University of Windsor
Scholarship at UWindsor

Electronic Theses and Dissertations | Theses, Dissertations, and Major Papers

7-29-2020

Applications of Machine Learning in Spectrum Sensing for Cognitive Radios

Abdalaziz Ibrahim Osman Mohammed
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation

Mohammed, Abdalaziz Ibrahim Osman, "Applications of Machine Learning in Spectrum Sensing for Cognitive Radios" (2020). *Electronic Theses and Dissertations*. 8418.
https://scholar.uwindsor.ca/etd/8418

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Applications of Machine Learning in Spectrum Sensing for Cognitive Radios

By

Abdalaziz Mohammed

A Thesis

Submitted to the Faculty of Graduate Studies through the
Department of Electrical and Computer Engineering

in Partial Fulfilment of the
Requirements for the Degree of Master
of Applied Science at the University of
Windsor

Windsor, Ontario, Canada

2020

© 2020 Abdalaziz Mohammed

Application of Machine Learning in Spectrum Sensing for Cognitive Radios

Abdalaziz Mohammed

APPROVED BY:

____

A. Azab

Department of Mechanical, Automotive and Materials Engineering

____

M. Khalid

Department of Electrical and Computer Engineering

____

E. Abdel-Raheem, Advisor

Department of Electrical and Computer Engineering

June 8, 2020

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Spectrum sensing is an essential component in cognitive radios. The machine learning (ML) approach is part of artificial intelligence which develops systems capable of learning and improving from experience. ML algorithms are promising techniques for spectrum sensing as a favored solution to tackle the limitations of conventional spectrum sensing techniques while improving detection performance. The supervised ML algorithms, support vector machine (SVM), k-nearest neighbor (kNN), decision tree (DT), and ensemble are applied to detect the existence of primary users (PUs) in the TV spectrum band. This is accomplished by building classifiers using the collected data for the TV spectrum over different locations in the city of Windsor, Ontario. Then, the dimensionality reduction technique named Principal Component Analysis (PCA) is incorporated to reduce the duration of training and testing of the model, as well as reduce the risk of overfitting. This is achieved by transforming the input data into a lower-dimensional representation, which is known as the principal components. The Ensemble classification-based approach is employed to enhance the classifier predictivity and performance. Furthermore, the performance of the Ensemble classification method is compared with SVM, kNN, and DT classifiers. Simulation results have shown that the highest performance is achieved by combining multiple classifiers, i.e., the Ensemble, therefore, the detection performance has significantly improved. Simulation results have shown the impact of employing PCA on lowering the duration of training while maintaining the performance.

DEDICATION

To my lovely wife, and my son for their endless love,

encouragement, and support.

.

ACKNOWLEDGEMENTS

First and Foremost, all praises and thanks are due to almighty ALLAH, the Lord of the worlds, for his unlimited support in all times of my daily life and his guidance. I would love to express my deep and sincere gratitude to my supervisor Dr. Esam Abdel-Raheem for allowing me to do research and providing invaluable guidance throughout my study. I am grateful for his endless support, patience, and encouragement. I would like also to thank Dr. Faroq Awin for his support, motivation, valuable comments, and discussion. I would like to express appreciation to all my thesis committee members: Dr. Mohammed Khalid and Dr. Ahmed Azab for their valuable comments, notes, and discussion during my first seminar. I would like to acknowledge Mr. Aarron Younan and Dr. Kemal Tepe for providing me with the datasets. Finally, I would like to thank my lovely wife for believing in me and being at my side throughout my study; without your relentless support, patience, and encouragement, I could not have accomplished this.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AdaBoost	Adaptive Boosting
AI	Artificial Intelligence
ATSC	Advanced Television Systems Committee
Bagging	Bootstrap Aggregation
BPSK	Binary Phase Shift Keying
BW	Bandwidth
CR	Cognitive Radio
CRN	Cognitive Radio Network
CSS	Cooperative spectrum sensing
dB	Decibels
dBFS	Decibels relative to full scale
DR	Dimensionality Reduction
DSS	dynamic spectrum sharing
DT	Decision Trees

DTV	Digital Television
GUI	Graphical User Interface
FFT	Fast Fourier Transform
FP	False Positive
FPR	False Positive Rate
FN	False Negative
HDF	Hierarchical Data Format
HMM	Hidden Markov Model
ID3	Iterative Dichotomiser 3
kNN	k-Nearest Neighbors
LS	Leaf Size
MDP	Markov Decision Process
ML	Machine Learning
MSK	Minimum Shift Keying
PCA	Principal Components Analysis
P_d	Probability of detection
P_f	Probability of false alarm
PU	Primary user
QoS	Quality of Service

RBF	Radial Basis Function
RF	Radio Frequency
ROC	Receiver operating characteristic
RSP	Received signal power
SDR	Software-Defined Radios
SNR	Signal to noise ratio
SU	Secondary user
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TP	True Positive
TPR	True Positive Rate
TN	True Negatives
TNR	True Negative Rate
TV	Television
TVWS	Television White Space
UHD	USRP Hardware Driver
UHF	Ultra-High Frequency
USRP	Universal Serial Radio Peripheral
VHF	Very high frequency

VSB	Vestigial sideband modulation
WRAN	Wireless Regional Area Network
WSD	White Space Database

CHAPTER1

Introduction

The rapid transformation in communication from voice to digital media has led to a huge demand for higher bandwidth. Due to the constraints of the existing frequency spectrum assignment, it becomes evident that the current static frequency sharing schemes cannot fully meet the demands of a rising number of higher data-rate applications. Therefore, new approaches are required to introduce new ways to efficiently reutilize the usable spectrum. Cognitive radio (CR) is a promising resolution to the issue of frequency band underutilization by adopting the opportunistic use of parts of the spectrum not heavily occupied by licensed users. Spectrum sensing is one of the most essential elements in CR networks where each CR user must detect licensed users also known as primary users (PUs) if they are available and pinpoint the unusable spectrum when PUs is not present. Generally, it is fulfilled by sensing the RF environment [1,2].

The most important task of spectrum sensing is to refrain CR users from interfering with PUs whether by switching to an available band or by confining their interference with PUs at an acceptable level. Additionally, CR users should efficiently pinpoint and utilize the spectrum holes (SH), for the desired throughput and quality of service (QoS). Thus, the detection performance in spectrum sensing is vital to both PU and CR network performance. The detection performance of a CR user is assessed by two performance metrics, i.e., the probability of false alarm and probability of detection. The probability of false alarm, (P_f), is the probability of a CR user claiming a PU to be present when the spectrum band is free, while the probability of detection, (P_d), implies the possibility of a CR user claiming a PU to be present when the spectrum band is occupied. The failure in the detection will lead to interference with the PU and a false alarm will decrease the spectral efficiency [2].

Therefore, there is constantly a demand to improve the probability of detection and minimize the probability of false alarm through innovative approaches. In recent years, there has been a growing interest in adopting ML algorithms to CRs as a potential solution of sensing the spectrum to enhance the detection performance in low signal-to-noise ratio (SNR) scenarios. ML algorithms provide the CR the capability to learn when there is no prior knowledge of PU or noise with the help of the datasets collected from the RF environment. So, it will be prepared to assess the availability of the spectrum bands and be aware of the RF environment [3,4].

1.1 Motivation and Research Objectives

It is well-known that the frequency spectrum is congested. Exploiting the unoccupied TV spectrums, so-called TV white spaces (TVWS), is a tempting choice to meet the urgent needs. Even though cognitive radio networks have been widely researched as a probable solution for the spectrum scarcity and increasing spectrum underutilization, there are still some issues that need to be addressed. To utilize the spectrum effectively with the lowest interference to PUs, the secondary user (SU) should sense the spectrum band efficiently.

The detection performance of the conventional spectrum sensing techniques deteriorates when the signal-to-noise ratio (SNR) is low. Additionally, spectrum sensing techniques such as energy detector is very sensitive to noise uncertainty. Besides, some of the conventional spectrum sensing techniques require the prior knowledge of PU signal features which might not be available for CRs. For coping with this dilemma, ML algorithms can be regarded as efficient ways of sensing the spectrum without prior knowledge of the radio frequency (RF) environment. Furthermore, ML can periodically be learned to adapt to the changing RF environment. Moreover, the existing datasets generated from monitoring and observing the RF spectrum can be utilized to develop a spectrum sensing model.

The main objective of the research is to investigate the design of spectrum sensing models based on supervised ML techniques and dimensionality reduction techniques to assess the spectrum occupancy over the TV bands.

1.2 Thesis Contribution

The two main contributions of this thesis are:

1. A TV spectrum sensing model based on different supervised ML algorithms is proposed to accurately assess the spectrum occupancy over the TV channels and determine the identity of the user whether it is a PU or an SU. The proposed model is trained and tested using real data gathered over ten different locations across Essex County.
2. The proposed TV spectrum sensing model is modified to employ the PCA technique to reduce the dimensions of the gathered datasets, limit the risk of overfitting, and reduce the processing time of training and testing.

The performances of the two proposed models are evaluated and compared based on accuracy, F-measure, and Receiver Operating Characteristic (ROC).

1.3 Outline of the Thesis

The rest of the thesis is organized as follows: Chapter 2 provides an overview and background of CR networks, highlights of TVWS, IEEE 802.22, and provides an overview of ML Algorithms. Also, it introduces the proposed classification methods: SVM, kNN, DT, and Ensemble Classifications algorithms as well as a dimensionality reduction technique: PCA.

Chapter 3 presents the detailed research methodology, system model, data collection, and the proposed algorithms. Simulation results are illustrated in Chapter 4 where the performance measures of the classifiers are illustrated and compared with the proposed Ensemble classifier in terms of accuracy, F-measure with and without using PCA. Moreover, the ROC curves for the four classifiers are presented. Conclusions and future work are discussed in Chapter 5.

CHAPTER 2

Literature Overview

2.1 Introduction

With the rising usage of dynamic mobile applications, it is becoming decisive for wireless devices to learn from the surrounding environment. CR is described as radio devices able to learn and adapt to their environment. Spectrum sensing is a substantial component of CRs. In the last decade, several sensing techniques have been proposed based on matched filters, energy detection, pilot-sensing detection, cyclostationary detection, wavelet detection, and covariance detection. Moreover, cooperative spectrum sensing (CSS) was advocated as a means to improve sensing accuracy by tackling the inherent hidden terminal problems in wireless networks [5, 6].

In recent years, there has been growing interest in the applications of ML algorithms to CRs. A key element of any CR is the willingness to program themselves or to learn autonomously. As a result, CR is anticipated to be intelligent by nature. Learning is a substantial component of any intelligent system, which justifies it being designated as a fundamental requirement of CRs [7, 8].

There are three main conditions for intelligence: 1) Perception, 2) Learning, and 3) Reasoning and Act. First, perception can be accomplished via the sensing measurements of the spectrum. This permits the CR to detect ongoing RF activities in its surrounding environment. After acquiring the sensing observations, the CR attempts to learn from them to classify and organize the observations into suitable categories (knowledge). Finally, the reasoning ability allows the CR to use the knowledge acquired through learning to achieve its objectives. Therefore, CR must be equipped with the capability of learning from its experience by interacting with its RF environment by trying to make use of ML algorithms to coordinate the CR actions [9, 10].

2.2 Main Functions of Cognitive Radios

The spectrum management process in CR networks comprised of four key steps as presented in Fig. 2.1:

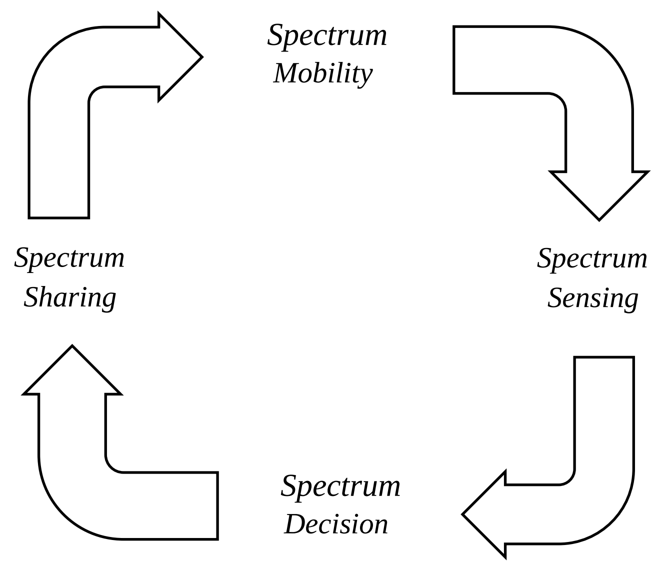


Figure 2.1: Main functions of cognitive radios [11].

- Spectrum sensing: CR users can only assign an unoccupied part of the spectrum. A CR user should then monitor the available spectrum bands, capture their information, and then identify the spectrum holes as depicted in Fig. 2.2.

- Spectrum decision: CR users can assign a channel subject to the availability of the spectrum. Channel allocation not only relies on the availability of spectrum, but it is also dictated based on internal and potentially external policies.

- Spectrum sharing: Since multiple CR users might be trying to access the spectrum, CR network access must be organized to avoid multiple users from colliding in overlapping segments of the spectrum.

- Spectrum mobility: Users of the CR are regarded as spectrum visitors. Thus, if a PU needs the specific portion of the spectrum in use, the communication must be continued in another unoccupied portion of the spectrum [12].

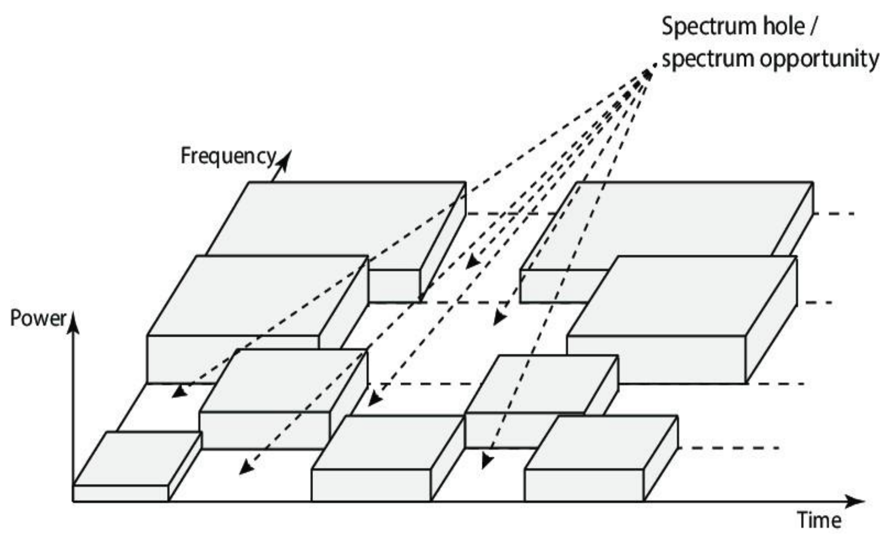


Figure 2.2: Demonstration of the spectrum hole [13].

