DETECTION OF MINI/MICRO UNMANNED AIR VEHICLE (UAV) UNDER CLUTTER PRESENCE AND ENVIRONMENTAL EFFECTS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

İSMAIL GÖKHAN DERE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2019

Approval of the thesis:

DETECTION OF MINI/MICRO UNMANNED AIR VEHICLE (UAV) UNDER CLUTTER PRESENCE AND ENVIRONMENTAL EFFECTS

submitted by **İSMAİL GÖKHAN DERE** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Eng.** _____

Prof. Dr. Mustafa Kuzuoğlu
Supervisor, **Electrical and Electronics Eng., METU** _____

Examining Committee Members:

Prof. Dr. Umut Orguner
Electrical and Electronics Engineering, METU _____

Prof. Dr. Mustafa Kuzuoğlu
Electrical and Electronics Eng., METU _____

Prof. Dr. Asım Egemen Yılmaz
Electrical and Electronics Engineering, Ankara University _____

Prof. Dr. Özlem Özgün
Electrical and Electronics Engineering, Hacettepe University _____

Assoc. Prof. Dr. Emre Özkan
Electrical and Electronics Engineering, METU _____

Date: 11.12.2019

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: İsmail Gökhan Dere

Signature:

ABSTRACT

DETECTION OF MINI/MICRO UNMANNED AIR VEHICLE (UAV) UNDER CLUTTER PRESENCE AND ENVIRONMENTAL EFFECTS

Dere, İsmail Gökhan
Master of Science, Electrical and Electronics Engineering
Supervisor: Prof. Dr. Mustafa Kuzuoğlu

December 2019, 73 pages

Detection, tracking and classification of Unmanned Air Vehicles (UAVs) is an emerging and crucial capability of radars in recent years. In the presence of clutter such as a crowded city or a foggy weather above sea surface, mini and micro UAVs become very difficult for radars to detect, track and classify. Classification information of UAV targets can be very useful for the critical infrastructures in order to provide security. Examined studies imply that kinematic and characteristic features such as Doppler velocity, Radar Cross Section (RCS) fluctuations and Signal-to-Noise (SNR) are helpful features for separating drones from other moving targets. However, selection of method and extracted features have an impact on success of classification. In this study, Support Vector Machine (SVM) classification method is proposed to be used as an inclusive, useful and flexible method. Moreover, additional classification stage is proposed in this study in order to increase the success rate of separating mini/micro UAV targets from the clutter targets which has very similar characteristic properties to drones. Experiments conducted in this study for the selection of classification method and features also show that additional classification stage has an improving impact on success rate. Thus, a method is proposed in this study including comparison of classification methods, features and improvement method.

Keywords: Mini/Micro Unmanned Air Vehicle, Clutter, Classification, Support Vector Machine, Feature Selection.

ÖZ

KARGAŞA VARLIĞI VE ÇEVRESEL ETKİLER ALTINDA MİNİ/MİKRO İNSANSIZ HAVA ARACI (İHA) TESPİTİ

Dere, İsmail Gökhan
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği
Tez Danışmanı: Prof. Dr. Mustafa Kuzuoğlu

Aralık 2019, 73 sayfa

İnsansız Hava Aracının (İHA) tespit, takip ve sınıflandırması son yıllarda radarlar için ortaya çıkan ve önem kazanan bir yetenektir. Kalabalık bir şehir veya deniz yüzeyi üstündeki sisli hava gibi bir kargaşa varlığında, mini/mikro İHAlar radarlarca tespiti, takibi ve sınıflandırması zor hedefler haline gelir. İHA hedeflerinin sınıflandırma bilgisi, kritik tesislerin güvenliğinin sağlanması için oldukça yararlı olabilir. İncelenen bazı çalışmalar Doppler imzası, Radar Kesit Alanı (RKA) salınımı ve Sinyal Gürültü Oranı (SGO) gibi kinematik ve karakteristik özellikleri dronları hareketli diğer hedeflerden ayırmak için yararlı öznitelikler olarak sunmaktadır. Bu çalışmada, Destek Vektör Makineleri (DVM) sınıflandırma yöntemi kapsayıcı, yararlı ve esnek bir yöntem olarak önerilmektedir. Ayrıca, bu çalışmada mini/mikro İHA hedefleri dronlara çok benzer karakteristik özelliklere sahip olan kargaşa hedeflerinden ayırma başarı oranını arttırmak için ilave bir eğitim aşaması önerilmektedir. Bu çalışmada sınıflandırma yöntemi ve özniteliklerin seçimi için gerçekleştirilen denemeler, ilave bir sınıflandırma aşamasının başarı oranını arttıran bir geliştirici etkiye de sahip olduğunu göstermiştir. Böylece, sınıflandırma yöntemlerinin, özniteliklerin karşılaştırmalarını ve performans arttırma yöntemini içeren bir çalışma sunulmuştur.

