsavai, R. R., Bright, E., Varun, C., Budhendra, B., Cheriyyadat, A., and Grasser, J. (2011). Machine learning approaches for high-resolution urban land cover classification: a comparative study. In *Proceedings of the 2nd international conference on computing for geospatial research & applications*, pages 1–10.
- Vermeulen, G., Tullberg, J., and Chamen, W. (2010). Controlled traffic farming. In *Soil engineering*, pages 101–120. Springer.
- Verrelst, J., Berger, K., and Rivera-Caicedo, J. P. (2020). Intelligent sampling for vegetation nitrogen mapping based on hybrid machine learning algorithms. *IEEE Geoscience and Remote Sensing Letters*, 18(12):2038–2042.
- Verrelst, J., Dethier, S., Rivera, J. P., Munoz-Mari, J., Camps-Valls, G., and Moreno, J. (2016). Active learning methods for efficient hybrid biophysical variable retrieval. *IEEE Geoscience and Remote Sensing Letters*, 13(7):1012–1016.
- Veysi, S., Naseri, A. A., Hamzeh, S., and Bartholomeus, H. (2017). A satellite based crop water stress index for irrigation scheduling in sugarcane fields. *Agricultural water management*, 189:70–86.
- Vickers, A. J. (2005). Parametric versus non-parametric statistics in the analysis of randomized trials with non-normally distributed data. *BMC medical research methodology*, 5(1):1–12.
- Vidal, A. et al. (1994). Evaluation of a temporal fire risk index in mediterranean forests from noaa thermal ir. *Remote Sensing of Environment*, 49(3):296–303.
- Vijayanarasimhan, S. and Grauman, K. (2009). What’s it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations. In *2009 IEEE conference on computer vision and pattern recognition*, pages 2262–2269. IEEE.
- Vila, J. P. S. and Barbosa, P. (2010). Post-fire vegetation regrowth detection in the deiva marina region (liguria-italy) using landsat tm and etm+ data. *Ecological Modelling*, 221(1):75–84.
- Vitányi, P. M. and Li, M. (2000). Minimum description length induction, bayesianism, and kolmogorov complexity. *IEEE Transactions on information theory*, 46(2):446–464.
- Voordijk, H. (2009). Construction management and economics: the epistemology of a multidisciplinary design science. *Construction management and economics*, 27(8):713–720.
- Vuolo, F., Neuwirth, M., Immitzer, M., Atzberger, C., and Ng, W.-T. (2018). How much does multi-temporal sentinel-2 data improve crop type classification? *International journal of applied earth observation and geoinformation*, 72:122–130.

- Wallace, J. F., Caccetta, P. A., and Kiiveri, H. T. (2004). Recent developments in analysis of spatial and temporal data for landscape qualities and monitoring. *Austral Ecology*, 29(1):100–107.
- Wan, J., Wang, D., Hoi, S. C. H., Wu, P., Zhu, J., Zhang, Y., and Li, J. (2014). Deep learning for content-based image retrieval: A comprehensive study. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 157–166.
- Wang, J., Price, K. P., and Rich, P. M. (2001). Spatial patterns of ndvi in response to precipitation and temperature in the central great plains. *International Journal of Remote Sensing*, 22(18):3827–3844.
- Wang, L., Sousa, W., and Gong, P. (2004). Integration of object-based and pixel-based classification for mapping mangroves with ikonos imagery. *International Journal of Remote Sensing*, 25(24):5655–5668.
- Wang, Z. and Brenning, A. (2021). Active-learning approaches for landslide mapping using support vector machines. *Remote Sensing*, 13(13):2588.
- Warmuth, M. K., Liao, J., Rätsch, G., Mathieson, M., Putta, S., and Lemmen, C. (2003). Active learning with support vector machines in the drug discovery process. *Journal of chemical information and computer sciences*, 43(2):667–673.
- Warmuth, M. K., Rätsch, G., Mathieson, M., Liao, J., and Lemmen, C. (2001). Active learning in the drug discovery process. *Advances in Neural information processing systems*, 14.
- Welch, B. L. (1939). Note on discriminant functions. *Biometrika*, 31(1/2):218–220.
- Wielicki, B. A., Barkstrom, B. R., Harrison, E. F., Lee III, R. B., Smith, G. L., and Cooper, J. E. (1996). Clouds and the earth’s radiant energy system (ceres): An earth observing system experiment. *Bulletin of the American Meteorological Society*, 77(5):853–868.