2.3 TV White Space (TVWS)

TV spectrum portions that are not occupied by a PU at a given time and location are called TV white spectrum. The frequency bands allocated to TV bands change according to the geographical location. For example, in Europe, TVWS spans from 470 to 790 MHz while in the United States, it ranges from 54 to 698 MHz. The white space database (WSD) frequency bands in Canada are illustrated in Table 2.1. CR is being intensively researched as the enabling technology for the TVWS [14, 15].

Table 2.1: Overview of authorized white space frequency bands in Canada [16].

Frequency Bands (MHz)	TV Channels	Personal/Portable WSD	Fixed WSD
54-60	2	Not permitted	✓
60-72	3-4	Not permitted	Not permitted
76-88	5-6	Not permitted	✓
174-216	7-13	Not permitted	✓
470-512	14-20	✓	✓
512-608	21-36	✓	✓
608-614	37	Not permitted	Not permitted

2.3.1 Advanced Television Systems Committee (ATSC)

ATSC is a standard for digital TV (DTV) transmission which defines a system designed to distribute high-quality video and audio and ancillary signal within a 6 MHz digital TV channel. The 8 Vestigial sideband modulation (8VSB) used in ATSC (DTV) has a pilot carrier, which serves as a DTV receiver reference. The pilot carrier is about 310 kHz above the lower edge of the band as shown in Fig. 2.3 [17].

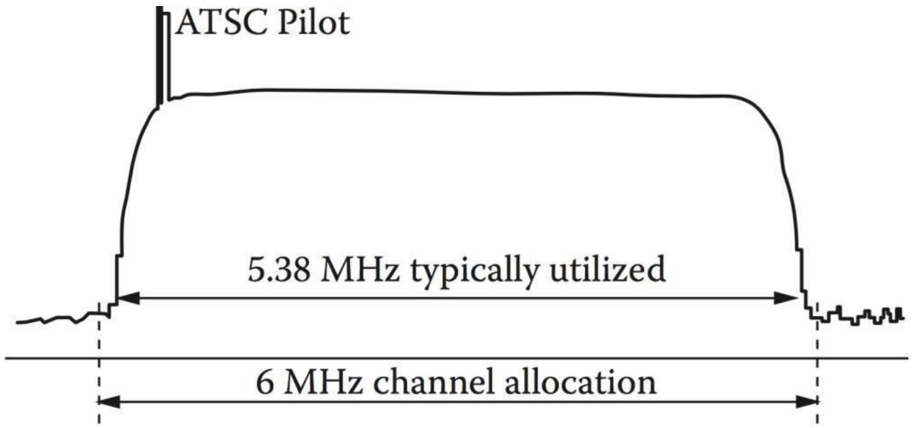


Figure 2.3: The spectrum of the ATSC signal [17].

2.4 IEEE 802.22

The IEEE 802.22 is a standard for Wireless Regional Area Network (WRAN) using white spaces in the TV frequency spectrum. The development of the IEEE 802.22 WRAN standard is intended at using CR methods to allow the sharing, on a non-interfering basis of the geographically unoccupied spectrum assigned to the television broadcasting service. It is the world's first attempt to define a standardized air interface based on CR techniques for the opportunistic use of non-interfering TV bands. IEEE 802.22 WRANs are designed to operate in TV broadcast bands while ensuring that no harmful interference is caused to the incumbent operation: digital TV and analog TV broadcasting, and low-power licensed devices such as wireless microphones [18].

2.5 Machine Learning

It is an artificial intelligence (AI) application that creates systems with the capacity to learn and improve automatically from experience without being explicitly programmed. ML focuses on computer programs being developed that can access data and use it to learn for themselves. The learning process begins with observations or data, such as examples, direct experience, or instruction to look for data patterns and make better decisions in the future. The main objective is to allow computers to automatically learn without human intervention or aid and to adjust actions accordingly [19].

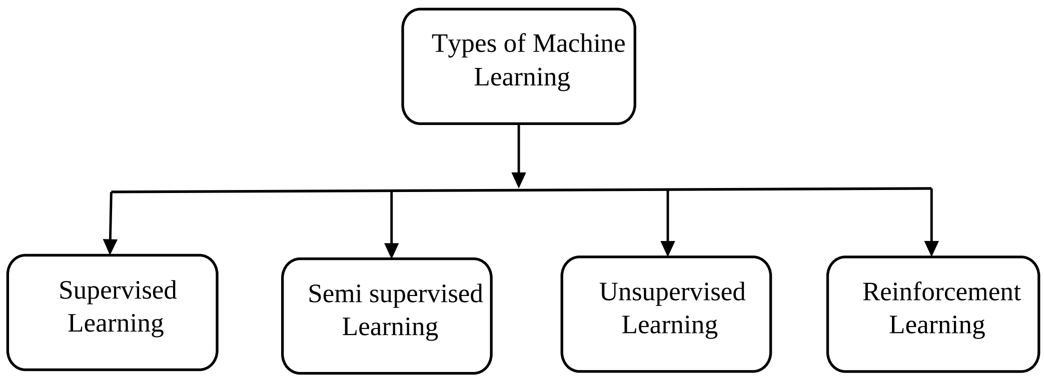


Figure 2.4: Classification of machine learning.

Various types of ML methods have been designed to solve problems in different fields. These ML techniques can be categorized into four types depending on the training method as depicted in Fig. 2.4. The supervised and supervised learning are discussed as the proposed ML algorithms fall under the two categories.

2.6 Supervised Learning

Supervised learning algorithms develop a mathematical model of a set of data comprising both the inputs and the needed outputs. The data is known as training data and contains training examples set. Every training example has one or more inputs and the required output. Each training example in the mathematical model is depicted by an array or vector, often named a feature vector and the training data is viewed as a matrix. Supervised learning algorithms learn a mapping function that can be used to predict the output correlated with new inputs. The mapping function will allow the algorithm to correctly determine the output for inputs that were not part of the training data. An algorithm is said to have learned to perform that task when it improves the accuracy of its outputs or predictions over time. Supervised learning algorithms include classification and regression which are presented in Fig. 2.5. SVM, kNN, DT, and Ensemble classifications are the most common supervised learning algorithms. If the CR has prior information about the RF environment, it might exploit this knowledge by using supervised learning techniques [20].

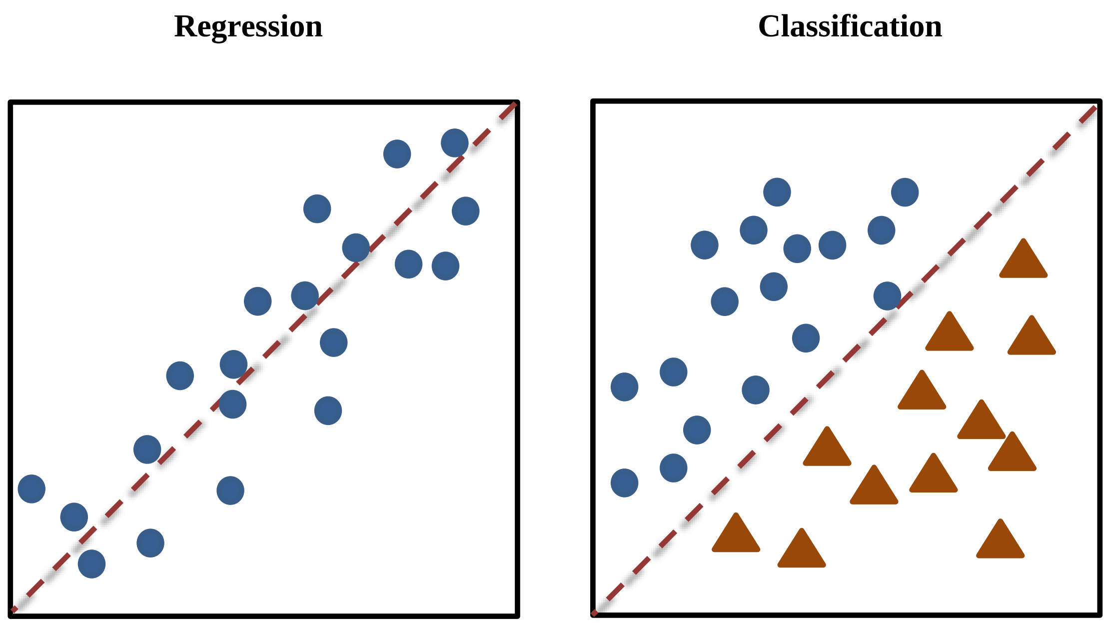


Figure 2.5: Outline of supervised learning (classification vs. regression).

2.6.1 Support Vector Machines (SVM)

The SVM classifies data by finding the linear decision boundary (hyperplane) that separates all data points of one class from those of the other class. The best hyperplane for an SVM is the one with the largest margin between the two classes. If the data is not linearly separable, SVM applies a kernel function to transform nonlinearly separable data into higher dimensions where a linear decision boundary can be found as depicted in Fig. 2.6 [21].

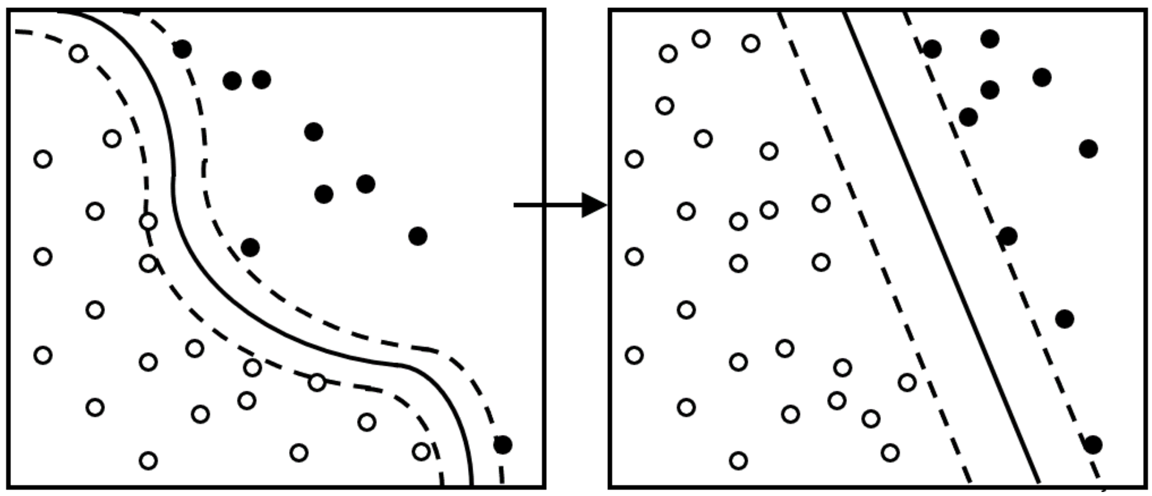


Figure 2.6: Basic idea of the kernel function in SVM.

Advantages:

- It is effective in high dimensional spaces (i.e. nonlinearly separable data).
- It employs the kernel function to solve classification problems.
- SVM models have generalization in practice.

Disadvantages:

- The performance is degrading when the data set has more noise (i.e. target classes are overlapping).
- Choosing a kernel function is not easy.
- Long training time for large datasets.

- It underperforms when the number of features for each data point exceeds the number of samples in the training data.

SVM uses several types of kernel such as linear, polynomial, radial basis (gaussian) as presented in Fig. 2.7.

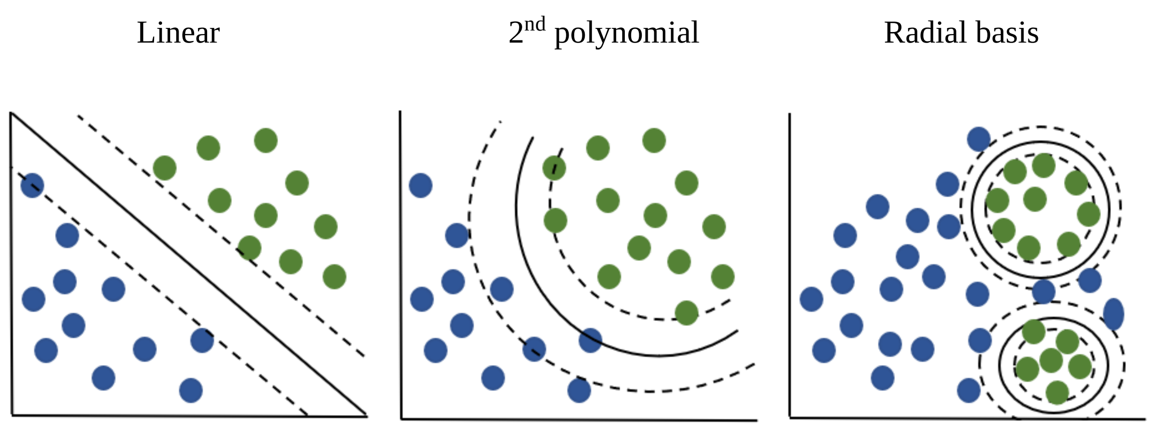


Figure 2.7: Linear, 2nd polynomial, and radial basis kernels.

2.6.2 k-Nearest-Neighbors (kNN)

kNN is a non-parametric technique used for classification in pattern recognition [22]. It is based on the fact that a data point in the learning dataset is classified by a majority vote of its k neighbors and gives greater weights for close neighbors in the classification than neighbors which are far from the data point. Figure 2.8 demonstrates assigning a class to data point when k = 1 and 3. Distance metrics such as Euclidean, city block, cosine, and Chebychev are used to find the nearest neighbor [22].

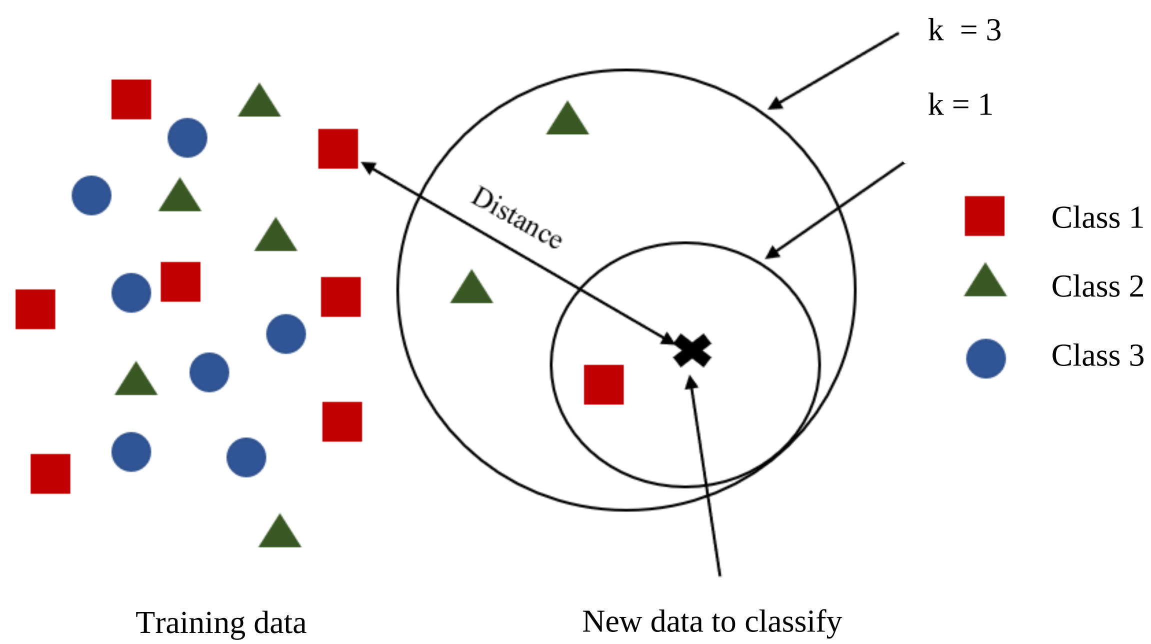


Figure 2.8: kNN classification algorithm.