Anahtar Kelimeler: Mini/Mikro İnsansız Hava Aracı, Kargaşa, Sınıflandırma, Destek Vektör Makinesi, Öznitelik Seçimi

To my beloved family and dear friends…

ACKNOWLEDGEMENTS

I would like to express that it was a great opportunity for me to study with Prof. Dr. Mustafa Kuzuoğlu during this study to to gain experience in the area of radar applications. I am grateful to him for his guidance and encouragements to complete this study.

I would like to thank to my beloved family members, my mother Cemile Dere, my father Hasan Adnan Dere, my sister Emine Gözde Dere and my wife Gonca Özten for their continuous support during the period of study.

Tevfik Bahadır Sarıkaya played a key role by advising new and unique ideas to solve the problems in my thesis. I would like to express my endless gratitude to him hereby.

I would like to express my gratitude to my dear friends who helped me a lot to concentrate on my study both mentally and physically, Müntaz Faydalı, Mehmet Onur Padar, İzzet Serbest and Ataç Sancar.

Finally, I would like to thank to my company ASELSAN Inc. for the equipments and facilities I was provided with and my manager Yusuf Bora Kartal for his continuous support.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS

RF	:	RADIO FREQUENCY
FT	:	FOURIER TRANSFORM
RADAR	:	RADIO DETECTION AND RANGING
CFAR	:	CONSTANT FALSE ALARM RATE
CW	:	CONTINUOUS WAVE
PRI	:	PULSE REPETITION INTERVAL
PRF	:	PULSE REPETITION FREQUENCY
PW	:	PULSE WIDTH
FMCW	:	FREQUENCY MODULATED CONTINUOUS WAVE
GSR	:	GROUND SURVEILLANCE RADAR
ANN	:	ARTIFICIAL NEURAL NETWORK
SVM	:	SUPPORT VECTOR MACHINE
GMM	:	GAUSSIAN MIXTURE MODEL
LO	:	LOCAL OSCILLATOR
K-NN	:	K NEAREST NEIGHBOR
DAC	:	DISCRIMINANT ANALYSIS CLASS
ADC	:	ANALOG TO DIGITAL CONVERTER
RCS	:	RADAR CROSS SECTION
ML	:	MAXIMUM LIKELIHOOD
T/R	:	TRANSMIT/RECEIVE
STING	:	STATISTICAL INFORMATION GRID-BASED
UAV	:	UNMANNED AIR VEHICLE

CHAPTER 1

INTRODUCTION

1.1. Statement Of The Problem

Worldwide-used ground surveillance radars which are designed for military and civilian uses aim to detect targets of interest according to the radar type. UAV (Unmanned Air Vehicle) is an emerging target to be detected by ground surveillance radars for military missions as well as pedestrians, cars, tanks and convoys. With the increasing technology of UAVs, ground surveillance radars needed to specialize in detection, tracking and classification algorithms.

Radar systems aim to detect UAVs using signal processing techniques and use detections consecutively for tracking and classification stages. Detection of the targets is carried out by transmitting electromagnetic wave toward a region of interest and processing the reflections. Signal processing techniques such as CFAR (Constant False Alarm Rate) applications are used in the generation stages of detections and tracks consequently. Tracking is the next step which is applied by radars within models, user-defined rules and filters and generates information about targets in a report structure. Track reports contain information of tracks that are generated by radar tracking algorithms.