- Wigmore, I. (2022). Prediction error. <https://www.techtarget.com/whatis/definition/prediction-error>. Accessed: 21 04 2022.
- Wilhite, D. A. and Glantz, M. H. (1985). Understanding: the drought phenomenon: the role of definitions. *Water international*, 10(3):111–120.
- Williams, D. L., Stauffer, M. L., and Leung, K. (1979). A forester’s look at the application of image manipulation techniques to multitemporal landsat data. In *LARS SYMPOSIA*.
- Wilson, E. H. and Sader, S. A. (2002). Detection of forest harvest type using multiple dates of landsat tm imagery. *Remote Sensing of Environment*, 80(3):385–396.
- Wong, T.-T. (1998). Generalized dirichlet distribution in bayesian analysis. *Applied Mathematics and Computation*, 97(2-3):165–181.
- Woodcock, C. E., Macomber, S. A., Pax-Lenney, M., and Cohen, W. B. (2001). Monitoring large areas for forest change using landsat: Generalization across space, time and landsat sensors. *Remote sensing of environment*, 78(1-2):194–203.
- Wootters, W. K. (1981a). Statistical distance and hilbert space. *Phys. Rev. D*, 23:357–362.
- Wootters, W. K. (1981b). Statistical distance and hilbert space. *Physical Review D*, 23(2):357.

- Wright, C. and Gallant, A. (2007). Improved wetland remote sensing in yellowstone national park using classification trees to combine tm imagery and ancillary environmental data. *Remote Sensing of Environment*, 107(4):582–605.
- Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., and He, L. (2021). A survey of human-in-the-loop for machine learning. *CoRR*, abs/2108.00941.
- Xia, B., Zhang, H., Li, Q., and Li, T. (2015). Pets: a stable and accurate predictor of protein-protein interacting sites based on extremely-randomized trees. *IEEE transactions on nanobioscience*, 14(8):882–893.
- Xia, G.-S., Hu, J., Hu, F., Shi, B., Bai, X., Zhong, Y., Zhang, L., and Lu, X. (2017). Aid: A benchmark data set for performance evaluation of aerial scene classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3965–3981.
- Xie, Y., Sha, Z., and Yu, M. (2008). Remote sensing imagery in vegetation mapping: a review. *Journal of plant ecology*, 1(1):9–23.
- Xiong, J., Thenkabail, P. S., Tilton, J. C., Gumma, M. K., Teluguntla, P., Oliphant, A., Congalton, R. G., Yadav, K., and Gorelick, N. (2017). Nominal 30-m cropland extent map of continental africa by integrating pixel-based and object-based algorithms using sentinel-2 and landsat-8 data on google earth engine. *Remote Sensing*, 9(10):1065.
- Xu, H. (2006). Modification of normalised difference water index (ndwi) to enhance open water features in remotely sensed imagery. *International journal of remote sensing*, 27(14):3025–3033.
- Xu, Z., Akella, R., and Zhang, Y. (2007). Incorporating diversity and density in active learning for relevance feedback. In *European Conference on Information Retrieval*, pages 246–257. Springer.
- Xu, Z., Yu, K., Tresp, V., Xu, X., and Wang, J. (2003). Representative sampling for text classification using support vector machines. In *European conference on information retrieval*, pages 393–407. Springer.
- Yang, J. et al. (2003). Automatically labeling video data using multi-class active learning. In *Proceedings Ninth IEEE international conference on computer vision*, pages 516–523. IEEE.
- Yang, L., MacEachren, A. M., Mitra, P., and Onorati, T. (2018). Visually-enabled active deep learning for (geo) text and image classification: a review. *ISPRS International Journal of Geo-Information*, 7(2):65.
- Yang, Y., Ma, Z., Nie, F., Chang, X., and Hauptmann, A. G. (2015). Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113(2):113–127.
- Yang, Y. and Newsam, S. (2010). Bag-of-visual-words and spatial extensions for land-use classification. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pages 270–279.
- Yang, Y., Yin, X., Zhao, Y., Lei, J., Li, W., and Shu, Z. (2021). Batch mode active learning based on multi-set clustering. *IEEE Access*, 9:51452–51463.

- Yu, G., Chen, X., Domeniconi, C., Wang, J., Li, Z., Zhang, Z., and Zhang, X. (2020). Cmal: Cost-effective multi-label active learning by querying subexamples. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1.
- Yu, H., Dai, Z., and Callan, J. (2021). Pgt: Pseudo relevance feedback using a graph-based transformer. In *European Conference on Information Retrieval*, pages 440–447. Springer.