Advantages:

- Simple in implementation.
- Robust and versatile.
- Few parameters to tune distance metric and k.

Disadvantages:

- Lazy learner.
- Slow if there are many training examples.
- Sensitive to k value.

2.6.3 Decision Tree (DT)

It classifies instances by ranking them from the root to some leaf node down the tree that produces classification. Each node in the tree determines a test of some instance attribute, and each branch coming down from that node relates to one of the potential values for that attribute. An instance is classified by launching at the tree's root node, examining the attribute specified by that node, and then going down the tree branch corresponding to the attribute value. The main algorithm for building DT is called ID3 (Iterative Dichotomiser 3). It deploys a top-down, greedy lookup for possible branches without backtracking. ID3 develops a decision tree utilizing entropy and information gain to select the attribute that is most essential for classifying. [23,24].

Advantages:

- Less effort for data preparation during pre-processing.
- Easy to use and interpret.
- It can be combined with other decision techniques.

Disadvantages:

- Lots of layers, which makes it complex.
- Overfitting issue.
- The computational complexity of multiclass may increase.

2.6.4 Ensemble Classification

Ensemble classification is the process of effectively generating and combining multiple classifiers to solve a specific ML problem as illustrated in Fig. 2.9. Ensemble learning is mainly used to improve the performance of a model in classification or prediction or to reduce the probability of a poor or unfortunate choice. Bagging and boosting are the most common methods used in Ensemble classification [25].

- Bootstrap Aggregation (Bagging)

Bagging is the most common type of Ensemble classification. It generally trains multiple independent classifiers, each trained by sampling with replacement percentage of instances from the training data. The diversity in the Ensemble is ensured by the variations in replicas on which each classifier is trained. The trained classifiers are then combined through a simple majority voting. It is best suited for problems with relatively small available training datasets [26].

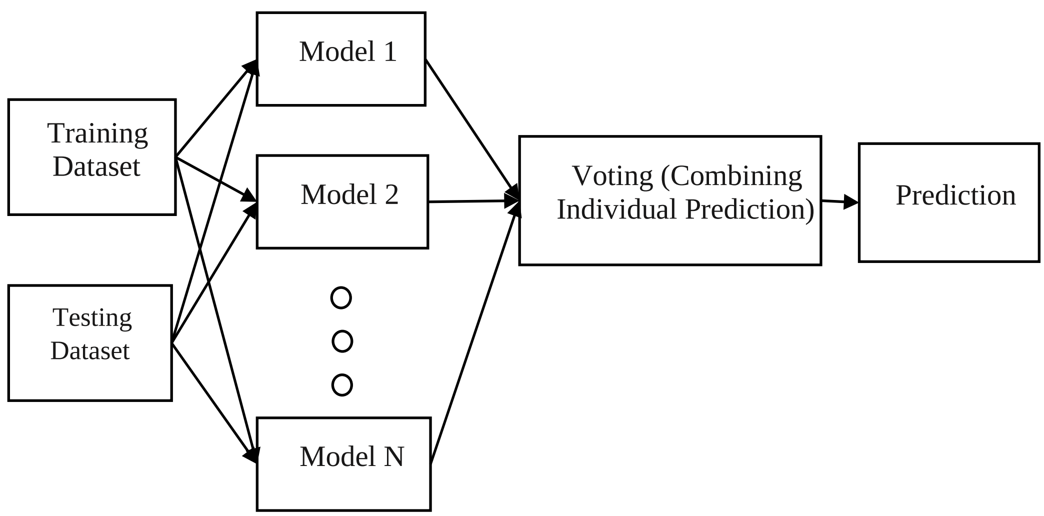


Figure 2.9: Basic outline of the Ensemble technique [31].

- Boosting and AdaBoost:

Boosting is an iterative method to produce a robust classifier that capable of arbitrarily attaining low training error from an Ensemble of weak classifiers. It also combines an Ensemble of weak classifiers using simple majority voting. The training dataset for each subsequent classifier is progressively focused on instances misclassified by the former classifiers [27].

Advantages:

- More accurate prediction results.
- Stable and more robust model.

Disadvantages:

- Higher computation.
- The selection of models for creating an Ensemble is not easy.

2.7 Unsupervised Learning

Unsupervised learning algorithms take a set of data that only comprises inputs, and try to find structure in the data, such as grouping or clustering data points. Thus, the algorithms learn from unlabeled data. Rather than reacting to feedback, unsupervised learning algorithms identify commonalities in the data and respond positively to each new piece of data based on the presence or absence of such commonalities. Clustering and PCA are two of the main techniques used in unsupervised learning. They are presented in Fig. 2.10 and 2.11, respectively. Unsupervised learning is suitable for CRs operating in alien RF environments [28].

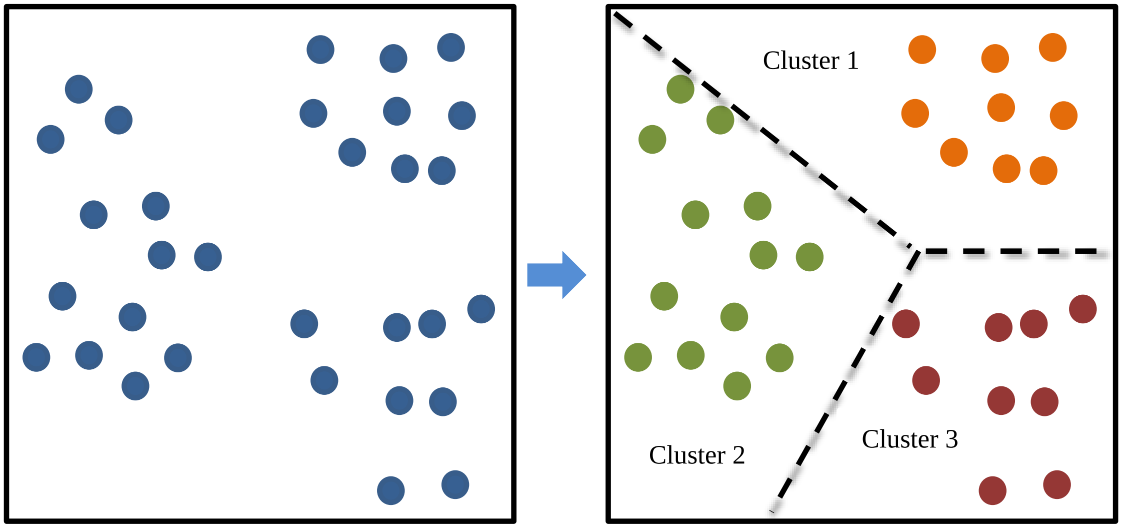


Figure 2.10: Outline of unsupervised learning (clustering).

2.7.1 Principal Component Analysis

PCA is by far the most common and popular unsupervised learning method for dimensionality reduction. It conducts dimensionality reduction by embedding the data into a linear subspace of lower dimensionality. A low-dimensional representation of the data is constructed by PCA that describes as much of the variance in the data as possible. It is attained by finding a linear basis of reduced dimensionality for the data in which the amount of variance in the data is maximal. Figure 2.11 illustrates finding two principal components that represent the new dimensions of the data points [29, 30].

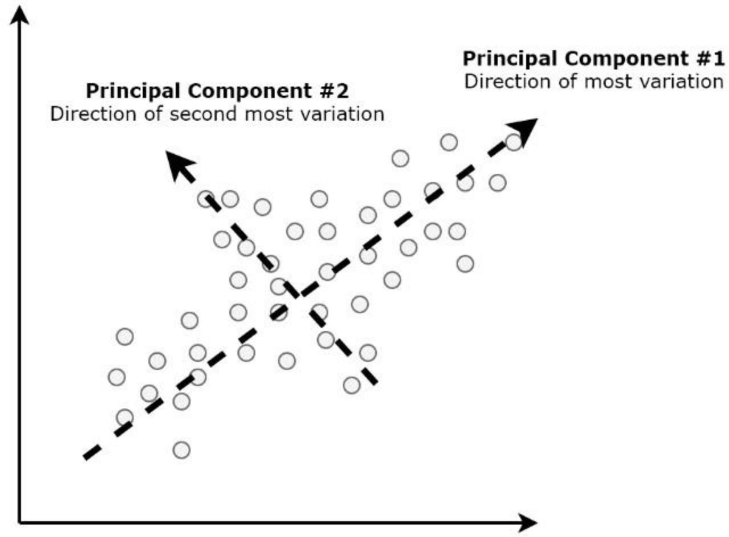


Figure 2.11: Outline of principal component analysis [30].

Advantages:

- Removes correlated features.
- Speed up the ML algorithm.
- Reduces overfitting.

Disadvantages:

- Principal components are not as readable and interpretable as original features.
- Sensitive to scaling, so data standardization is a must before PCA.
- Risk of missing information by selecting an improper number of principal components.

Unsupervised learning is mainly used in data analysis to find hidden patterns, groupings of data, and dimensionality reduction while supervised learning is best suited for labeled data. Since the datasets that will be used in creating the spectrum sensing models based on ML are already organized into groups and labeled. Hence, supervised learning classification methods are the appropriate approach. To take advantage of unsupervised learning, PCA is proposed to reduce the dimension of the datasets. The main reason for adopting the four classifiers is that the SVM model can be generalized, i.e., it doesn't overfit. Also, kNN tuning is simple through k value and the distance metric. Additionally, DT requires less effort in data preprocessing while the Ensemble classification boosts the prediction results. On the other hand, PCA minimizes the risk of overfitting and accelerates the learning process.

2.8 Summary

The main functions of cognitive radios are spectrum sensing, spectrum decision, spectrum sharing, and spectrum mobility. TVWS refers to spectrum portions that are not occupied by a PU. The two major categories of ML are supervised and unsupervised learning. Supervised learning deals with labeled data. The most popular supervised learning algorithms are SVM, kNN, DT, and Ensemble classification techniques. On the other hand, unsupervised learning attempts to find a hidden pattern in data consists of only inputs and group it into clusters. PCA can be classified as unsupervised learning.

CHAPTER 3

Research Methodology

3.1 Introduction

Preprocessing of the ATSC channels datasets is the preliminary phase of creating the ML models. It incorporates removing outliers such as spikes in the Fast Fourier transform (FFT) samples caused by artifact inherent to software-defined radio (SDR) employing homodyne receivers and features scaling using normalization. Combining a dimensionality reduction method, PCA with the four classifiers is explored as a second approach to accelerate the training and testing of the classifiers. The ML algorithms analyze the training data and create an inferred function which can be used for mapping new instances. An optimal scenario will allow for the algorithms to correctly determine the class labels of the testing data. Finally, the performance of the classifiers is evaluated using the performance metrics.

3.2 ATSC Channels Datasets

3.2.1 Collection Sites

The collection of datasets was undertaken at ten different sites across Essex County, Ontario as illustrated in Fig. 3.1. The data collection script is based on a closed-loop GNU Radio flowgraph. It is a free and open-source software development toolkit that provides signal processing blocks to implement software radios. The Universal Serial Radio

Peripheral (USRP) hardware driver source provides outputs of values that are proportional to the in-phase and quadrature (IQ) components of the voltage seen at the USRP's antenna terminal. Datasets comprise of FFT samples at given timesteps equal 171 microseconds and amplitude relative to the full scale of the USRP as presented in Figs.3.2-3.6. The datasets cover the TV channels band from 490 to 740 MHz, gathered at each site to be employed as training and testing data [31].

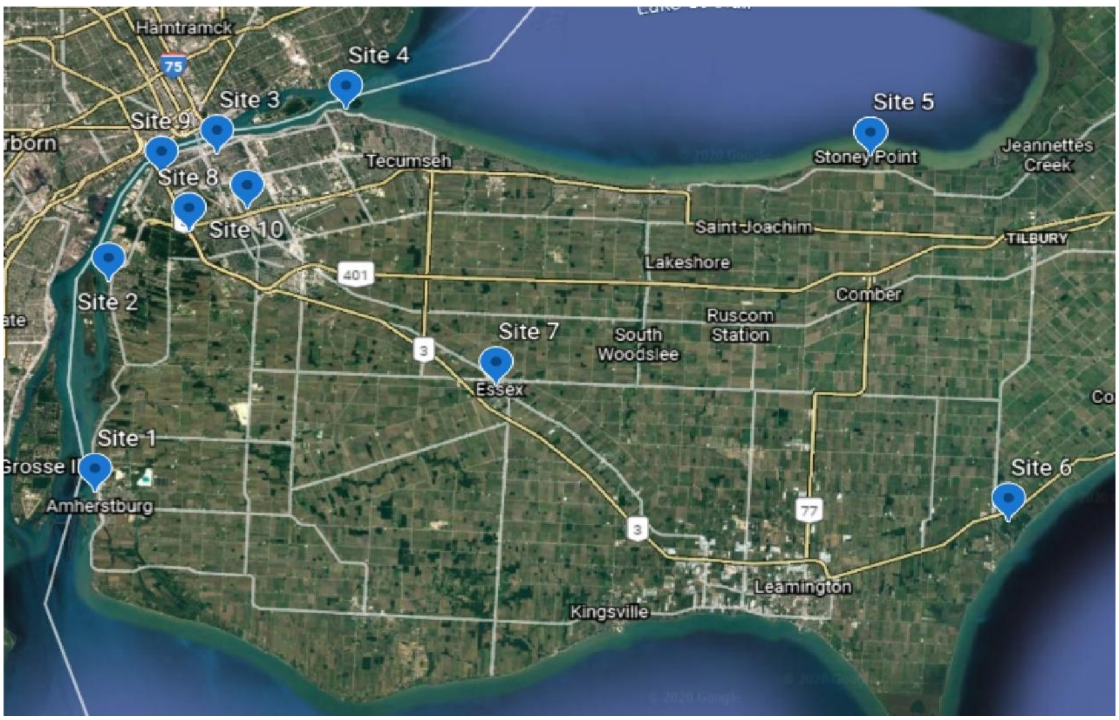


Figure 3.1: Map of dataset collection Sites across Essex County, Ontario, Canada [31].