Seperation of UAVs from non-UAV targets is carried out by using either track reports or detection reports depending on the data processing capability of radar processing unit. Appropriate method is selected for this usage in the presence of clutter. Features extracted from track reports are determined to classify and selected classification method is applied in classification stage. Selection of convenient discriminative features and usage of consecutive stages make drones discrimination from not drone targets more successful.

1.2. Scope Of The Thesis

This thesis presents the problem of UAV discrimination from the clutter environment with a radar system and proposes a solution that is composed of data-driven SVM classification method using track information. The motivation of choosing this method is that SVM classification method is a supervised learning tool which can be useful for small data space using the selected features from track reports of UAV and non-UAV targets. SVM is used in this thesis as a data-driven method that captures kinematic and electromagnetic discriminative properties of UAV. Training section of SVM is crucial to classify UAV in the clutter existence since it brings design flexibility by choosing desired discriminative features for classification purpose.

Tracking algorithms provide tracks from the moving and consistent detections with gating functions. Moreover, track reports are more useful for classification than detection reports since tracking model brings more features about targets. For this reason, classification is carried out based on track reports in this thesis. By establishing the UAV data gathering setup with the Pulse Doppler radar, in the varying scenarios track reports are established by the period of rotating radar. Thus, for determined several scenarios, features are extracted from track reports within this rotation period. However, for less complexity and less work load at the processor, it is important to have smaller feature space. Discriminative and uncorrelated features make classification algorithms work more efficiently and training of system easier.

The first step of this thesis study is to determine an optimal classification method based on track data obtained by a Pulse Doppler radar. This selection is data dependent since classification success depends on data properties such as data variety, amount, resolution along with the discriminativity of features gathered by radar which may change from radar to radar. Therefore, the optimal method for gathered data is aimed to be reached by the analysis.

The second step of this study is building the structure of the selected classification model which has the best performance in the first step. SVM model is selected because

of its high performance of separating UAV from targets in the field of interest based on gathered track data as well as its flexibility and usage efficiency in the radar processor. In order to form the SVM structure, convenient Kernel functions is selected by comparing the performance from a variety of Kernel equations.

After the decision of Kernel function, track reports are used to extract the features and select the best features for SVM method depending on the track data. In order to generate a reliable finite training data set with the smallest dimension, the features are selected as uncorrelated as possible. In addition, selecting uncorrelated features also provides unrepeated training functions. Therefore, finding an appropriate feature set increases the efficiency of classification.

At the next step training feature set and test set is determined to complete SVM classification. Defining the classification model parameters the method is applied and the classification performance is analyzed. This study aims to reach the best classification performance by selecting the optimal method, optimal kernel functions and best separating feature set. The observed performance shows that the defined classes give similar reactions to some features. Therefore, discrimination performance of UAV from clutter is tried to be increased with an extra SVM stage.

Finally, the performance increasing second stage of SVM is applied and the results are analyzed.

1.3. Organization Of The Thesis

In the context of this thesis, there are six chapters for composition of UAV classification problem and solution. In the first chapter, introduction to thesis structure, problem to be focused and the scope of the study are stated.

The second chapter presents background information about radars, brief radar operation principles, operational steps and definition of UAVs. A theoretical radar introduction to radar systems with radar equation calculation is presented and then basic principles of mini/micro UAVs are examined in this chapter.

In the third chapter, background of classification methods is presented for the selection to be used in this thesis. Learning algorithms with definition of supervised and unsupervised learning are examined. Clustering and component analysis as widely used unsupervised learning methods and supervised learning methods such as regression and classification are inspected.

The fourth chapter presents the data collection from a pulse doppler radar system for the usage in the thesis. Radar installation for collecting UAV data in the presence of different types of clutter is mentioned with the important points and brief information of UAV flight scenarios.

In the fifth chapter, selection of appropriate classification method is examined through the data collected with the radar. Several classification methods are compared and discussed by using the collected data and the motivation of selecting the classification methods is included in this chapter. Feature extraction and selection of features that are used in SVM method are presented using the results of conducted experiments. Then, kernel is selected by comparing some kernel equations through the collected radar track reports. Once the classification methods and kernels are analyzed and chosen, UAV and non-UAV discrimination with the training and test stages is applied. This classification success is examined and analyzed by using the results. At the next part, a method for increasing the classification success is proposed and applied. The increase of success of the applied method for the gathered radar track data is presented. This chapter contains the explanation of the reached results of the UAV classification in the clutter representation.