- Yuan, W., Han, Y., Guan, D., Lee, S., and Lee, Y.-K. (2011). Initial training data selection for active learning. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication*, pages 1–7.
- Zarco-Tejada, P. J., Hubbard, N., and Loudjani, P. (2014). Precision agriculture: An opportunity for eu farmerspotential support with the cap 2014–2020. *Joint Research Centre (JRC) of the European Commission*.
- Zarco-Tejada, P. J., Rueda, C. A., and Ustin, S. L. (2003). Water content estimation in vegetation with modis reflectance data and model inversion methods. *Remote sensing of Environment*, 85(1):109–124.
- Zeng, L., Wardlow, B. D., Xiang, D., Hu, S., and Li, D. (2020). A review of vegetation phenological metrics extraction using time-series, multispectral satellite data. *Remote Sensing of Environment*, 237:111511.
- Zerrouki, N. and Bouchaffra, D. (2014). Pixel-based or object-based: Which approach is more appropriate for remote sensing image classification? In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 864–869. IEEE.
- Zhai, H., Zhang, H., Zhang, L., and Li, P. (2018). Cloud/shadow detection based on spectral indices for multi/hyperspectral optical remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 144:235–253.
- Zhan, N. and Kitchin, J. R. (2022). Uncertainty quantification in machine learning and nonlinear least squares regression models. *AIChE Journal*, 68(6):e17516.
- Zhang, C. and Chen, T. (2002). An active learning framework for content-based information retrieval. *IEEE transactions on multimedia*, 4(2):260–268.
- Zhang, C., Ren, H., Liang, Y., Liu, S., Qin, Q., and Ersoy, O. K. (2017). Advancing the prospect-5 model to simulate the spectral reflectance of copper-stressed leaves. *Remote Sensing*, 9(11):1191.
- Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., and Atkinson, P. M. (2019). Joint deep learning for land cover and land use classification. *Remote sensing of environment*, 221:173–187.
- Zhang, G. and Yi, L. (2012). Feature selection using rough set theory for object-oriented classification of remote sensing imagery. In *2012 20th International Conference on Geoinformatics*, pages 1–7. IEEE.
- Zhao, Y. et al. (2019). Long-term land cover dynamics (1986–2016) of northeast china derived from a multi-temporal landsat archive. *Remote Sensing*, 11:5.
- Zheng, H.-T., Wang, Z., Ma, N., Chen, J., Xiao, X., and Sangaiah, A. K. (2018). Weakly-supervised image captioning based on rich contextual information. *Multimedia Tools and Applications*, 77(14):18583–18599.

- Zhong, Y., Zhu, Q., and Zhang, L. (2015). Scene classification based on the multifeature fusion probabilistic topic model for high spatial resolution remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 53(11):6207–6222.
- Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. *Advances in neural information processing systems*, 27.
- Zhou, W., Newsam, S., Li, C., and Shao, Z. (2018). Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS journal of photogrammetry and remote sensing*, 145:197–209.
- Zhou, X., Zhang, J., Chen, D., Huang, Y., Kong, W., Yuan, L., Ye, H., and Huang, W. (2020). Assessment of leaf chlorophyll content models for winter wheat using landsat-8 multispectral remote sensing data. *Remote Sensing*, 12(16):2574.
- Zhu, X., Ghahramani, Z., and Lafferty, J. D. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pages 912–919.
- Zhu, X., Zhang, P., Lin, X., and Shi, Y. (2007). Active learning from data streams. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 757–762. IEEE.
- Zhu, X., Zhang, P., Lin, X., and Shi, Y. (2010). Active learning from stream data using optimal weight classifier ensemble. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(6):1607–1621.
- Zhu, X. J. (2005). Semi-supervised learning literature survey. In *CS Technical Reports*. University of Wisconsin-Madison Department of Computer Sciences.
- Zhu, Z. and Woodcock, C. E. (2012). Object-based cloud and cloud shadow detection in landsat imagery. *Remote sensing of environment*, 118:83–94.
- Zhu, Z. and Woodcock, C. E. (2014). Automated cloud, cloud shadow, and snow detection in multitemporal landsat data: An algorithm designed specifically for monitoring land cover change. *Remote Sensing of Environment*, 152:217–234.
- Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320.
- Zou, Q., Ni, L., Zhang, T., and Wang, Q. (2015). Deep learning based feature selection for remote sensing scene classification. *IEEE Geoscience and Remote Sensing Letters*, 12(11):2321–2325.