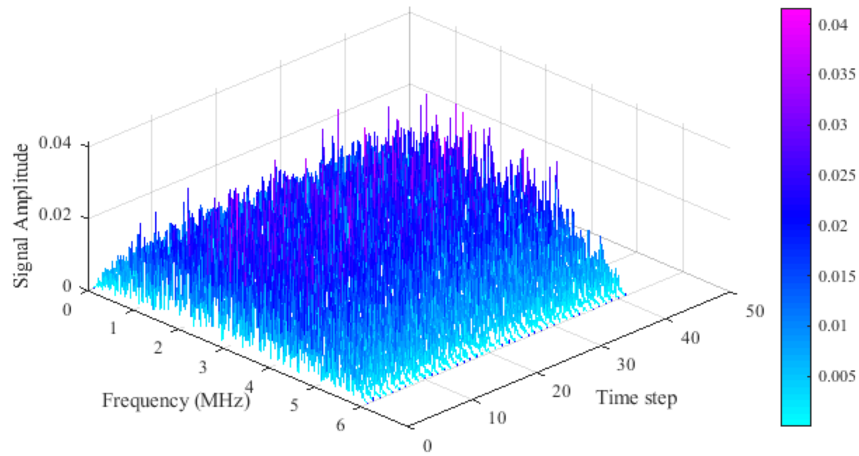
3.2.2 ATSC Channels Classes

The bandwidth of each channel at the collection site was manually examined to create labels for the datasets using the USRP and GQRX software as a spectrum analyzer. Five classes are identified and presented in Table 3.1 [31].

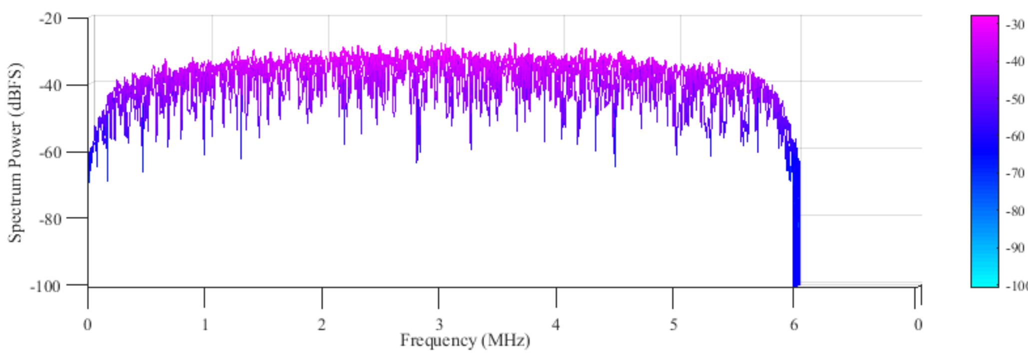
Table 3.1: ATSC channels classes

ATSC channels classes	Class (C)
Unoccupied (PUs and SUs are inactive)	0
ATSC Present, strong (PU active)	1
ATSC Present, weak (PU active)	2
Strong interference (SU active)	3
Weak signal interference (SU active)	4

1. Unoccupied channel (Class 0): it corresponds to a case where neither PU nor SU is found to exist in the channel. The channel in Fig. 3.2 is available for use by CR.
2. ATSC present, strong (Class 1): it corresponds to the presence of a PU (i.e. a broadcast television station). The signal spectrum appears relatively uniform and flat across the bandwidth with a pilot tone located at the left edge of the channel as shown in Fig. 3.3. The channel is considered unavailable for use and the CR must not transmit in this band in this condition.
3. ATSC present, weak (Class 2): This class corresponds to situations where an ATSC signal is found to be very faintly visible over the bandwidth. The channel in Fig. 3.4 exhibits a pilot tone with a weaker amplitude.
4. Interference, strong (Class 3): This class corresponds to the presence of other SUs or unknown signals whose amplitudes are sufficiently large over the channel bandwidth as exhibited in Fig. 3.5.
5. Interference, weak (Class 4): the graph in Fig. 3.6 represents weak interference class corresponds to the presence of other SUs or unknown signals whose amplitudes are small over the channel bandwidth.

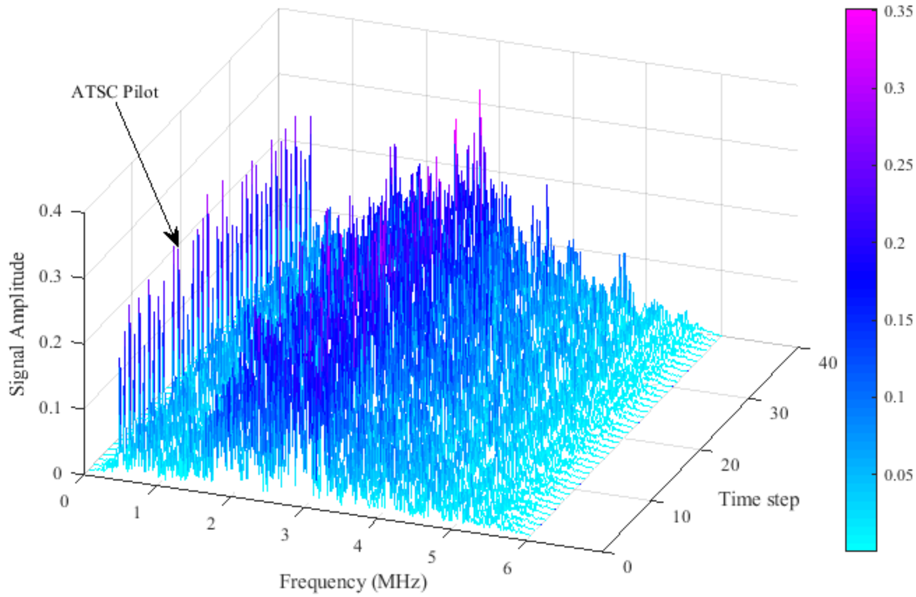


(a)

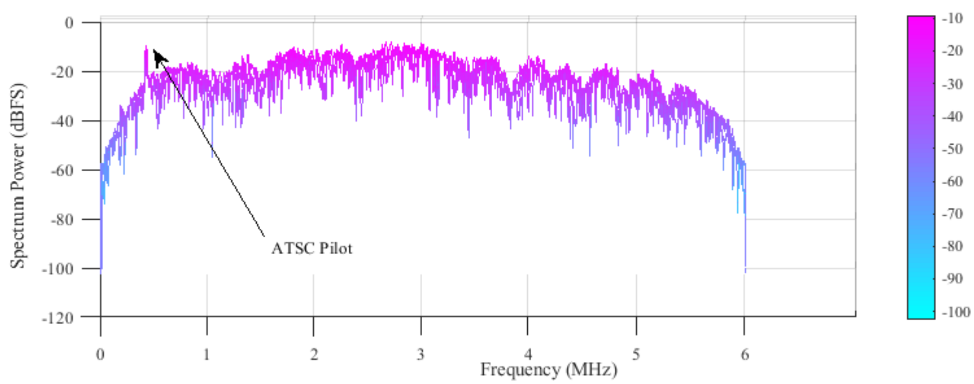


(b)

Figure 3.2: Unoccupied channel 51 at site 1 (a) FFT samples and (b) waterfall display of spectrum power in (dBFS).

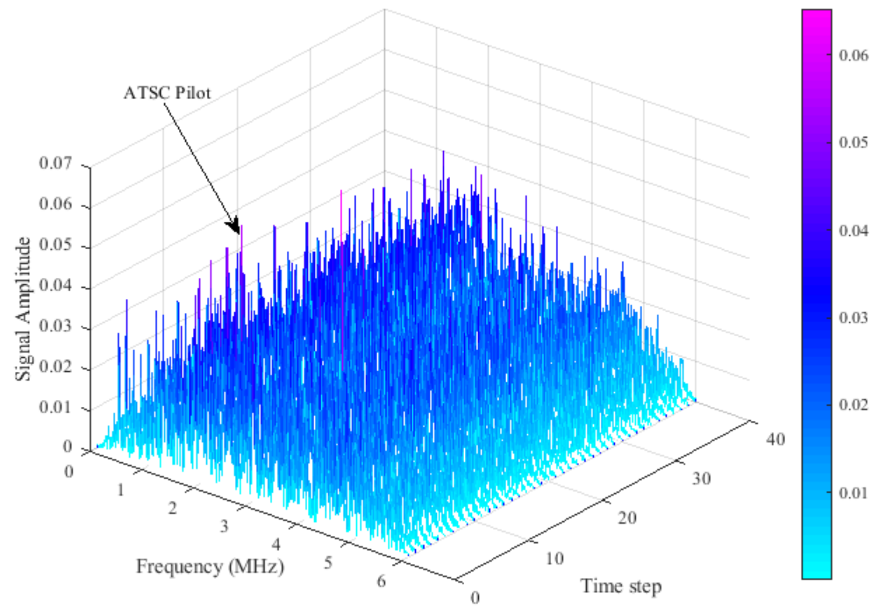


(a)

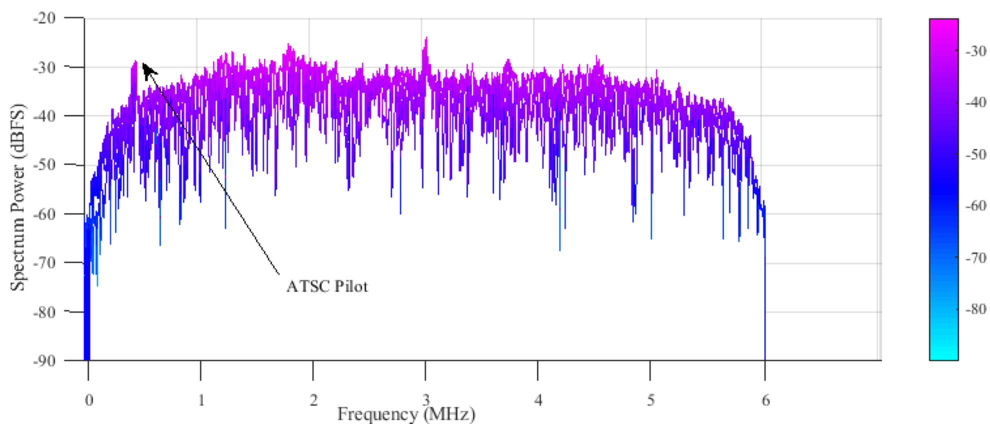


(b)

Figure 3.3: ATSC signal present, strong in channel 41 at site 1 (a) FFT samples, and (b) waterfall display of spectrum power in (dBFS).

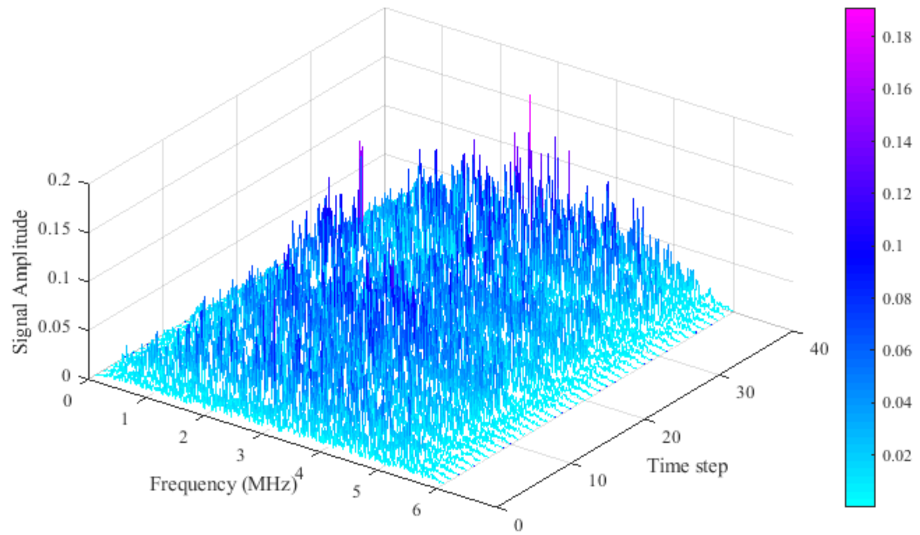


(a)

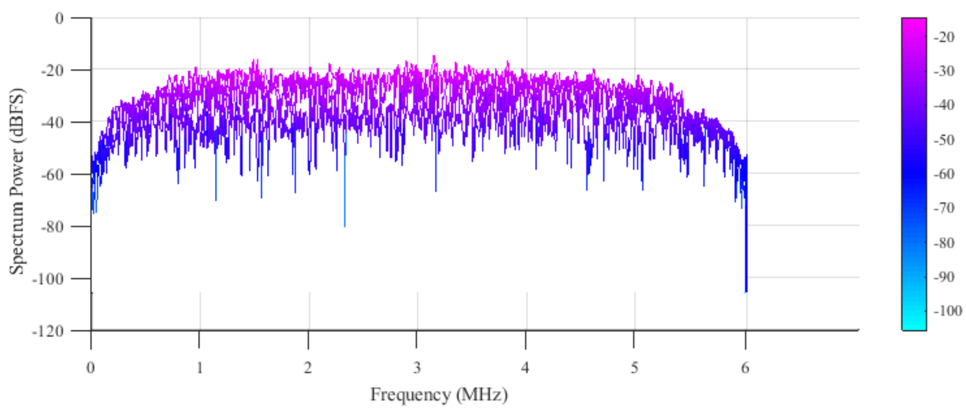


(b)

Figure 3.4: ATSC signal present, weak in channel 21 at site 1 (a) FFT samples, and (b) waterfall display of spectrum power in (dBFS).

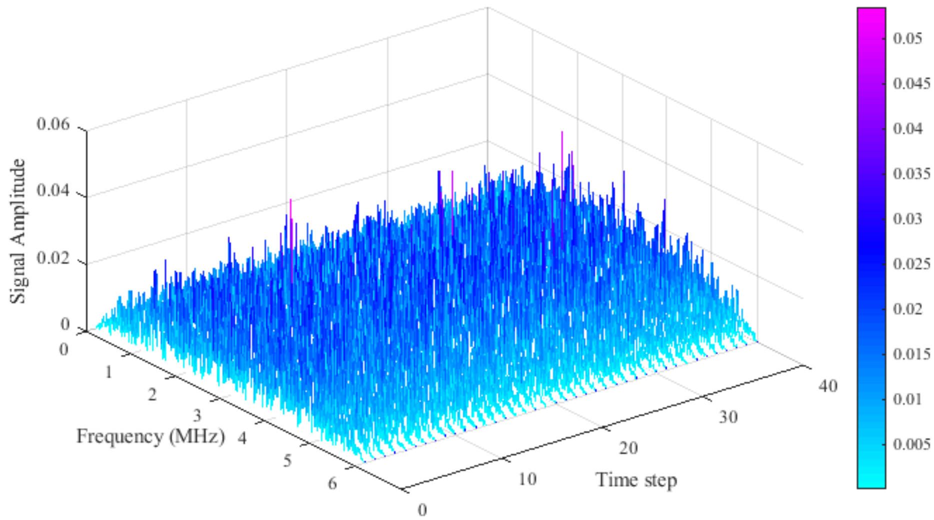


(a)

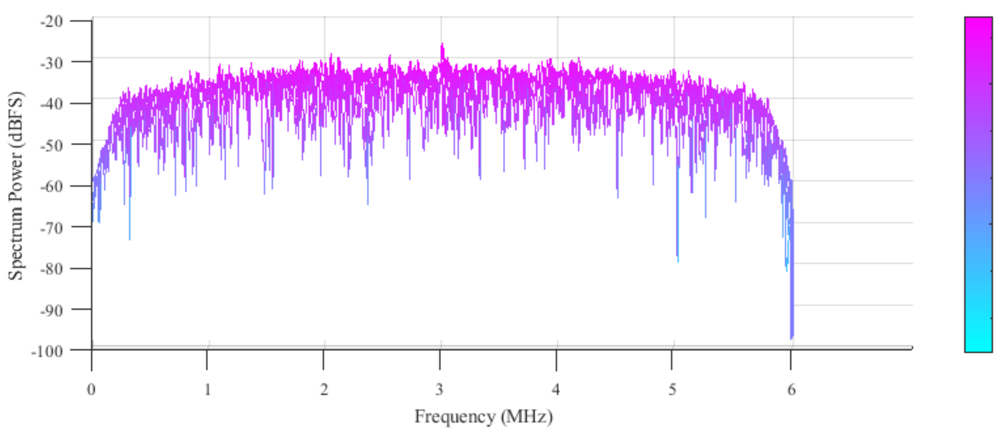


(b)

Figure 3.5: Interference, strong in channel 44 at site 1 (a) FFT samples, and (b) waterfall display of spectrum power in (dBFS).