The sixth chapter is composed of conclusive remarks about the thesis and discussions of future considerations. The results of the experiment and classification study which is examined in the fifth chapter are discussed with future research suggestions in Chapter 6.

CHAPTER 2

UAV DETECTION AND TRACKING WITH RADAR

2.1. Radar Background

In this chapter, background information about radar, detection and tracking processes of pulse doppler radar in clutter presence, feature extraction and feature set gathered with radar are given. At the beginning of the chapter, basic information about radars is given. Following the background information, UAV concept, detection and tracking UAVs in clutter presence with radars are examined. The second chapter includes topics covering the features which are extracted from Pulse Doppler radar.

Radar (Acronym which stands for Radio Detection and Ranging) is the name of the electrical system which includes sophisticated transducer/computer components to detect, locate, identify, image and classify the targets [2]. It works by transmitting radiofrequency electromagnetic waves toward a region of interest in space and collecting the reflected signal from objects in that region. The reflected signal from the objects in the region of interest includes the information to determine the presence of target as well as its range, velocity and other features. The electronical components of radars may change according to the purpose of the system. However, as illustrated in Figure 2.1, radar system is generally combined by antenna, transmitter, receiver and signal processor at least [1]. Antenna is the mechanical passive device for transmitting electromagnetic (EM) waves into the medium after the signal is generated by the transmitter. As well as transmission use, antenna is a receiver subsystem which takes EM waves from the medium and introduce to the receiver. Antenna is connected to the transmitter and receiver through a transmitter/receiver (T/R) device (i.e. circulator or switch) which provides isolation and simultaneous selection of circuits. The reflected signal from the object in the medium radar radiated is applied to the receiver

circuits. In this section, radio frequency (RF) signal is amplified, converted into intermediate frequency (IF), applied to analog digital converter (ADC) and then introduced to signal processor by using a detector to remove the carrier from the modulated return signal. Depending on radar needs, receiver part usually contains amplifiers, mixers, matched filters, local oscillators and ADCs. Received signal from the target is processed by the radar receiver in the signal processor to determine the properties of target and carry out detection and tracking steps. As well as waveform and features selected to use in radar applications, range, reliability, cost, size and maintainance are also important parameters to influence radar design, transmitter and receiver selection [2]. Together with the issues above, in the radar design, target parameters and operation conditions such as climate, atmospheric attenuation, atmospheric refraction, reflection and clutter property are taken into consideration for the best detection performance. In order to reach the best performance, selection of radar operation frequency, antenna type, waveform parameters are important factors for design process.

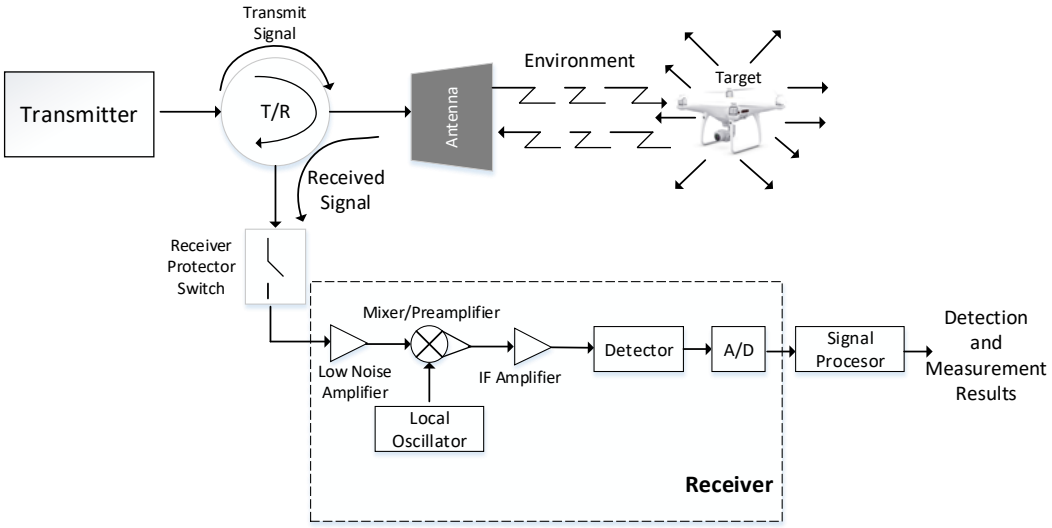

Figure 2.1 Radar Basic Principles and Components

Radars are electronical devices built to radiate EM waves at several frequency ranges at their best performances. For varying missions, radar operation frequency is