(a)



(b)

Figure 3.6: Interference, weak in channel 35 at site 1 (a) FFT samples, and (b) waterfall display of spectrum power in (dBFS).

3.2.3 Dataset Structure

The datasets are composed of files in hierarchical data format (HDF5) that are designed to store and organize large and multidimensional datasets. Each file corresponds to a single scan of the 250 MHz bandwidth of interest at the sites. It split hierarchically among five-channel groups (i.e. classes) as shown in Fig. 3.7 [31].

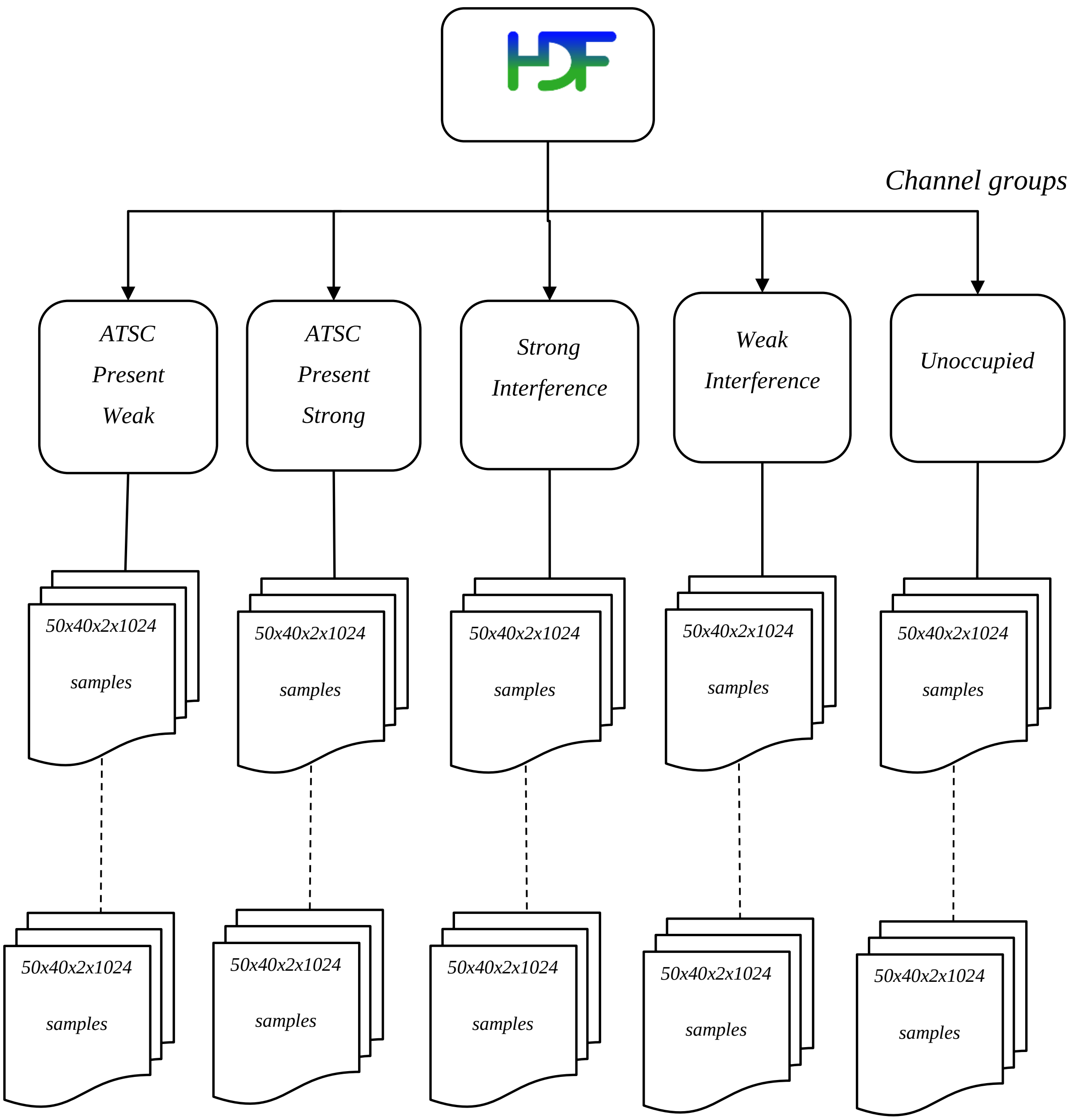


Figure 3.7: Dataset Structure [31].

The channel groups are then further divided into multidimensional arrays containing sequences of complex FFT points corresponding to the frequency spectrum of a single 6 MHz ATSC channel at given timesteps. Each row in the array contains 1024 complex FFT points corresponding to snapshots of the spectrum at uniformly- spaced time steps.

3.3 System Model and Assumptions

The ATSC licensed users in the frequency band from 490 to 740 MHz are considered as PUs. The system consists of M channels occupied by PUs and unknown signals. The CR will be referred to as SU which will share channels with the PU if the PU or interference (other SUs) are not present. The channel m is available only for the CR network to exploit when there is no PU or SU in the active state in that channel. If A denotes channel availability:

$$A = \begin{cases} +1 & \text{if } C_m = 0 \text{ for all } m, \\ -1 & \text{otherwise} \end{cases} \tag{3.1}$$

where C_m denotes the class of channel m and M is the number of channels.

The main goal of the proposed ML models is to correctly assess the availability of channels. Since the suggested algorithms are supervised learning, FFT samples for each channel will be used with their labels to train the classifiers. This is equivalent to designing a classifier to correctly map the FFT samples to the class of the channels in the framework of ML. The ML model will be able to detect the presence of PU by observing the pilot signal.

Let Y represents the set of energy samples for training dataset after normalization (i.e. the input):

$$Y = \begin{pmatrix} y_{11} & \cdots & y_{1N} \\ \vdots & \ddots & \vdots \\ y_{T1} & \cdots & y_{TN} \end{pmatrix}, \tag{3.2}$$

where N and T represent the number of FFT samples for each channel and the scanning duration for the channels, respectively. Let C represent the class of the channels (i.e. the output) corresponding to Y:

$$C = \begin{pmatrix} c_1 \\ \vdots \\ c_T \end{pmatrix} \tag{3.3}$$

Let Z is the output of the PCA that represent the transformation of Y into a lower dimension:

$$Z = \begin{pmatrix} z_{11} & \cdots & z_{1n} \\ \vdots & \ddots & \vdots \\ z_{T1} & \cdots & z_{Tn} \end{pmatrix}, \tag{3.4}$$

where $n \ll N$.

In Fig. 3.8, the framework of the proposed ML models is demonstrated, comprising of the data preprocessing, dimensionality reduction, classification training, and testing. The dotted blocks illustrate using PCA as a dimensionality reduction technique before the classification.

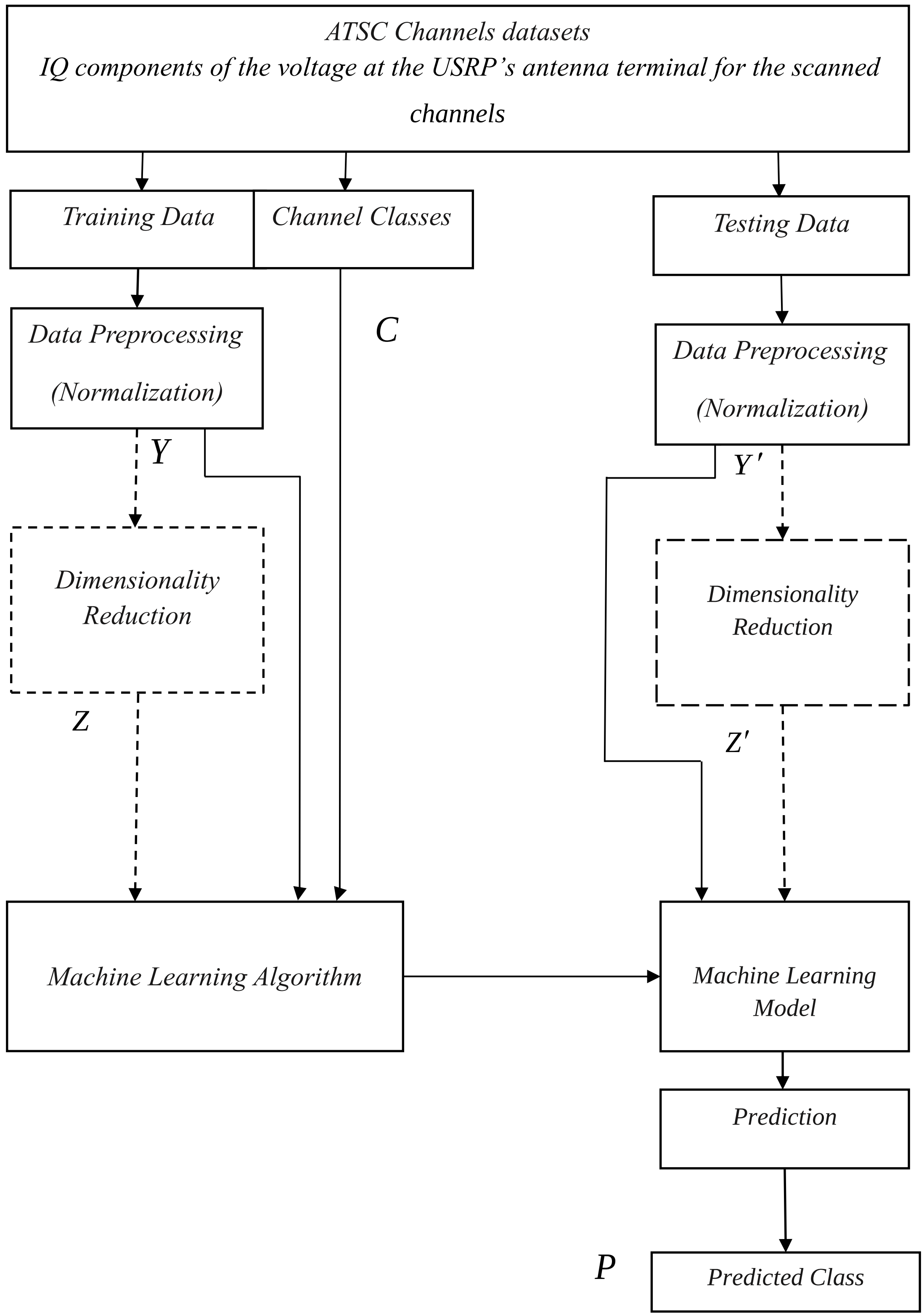


Figure 3.8: ML-based spectrum sensing framework.

Once the classifier has been effectively trained, the samples in the testing data are ready for evaluation of the performance of the classifiers. Let Y' and Z' denote the FFT samples for testing and their new representations after applying dimensionality reduction, respectively. If C' donates the corresponding channel classes for the testing samples and P is the predicted class by the classifier. Hence,

$$P = \begin{cases} C' & \text{detection} \\ \text{otherwise} & \text{misdetection} \end{cases} \tag{3.5}$$

For instance, If the testing samples are for channel occupied by active PU and the model classifies the channels as unoccupied, it is misdetection.

3.4 Performance Measurement

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class as illustrated in Fig. 3.9. The confusion matrix shows how the classification model can distinguish between samples belonging to different classes when it makes predictions [32].

True Class

<table>
<tr><th></th><th>Class 0</th><th>Class 1</th><th>Class 2</th><th>Class 3</th><th>Class 4</th></tr>
<tr><td>Class 0</td><td colspan="2" rowspan="2">True Negative (TN)</td><td rowspan="2">False Positive (FP)</td><td colspan="2" rowspan="2">True Negative (TN)</td></tr>
<tr><td>Class 1</td></tr>
<tr><td>Class 2</td><td colspan="2">False Negative (FN)</td><td>True Positive (TP)</td><td colspan="2">False Negative (FN)</td></tr>
<tr><td>Class 3</td><td colspan="2" rowspan="2">True Negative (TN)</td><td rowspan="2">False Positive (FP)</td><td colspan="2" rowspan="2">True Negative (TN)</td></tr>
<tr><td>Class 4</td></tr>
</table>

Predicted Class

Figure 3.9: Confusion matrix for multiclass classification [33].

In Fig. 3.9, TP is the outcome where the model correctly predicts samples belong to class 2 while TN is an outcome where the model correctly predicts samples do not belong to class 2. FP is an outcome where the model incorrectly predicts samples of other classes as samples of class 2. FN is an outcome where the model incorrectly predicts samples of class 2 as they belong to other classes.

- Accuracy is the measure of all the correctly identified samples. It is mostly used when all the classes are equally important and is calculated by the formula:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{3.6}$$

- The recall is the percentage of actual positives that are correctly identified. It is also called the true positive rate (TPR), the sensitivity, or the probability of detection and is calculated by the formula:

$$\text{TPR} = \frac{\text{TP}}{\text{P}} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.7}$$

- Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives. It shows the ability of a classification model to return only actual samples belong to the class.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.8}$$

- The false-positive rate (FPR**)** is the proportion of negative samples incorrectly identified as positive samples in the testing data. It is also called the probability of false alarm.

$$\text{FPR} = \frac{\text{FP}}{\text{N}} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{3.9}$$

- F-measure (Also called F1 score) is the harmonic mean of precision and recall and gives a better measure of the incorrectly classified cases than the accuracy metric. It is calculated as:

$$\text{F-measure} = \frac{2\ (\text{Precision} + \text{Recall})}{\text{Precision} \times \text{Recall}} \tag{3.10}$$

$$= \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}\,.$$

- ROC curve is a graphical tool that illustrates the behavior of TPR with respect to the FRP which reflects the performance of a classification model at different classification thresholds.

It is also notable that K-fold cross-validation is employed in performance evaluation since it usually leads to a less biased or less optimistic estimation of the model than other techniques. In this approach, the dataset is randomly divided into K groups or folds, then the model is trained using (K – 1) folds and the models are tested using the remaining Kth fold. the accuracy and F-measure for each fold were recorded as shown in Table 4.1. The process is repeated until every K-fold serves as the test set. The value of K is set to 5. Two approaches for spectrum sensing based on ML models are adopted as follows:

Algorithm 3.1: Spectrum sensing Algorithm based on ML classification techniques

---

1. Let Y in (Eqn. 3.2) be the normalized values of the FFT samples collected by the USRP and C in (Eqn.3.3) are the classes labeled for the scanned channels.
2. Choose a classifier and initialize the parameters.
3. Train the classifier model using Y and C in step 1.
4. Cross validate and test the trained model using Y .

5. Predict the class of the testing data C using the trained model.
6. Compute F-measure and accuracy.

Algorithm 3.2: Spectrum sensing Algorithm based on dimensionality reduction and ML classification techniques

---

1. Let Y in (Eqn. 3.2) be the normalized values of the FFT samples collected by the USRP and C in (Eqn. 3.3) are the classes labeled for the scanned channels.
2. Select a suitable value n for the number of principal components for PCA where $n \ll N$.
3. Compute Z in (Eqn. 3.4) by finding the principal components of Y in (Eqn. 3.2) using PCA.
4. Choose a classifier and initialize the parameters.
5. Train the classifier model using C and Z in steps 1 and 3, respectively.
6. Cross validate and test the trained model using Y.
7. Predict the class of the testing data C using the trained model.
8. Compute F-measure and accuracy.

3.5 Summary

The available are employed for training and testing purposes. Datasets consist of FFT samples that cover the TV channels band from 490 to 740 MHz. ML-based spectrum sensing framework has proposed. It includes data preprocessing, feature extraction, and it ends with predicting the class of testing data. Moreover, Accuracy, F-measure, and ROC curve will be used to evaluate the performance of the four classifiers.

CHAPTER 4

Results and Discussions

Based on real spectrum data collected over different locations in Essex County [31], the results of simulating the proposed spectrum sensing-based ML models using different classifiers are evaluated employing different performance metrics. The performance metrics are confusion matrix, accuracy, F-measure, and ROC provided in Chapter 3. Also, the collected datasets are examined and it is concluded that the results follow the same pattern without loss of generality. Moreover, the impact of employing PCA on the detection performance of the proposed model is discussed in this chapter.

4.1 Simulation results of the classifiers

In this section, the proposed Algorithm 3.1 in Chapter 3 is examined using the four classifiers, and performances are investigated. The parameters of four classifiers, i.e., type of kernel for SVM, the number of neighbors (k), and the type of distance measure for the kNN, are inspected.

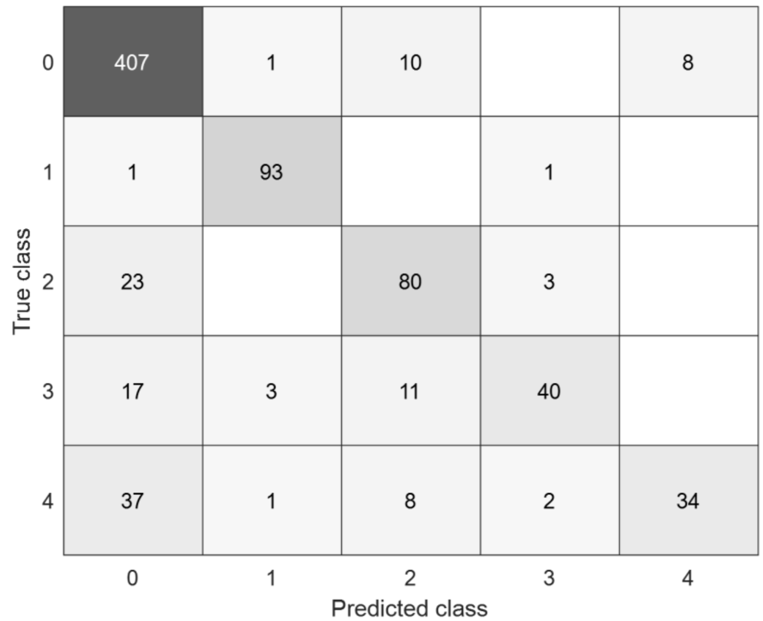


Figure 4.1: Confusion matrix for SVM classifier without PCA at site 1.

The accuracy and F-measure of fold 2 in Table 4.1 are obtained from the confusion matrix shown in Fig. 4.1 using Eqn. 3.6 and 3.10, respectively. It presents the TP, TN, FP, and FN of values of the five classes.

Table 4.1: SVM classifier performance without PCA at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9333	0.8333
fold2	0.9354	0.8385
fold3	0.9297	0.8244
fold4	0.9262	0.8154
fold5	0.9308	0.8269

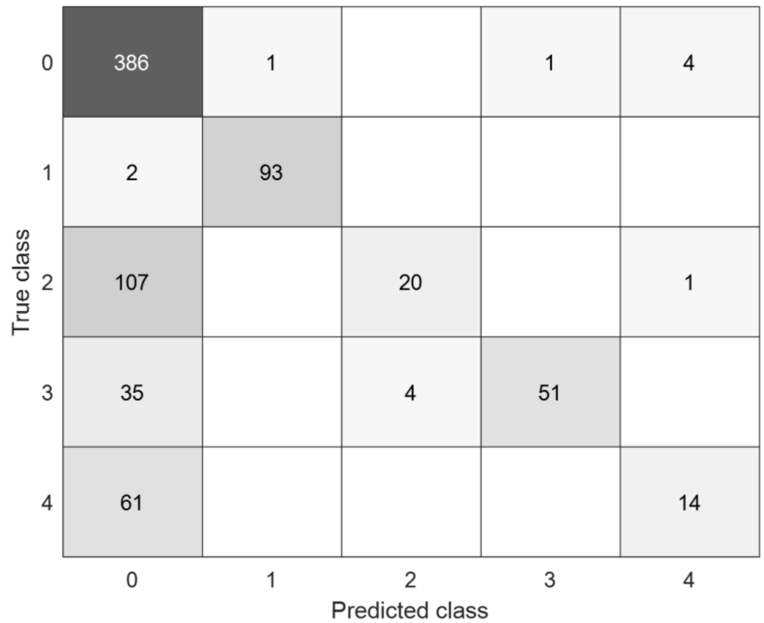


Figure 4.2: Confusion matrix for the kNN classifier without PCA at site 1.

Similarly, the confusion matrix in Fig. 4.2 illustrates TP, FP, FN, and TN of the kNN classifier where the accuracy and F-measure of fold 2 presented in table 4.2 are obtained. It is obvious from Tables 4.1 and 4.2 that the accuracy and F-measure for the SVM model are greater than their counterparts in the kNN model. Therefore, the SVM exceeds kNN due to the fact when F-measure increases, the recall (i.e. probability of detection) increases as shown in Eqn. 3.10. Additionally, the ML model performs well when it achieves the maximum number of TP and TN and the minimum numbers of FP and FN.

Table 4.2: kNN classifier performance without PCA at site 1.

5 K fold	Accuracy	F-measure
fold1	0.8887	0.7218
fold2	0.8892	0.7231
fold3	0.8846	0.7115
fold4	0.8826	0.7064
fold5	0.8846	0.7115

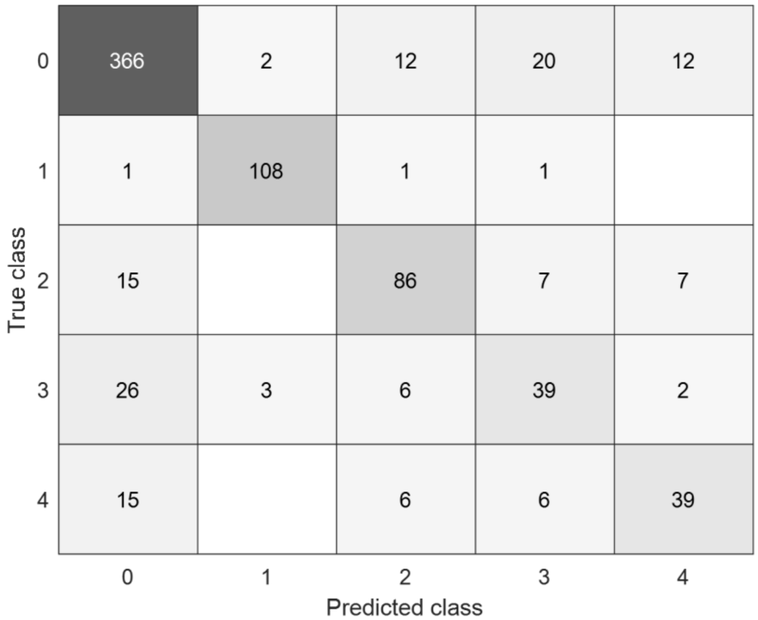


Figure 4.3: Confusion matrix for DT classifier without PCA at site 1.

In the same manner, Fig. 4.3 presents the confusion matrix for the DT classifier among the five classes where the accuracy and F-measure of fold 2 shown in Table 4.3 are obtained.

Table 4.3: DT classifier performance without PCA at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9144	0.7859
fold2	0.9272	0.8179
fold3	0.9164	0.7910
fold4	0.9108	0.7769
fold5	0.9149	0.7872

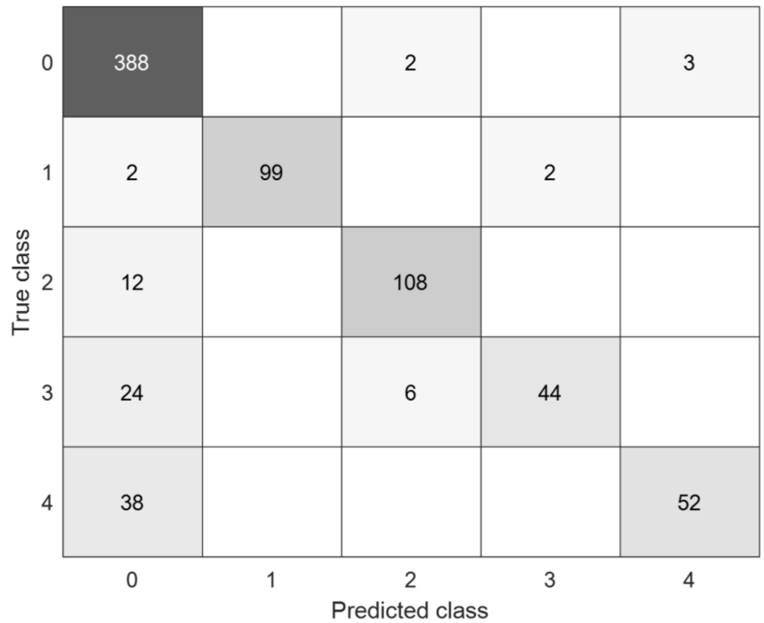


Figure 4.4: Confusion matrix for Ensemble classifier without PCA at site 1.

Figure 4.4 displays the confusion matrix of the Ensemble ML model using data collected from site 1. It is evident from Table 4.4 that the Ensemble classifier outperforms the SVM, kNN, and DT classifiers in terms of accuracy and F-measure as it achieves the maximum accuracy and F-measure, around 94 % and 86 %, respectively. Hence, it improves the probability of detection.

Table 4.4: Ensemble classifier performance without PCA at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9446	0.8615
fold2	0.9436	0.8590
fold3	0.9374	0.8436
fold4	0.9441	0.8603
fold5	0.9533	0.8833

In order to develop the ROC curves for the different classifiers. Algorithm 3.1 is implemented in Matlab and is executed for several attempts using different parameters and the two best results are shown in Figs. 4.5, 4.6, and 4.7.

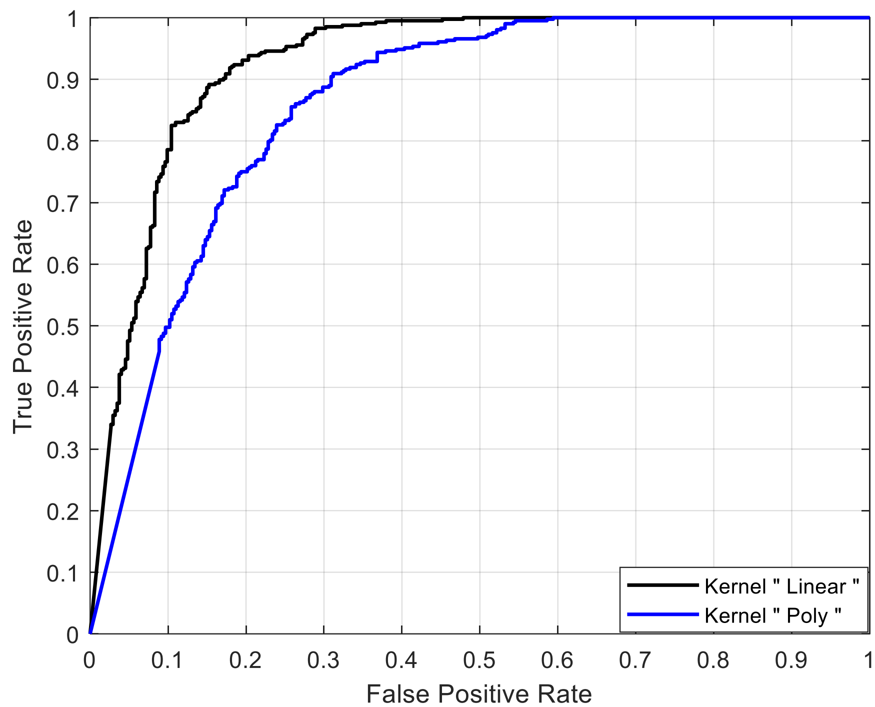


Figure 4.5: ROC curves for SVM classifier using linear and polynomial kernels at site 1.

Figure 4.5 illustrates the ROC curves for the SVM classifier for class 0 (i.e. unoccupied channel). Linear and polynomial (poly) kernels for the SVM classifier are investigated to assess the detection performance. The larger the area under the ROC curve, the better the SVM model is at distinguishing between samples of the unoccupied channels and the samples of channels occupied by PU or SU. Furthermore, at FPR (probability of false alarm) = 0.2, the TPR (probability of detection) for linear and poly kernels are 0.93 and 0.75, respectively. Therefore, the linear kernel excels poly kernel in terms of TPR and FPR as shown in Fig. 4.5.