determined taking performance parameters such as range, resolution, antenna size and production parameters such as cost, reliability and maintainability into consideration. Widely used specific radar frequency bands have been assigned by International Telecommunication Union (ITU)[3]. These frequency band ranges and covering radar frequency bands can be seen in Table 2.1.

Table 2.1 Radar frequency bands

Radar Band	**Frequency Range**	**Bands Assigned**
HF	3-30 MHz	
VHF	30-300 MHz	138-144 MHz 216-225 MHz
UHF	300-1000 MHz	420-450 MHz 890-942 MHz
L Band	1-2 GHz	1.215-1.4 GHz
S Band	2-4 GHz	2.3-2.5 GHz 2.7-3.7 GHz
C Band	4-8 GHz	5.250-5.925 GHz
X Band	8-12 GHz	8.50-10.68 GHz
K_u Band	12-18 GHz	13.4-14 GHz 15.7-17.7 GHz
K Band	18-27 GHz	24.05-24.25 GHz
K_a Band	27-40 GHz	33.4-36 GHz
V Band	40-75 GHz	59-64 GHz
W Band	75-110 GHz	76-81 GHz 92-100 GHz
Milimeter Waves	110-300 GHz	126-142 GHz 144-149 GHz 231-235 GHz 238-248 GHz

Curry calls common radar center frequencies in several bands UHF, L-band, S-band, C-band and X-band with their advantages and disadvantages for varying applications [4]. As Curry presents in his publications, HF, VHF, UHF and L-band radars have relatively larger antennas for radar applications than the radars using upper frequency bands such as S-band, C-band, X-band, K_u-band, K-band, K_A-band, V-band, W-band and millimeter wave [4]. Additionally, by providing narrow beams with relatively small antennas, target precision measurement is better in the upper band radars such as X-band, K_u-band, K-bands than the lower band radars while radars are generally not suggested to operate in the atmospheric conditions because of severe atmospheric and rain losses at the upper frequencies above K_A-bands [4]. Operation frequency selection in radars directs the designer to the next step which is considering the optimum antenna type selection for reliable and considerable performance.

Table 2.2 Radar band characteristics and applications

Frequency Band	**Band Characteristics**	**Applications**
HF	Very large antennas	OTH radar
VHF	Large antennas	Search radar
UHF, L	Large antennas	Search radar
S, C	Medium size antennas Moderate measurement precision	Multifunction radar
X, K_u, K	Small antennas Precision measurement	Tracking radar Airborne radar
K_u, K, K_a	Very small antennas Good measurement precision Atmospheric and rain loss	Short-range radar Precision-guidance radar
V, W and millimeter	Severe atmospheric and rain loss	Space-to-space radar

The antenna which transmits EM energy into the region of interest and receives reflected energy from a target, is one of the most important elements of radar [1]. As can be seen in Table 2.2, radar antenna selection depends partially on frequency band that radar operates. In order to reach proper antenna operation performance, the radar

antenna is electrically matched with transmitter, receiver and free space having the capability of radiation type selectivity [1]. As illustrated in Figure 2.2, radiation type selectivity means that radar antenna transmits into space when transmitter generates EM wave and receives EM waves from the space and introduces to the receiver while in reception.