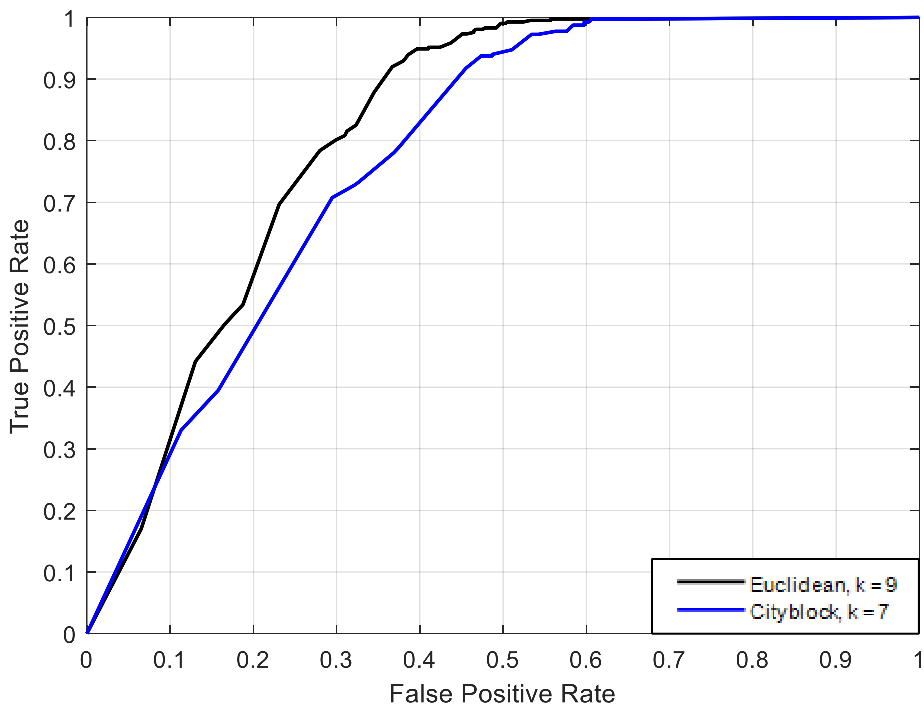


Figure 4.6: ROC curves for the kNN classifier among two distance metric and neighbor numbers at site 1.

Similarly, two distance metrics have been inspected to assess the kNN classifier performance. It can be observed from Fig. 4.6 that using euclidean as distance metric enhances the performance of kNN classification in terms of ROC in comparison to City block. In Fig. 4.7, the performance of the DT algorithm is examined based on the ROC curve using two leaf sizes (LS). At TPR (probability of detection) = 0.9, the for FPR (probability of false alarm) for DT with LF = 10 and 7 are 0.14 and 0.4, respectively. Thus, DT with LS =10 has a lower probability of false alarm.

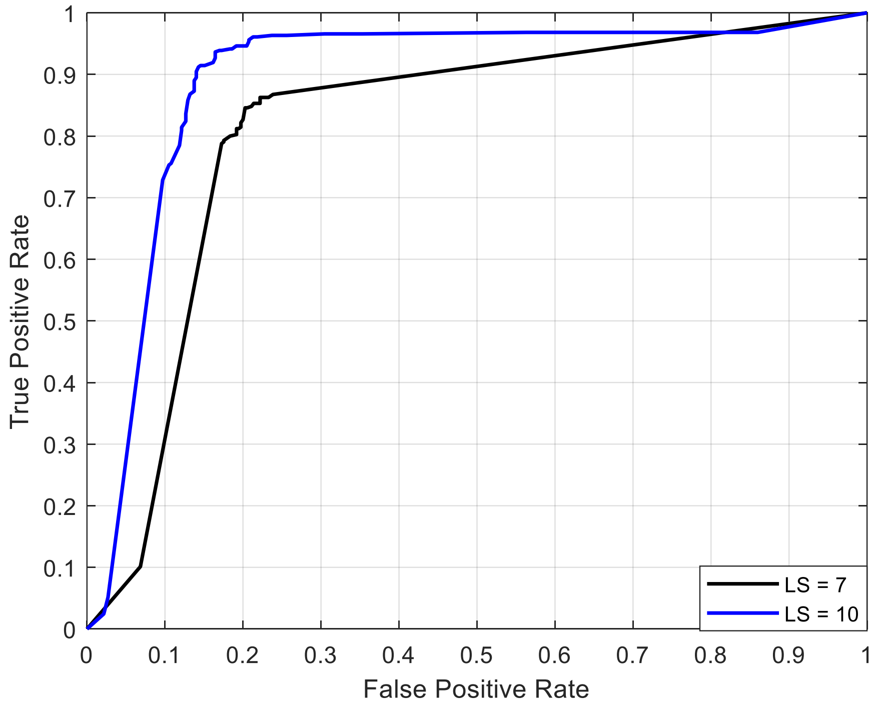


Figure 4.7: Comparison of ROC curve for DT classifier using various LS at site 1.

Figure 4.8 exhibits the ROC curves of the four classifiers without employing PCA. By inspection, the Ensemble has the highest probability of detection (TPR) for a given false alarm rate (FPR). For this reason, the Ensemble outperforms the other techniques by combining multiple classifiers while SVM comes in the second rank by finding the hyperplane maximizing the margin between the classes.

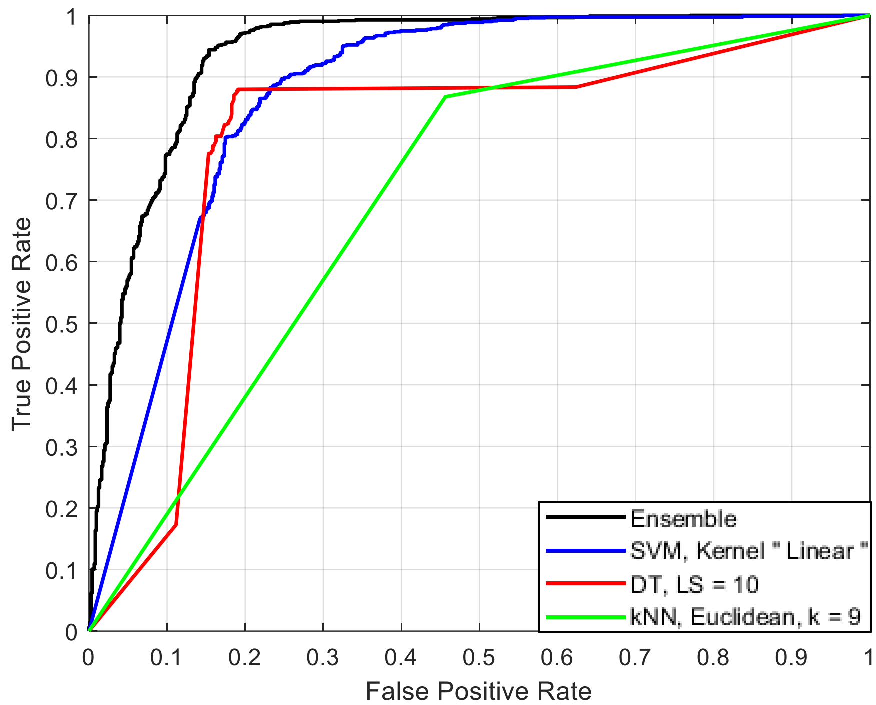


Figure 4.8: Comparison of ROC curves for all used classifiers without PCA at site 1.

4.2 Simulation results of the classifiers using the PCA technique

The performances of the four classification methods are evaluated after employing PCA using Algorithm 3.2 and compared with the results in section 4.1.

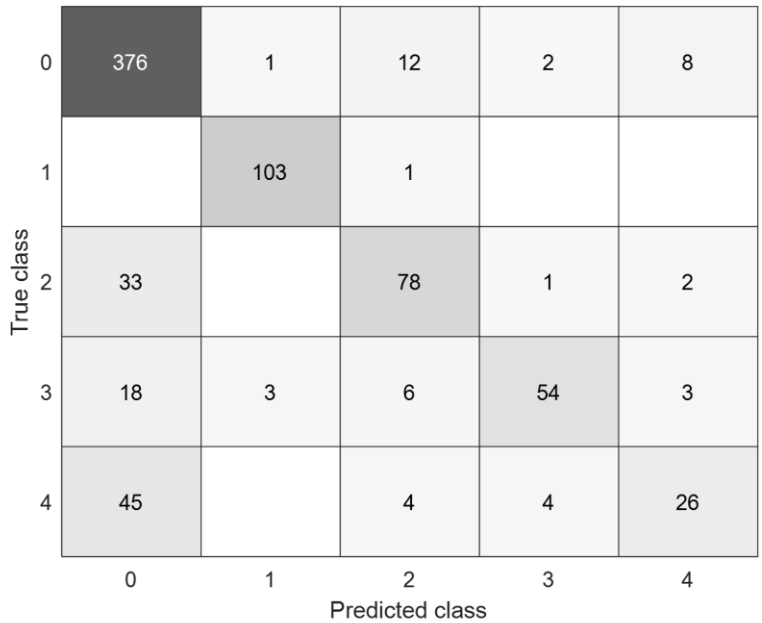


Figure 4.9: Confusion matrix for the SVM classifier using the PCA technique at site 1.

Figure 4.9 presents the confusion matrix of the SVM algorithm utilizing PCA where the accuracy and F-measure of fold 2 in table 4.5 are obtained. It can be seen from Tables 4.1 and 4.5, respectively, that the accuracy and F-measure are slightly dropped from 0.93 and 0.83 to 0.92 and 0.8, respectively as a result of losing information in the dataset after implementing PCA.

Table 4.5: SVM Classifier performance using the PCA technique at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9215	0.8038
fold2	0.9267	0.8167
fold3	0.9205	0.8013
fold4	0.9118	0.7795
fold5	0.9108	0.7769

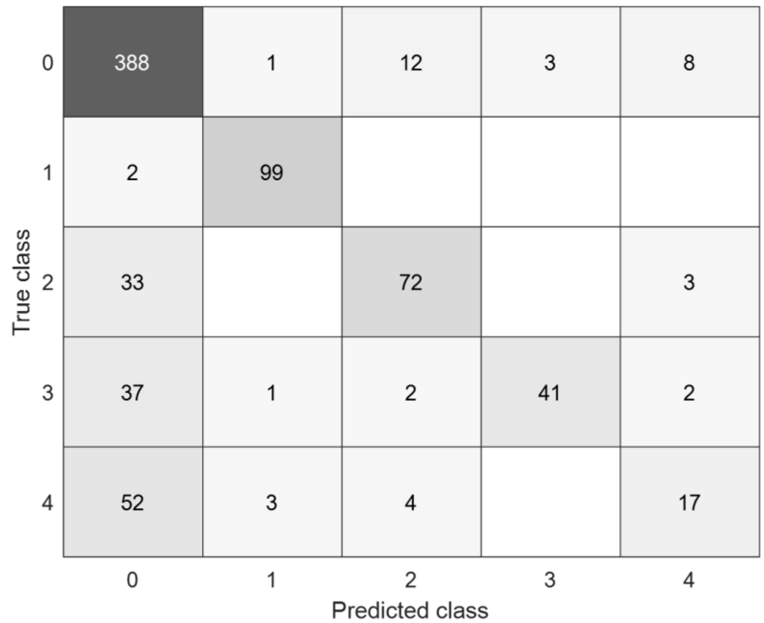


Figure 4.10: Confusion matrix for the kNN classifier using the PCA technique at site 1.

Following the same procedure, the accuracy and F-measure of fold 2 in table 4.6 are calculated using the kNN confusion matrix in Fig 4.10. It can be perceived from tables 4.2 and 4.6 that the accuracy and F-measure have increased from 0.88 and 0.72 to 0.92 and 0.8, respectively due to fact that the kNN classifier operates better with a small number of features (i.e. lower dimension) than a large number of features (i.e. higher dimension).

Table 4.6: kNN classifier performance using the PCA technique at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9262	0.8154
fold2	0.9164	0.7910
fold3	0.9241	0.8103
fold4	0.9103	0.7756
fold5	0.9195	0.7987

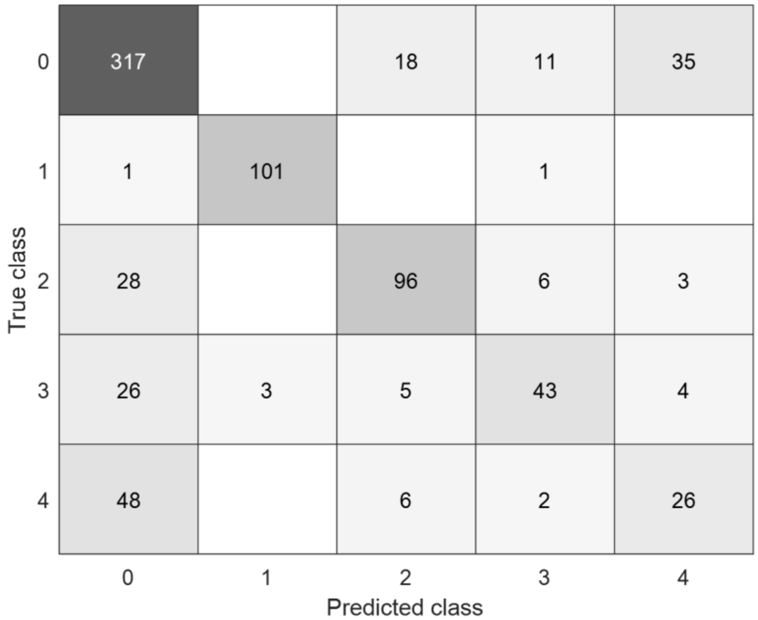


Figure 4.11: Confusion matrix for DT Classifier using the PCA technique at site 1.

Figure 4.11 displays the confusion matrix of the SVM algorithm using PCA where the accuracy and F-measure of fold 2 in table 4.7 are obtained. It can be shown from Tables 4.3 and 4.7 that the averages of the accuracy and F-measure nearly equal. Thus, PCA has reduced the number of features in the datasets while retaining the classification performance.

Table 4.7: DT classifier performance using the PCA at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9128	0.7821
fold2	0.8990	0.7474
fold3	0.8990	0.7474
fold4	0.9103	0.7756
fold5	0.9010	0.7526

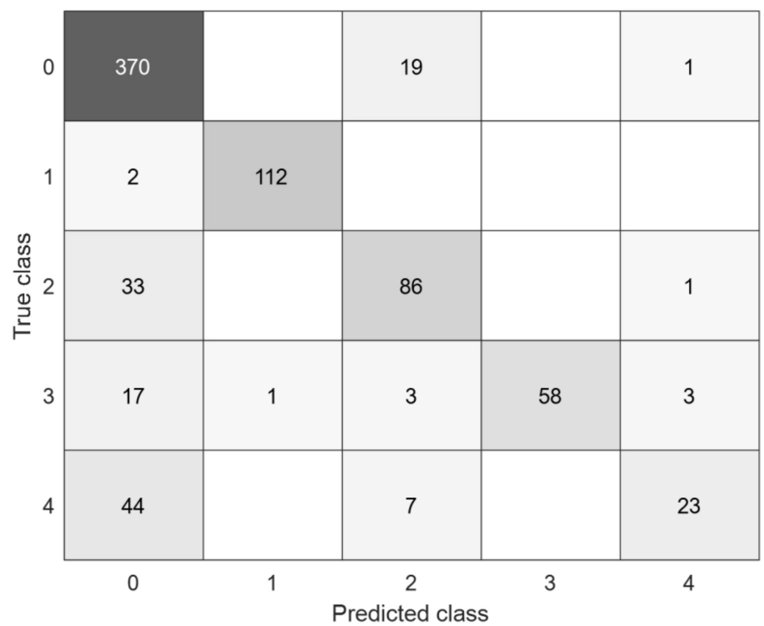


Figure 4.12: Confusion matrix for Ensemble classifier using the PCA technique at site 1.

Furthermore, the accuracy and F-measure of fold 2 in Table 4.8 are calculated using the Ensemble confusion matrix in Fig 4.12. It can be noted from Tables 4.4 and 4.8 that the accuracy and F-measure have slightly decreased from 0.94 and 0.86 to 0.92 and 0.83, respectively, as a result of losing information in the dataset after applying PCA.

Table 4.8: Ensemble classifier performance using the PCA technique at site 1.

5 K fold	Accuracy	F-measure
fold1	0.9282	0.8205
fold2	0.9328	0.8321
fold3	0.9256	0.8141
fold4	0.9241	0.8103
fold5	0.9354	0.8385

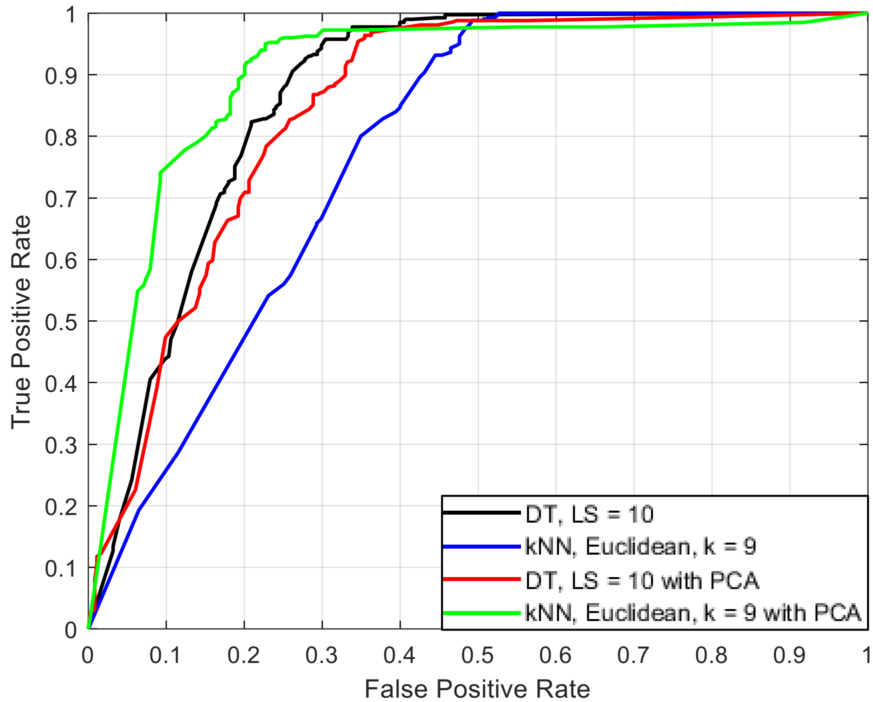


Figure 4.13: Comparison of ROC curves obtained by kNN and DT classifiers with and without using the PCA technique at site 1.

In Fig.4.13, ROC curves of kNN and DT classifiers with and without using the PCA technique are exhibited. It can be observed for a given FPR, the TPR of the kNN classifier with PCA is higher than kNN without PCA. Consequently, the performance is enhanced after implementing PCA, since the kNN classifier works efficiently on datasets of lower features. Besides that, Applying the PCA technique maintains the performance of the DT classifier by extracting features relevant to the classification problem. Likewise, ROC curves of Ensemble and SVM classifiers with and without using the PCA technique are depicted in Fig. 4.14. The classification performance based on ROC after utilizing the PCA technique is insignificantly declined as for Ensemble and SVM classifiers as a result of information loss during reducing the dimension.

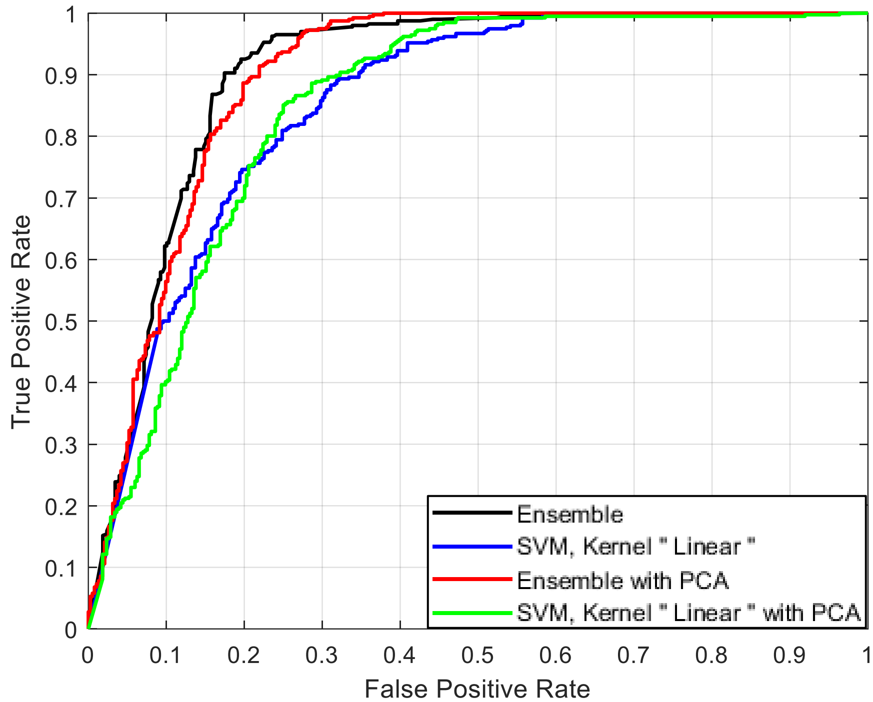


Figure 4.14: Comparison of ROC curves obtained by Ensemble and SVM classifiers with and without using the PCA technique at site 1.

Furthermore, the duration of the training for the proposed classification algorithms is computed and compared with/without the PCA technique as reported in Table 4.9. It is clear that the DT and kNN classification algorithms have the lowest training duration. It is also shown the Ensemble algorithm has the greatest training duration seconded by the SVM algorithm due to the intensive computational complexity of both algorithms compared with the kNN and DT algorithms.

Table 4.9: Comparison of training duration (in seconds) for the classifiers with and without using the PCA technique for 2000 training samples at site 1.

Classification Methods	Without PCA	With PCA
SVM	4.0480	2.3505
kNN	0.8315	0.6561
DT	2.8848	0.7673
Ensemble	27.2410	4.8514

4.3 Summary

The simulation results of the four classifiers have been presented and demonstrated in terms of accuracy, F-measure, and ROC. The results have shown that the Ensemble classifier exceeds the other classifiers based on performance metrics followed by the SVM classifier. Nevertheless, the performance comes at the cost of greater computational complexity. Plus, the SVM classifier attempts to find the hyperplane that maximizes the margin between the classes which results in increasing the training duration compared with the kNN and the DT classifiers. Furthermore, the results have shown that employing a dimensionality reduction method such as the PCA technique before the classifier is critical to speed up the training process across the four classifiers by extracting the most essential features and removing redundancy in datasets. However, PCA should be carefully exercised to maintain classification performance.

CHAPTER 5

Thesis Conclusions and Future works

5.1 Conclusions

Supervised ML algorithms have been investigated as an alternative route for sensing the spectrum to overcome limitations in the conventional spectrum sensing techniques. ML models based on the SVM, kNN, DT, and Ensemble classifiers are examined to detect the presence of PU and unknown users over the TV bands. The proposed models have been trained and validated using the available datasets collected at ten different sites across Essex County, Ontario. Unlike the single classifier approach, an Ensemble classification approach combines the outputs of multiple classifiers. Consequently, the overall classification performance of the Ensemble classifier excels single classification approach. Moreover, the datasets have been projected to a lower-dimensional space by applying the unsupervised learning PCA technique to remove redundancy in the data as well as speed up the learning process for the four classifiers and more importantly help in avoiding overfitting.

The results have revealed that the Ensemble classifier surpasses the other classifiers based on the performance measurements via joining multiple classifiers. However, the performance comes at the cost of greater computational complexity. Moreover, the SVM classifier exceeds kNN and, DT classifiers in terms of accuracy, and F-measure. Furthermore, the results have shown that applying a dimensionality reduction method such as PCA before the classifier decreases the training duration by extracting features and removing redundancy in datasets. Though, the PCA technique should be carefully used to maintain classification performance.

5.2 Future works

Recently, the application of ML algorithms in cognitive radios has evolved as an attractive and versatile research subject of interest to many researchers. For future works the following research areas are recommended:

1. A multistage classification-based spectrum sensing approach may be further explored to boost accuracy and F-measure.
2. The noise uncertainty and the radio conditions such as shadowing and multipath propagation fading has a detrimental effect on the sensing performance and result in a hidden terminal problem (HTP). Hence, the research can be extended to implement ML models for CSS to enhance the performance of the detection.
3. During the training phase of the classifiers, the class of the channels is not always accessible. hence, supervised learning methods may not be applicable. Therefore, unsupervised ML algorithms may be a convenient method for spectrum sensing.

REFERENCES

[1] T. Yucek and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications,” IEEE Communication Survey Tutorials, vol. 11, no. 1, pp. 116–130, 2009.

[2] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, “Cooperative spectrum sensing in cognitive radio networks: A survey,” Physical Communication, vol. 4, no. 1, pp.40-62, 2011.

[3] C. Jiang, H. Zhang, and Y. Ren, “Machine Learning Paradigms for Next-Generation Wireless Networks,” IEEE Wireless Communications, vol 24, no. 2, pp. 98 - 105, 2017.

[4] M. Bkassiny, Y. Li, and S. K. Jayaweera, “A Survey on Machine-Learning Techniques in Cognitive Radios,” IEEE Communications Surveys and Tutorials, vol. 15, no. 3, pp. 1136 - 1159, 2013.

[5] J. Mitola and G. Q. Maguire, “Cognitive radio: making software radios more personal,” IEEE Personal Communications, vol. 6, no. 4, pp. 13 –18, 1999.

[6] S. Haykin, “Cognitive radio: brain-empowered wireless communications,” IEEE Journal on Selected Areas Communications., vol. 23, no. 2, pp. 201–220, 2005.

[7] T. Costlow, “Cognitive radios will adapt to users,” IEEE Intelligent Systems, vol. 18, no. 3, p. 7, 2003.

[8] S. Jayaweera and C. Christodoulou, "Radiobots: Architecture, Algorithms, and Realtime Reconfigurable Antenna Designs for Autonomous, Self-learning Future Cognitive Radios", UNM Digital Repository, 2020. [Online]. Available: http://repository.unm.edu/handle/1928/12306. [Accessed: 12- Jun- 2020].

[9] R. S. Michalski, “Learning and cognition,” in World Conference on the Fundamentals of Artificial Intelligence (WOCFAI ’95), pp. 507–510, 1995.

[10] C. Clancy, J. Hecker, E. Stuntebeck, and T. O’Shea, “Applications of machine learning to cognitive radio networks,” IEEE Wireless Communications, vol. 14, no. 4, pp. 47 – 52, 2007.

[11] R. Samra, “A Dynamic Multi-Cast Routing Algorithm for Opportunistic Networks: Implementing the Expected Transmission Count Metric”, arXiv preprint arXiv:1605.04971, 2016.

[12] F. Awin, E. Abdel-Raheem, “Blind Spectrum Sensing Approaches for Interweaved Cognitive Radio System: A Tutorial and Short Course”, IEEE Communications Surveys and Tutorials, vol 21, no. 1, 2019.

[13] B. Aneja, K. Sharma, and A. Rana, “Spectrum Sensing Techniques for a Cognitive Radio Network”. Lecture Notes in Electrical Engineering, vol 509, Springer, 2019.

[14] T. Baykas, J. Wang, and M. A. Rahman “Overview of TV White Spaces: Current regulations, standards, and coexistence between secondary users”, IEEE Personal, Indoor, and Mobile Radio Communications Workshops (PIMRC Workshops), pp.38-43, 2010.

[15] T. Baykas, M. Kasslin, and M. Cummings, “Developing a standard for TV white space coexistence: Technical challenges and solution approach”, IEEE Wireless Communications, no. 19, vol. 1, 2012.

[16] “DBS-01 - White Space Database Specifications,” Spectrum management and telecommunications, 24-Jan-2020. [Online]. Available: https://www.ic.gc.ca/eic/site/smt-gst.nsf/eng/sf10928.html. [Accessed: 12-Jun-2020].

[17] M. S. Richer, G. Reitmeier, T. Gurley, G. A. Jones, J. Whitaker, and R. Rast, "The ATSC Digital Television System", Proceedings of the IEEE, vol. 94, no. 1, 2006.

[18] "IEEE 802.22 Working Group on Wireless Regional Area Networks," IEEE 802.22 WRAN WG Website. [Online]. Available: http://www.ieee802.org/22/. [Accessed: 12-Jun-2020].

[19] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," IBM Journal of Research and Development, vol. 3, no. 3, pp. 210–229, 1959.

[20] S. Russell, and P. Norvig, "Artificial Intelligence: A Modern Approach", Third Edition, Prentice-Hall. ISBN 9780136042594, 2010.

[21] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, 2, pp.121–167,1998

[22] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," The American Statistician, vol. 46, no. 3, p. 175, 1992.

[23] T. Mitchell, Machine learning. New York: McGraw Hill, 2017.

[24] "Machine Learning Decision Tree Classification Algorithm - Javatpoint," www.javatpoint.com. [Online]. Available: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm. [Accessed: 12-Jun-2020].

[25] R. Polikar, "Ensemble-based systems in decision making," IEEE Circuits System Magazine, vol. 6, no. 3, pp. 21–45, 2006.

[26] L. Breiman, "Random forests, Machine Learning", vol. 45, no. 1, pp. 5–32, 2001.

[27] R. E. Schapire, "The strength of weak learnability," Machine Learning, vol. 5, no. 2, pp. 197–227, 1990.

[28] S. Mishra, “Unsupervised Learning and Data Clustering,” Medium, 21-May-2017. [Online]. Available: https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a. [Accessed: 12-Jun-2020].

[29] L Van Der Maaten, E Postma, and J Van Den Herik , “Dimensionality Reduction: A Comparative Review”, Tilburg University Technical Report, TiCC-TR 2009-005, 2009.

[30] “Principal Component Analysis”, Statistixl.com, 2020. [Online]. Available: https://www.statistixl.com/features/principal-components/. [Accessed: 12- Jun- 2020].

[31] F. Awin, A. Younan, D. Corral-De-Witt, K. Tepe, and E. Abdel-Raheem, “Real-Time Multi-Channel TVWS Sensing Prototype Using Software Defined Radio,” IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2018.

[32] T. Fawcett, “An introduction to ROC analysis,” Pattern Recognition Letters, vol. 27, no. 8, pp. 861–874, 2006.

[33] P. D. Arvindpdmn, “Confusion Matrix,” Devopedia, 20-Aug-2019. [Online]. Available: https://devopedia.org/confusion-matrix. [Accessed: 12-Jun-2020].

VITA AUCTORIS

NAME:	Abdalaziz Mohammed
PLACE OF BIRTH:	Jeddah, Saudi Arabia
YEAR OF BIRTH:	1985
EDUCATION:	Bachelor of Science in Electrical Engineering, University of Khartoum, Khartoum, Sudan, 2006. Master of Applied Science in Electrical Engineering, University of Windsor, Windsor, Ontario, Canada, 2020.