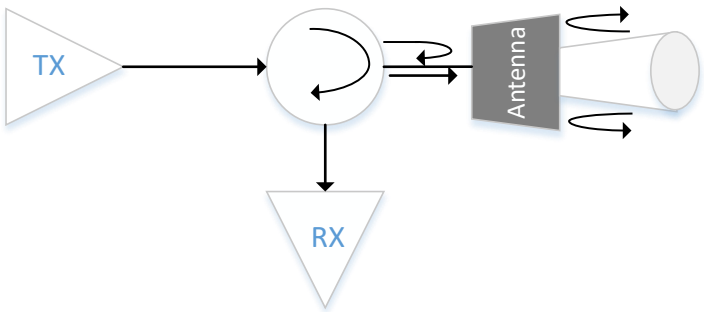

Figure 2.2 Radar antenna TX-RX matching

Depending on the radar design, waveform is chosen to fulfill the requirements of radar system. Radar waveforms are generally divided into four classes; which are CW, chirp pulses, phase coded waveforms and pulse bursts. CW waveforms usually do not require much effort to transmit and to process signal [4]. Chirp pulses waveform is a generally used pulse-compression technique that is capable of providing good range resolution while using relatively high energy containing long pulses [4]. Phase coded waveforms provide proper range resolution similarly to chirp pulses by including a combination of subpulses which are transmitted with relative phases [4]. Pulse-burst waveform is a summation of a train of pulses, separated in time, processed coherently in the receiver matched filter [4]. In this thesis, K-band Pulse Doppler radar which uses Doppler effect phenomena usually in signal processing algorithms is used to detect and track UAV targets in the presence of clutter. Pulse Doppler radars extract and use Doppler frequency shift related with its pulse repetition interval and duty cycle. A frequency shift occurs in the received signal from the target if it has a velocity

resulting with a movement. This phenomena is called “Doppler effect” which is a widely used term for radar applications.

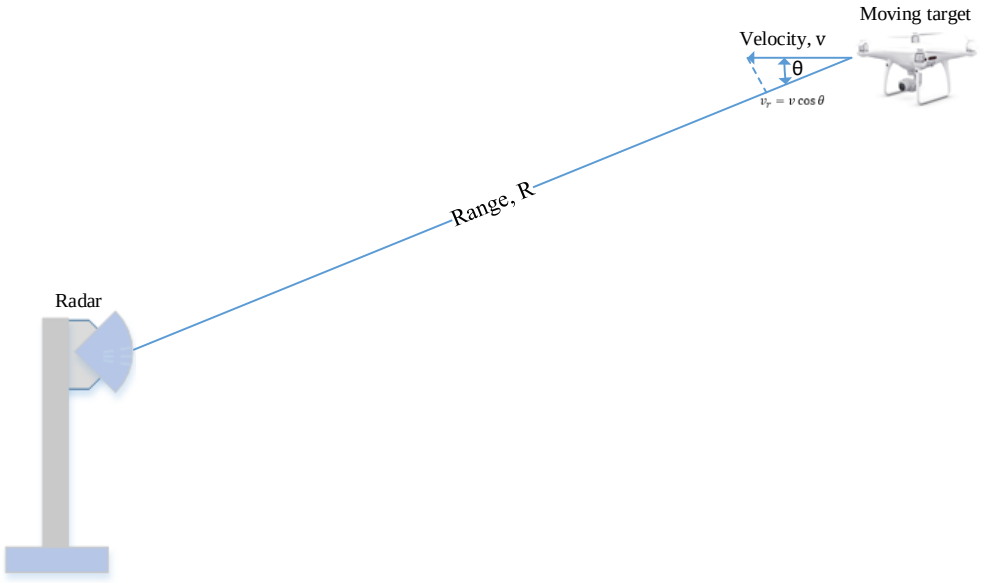

Figure 2.3 Geometry of radar and target in deriving the Doppler frequency shift

Assuming that R is the range from the radar to the target and wavelength of the electromagnetic wave is λ, v_r is the velocity of moving target towards to the radar, as it is illustrated in Figure 2.3, then the Doppler frequency shift can simply be obtained as;

$$f_d = \frac{2v_r}{\lambda} = \frac{2v_r f_0}{c} \quad (1)$$

where f_0 is the frequency of radar transmission and c is the velocity of electromagnetic wave propagation, which is approximately 3 x 10^8 m/sec.

Pulse Doppler radars uses the information Doppler frequency shift which is generated in pulsed waveforms.

2.2. Radar Detection and Tracking Under Clutter Presence

As explained in Section 2.1, radar design includes several important steps such as selection of frequency band, antenna and waveform to develop a system fulfilling the requirements. After completing the system design of a radar, radar equation is an

important milestone for designer to measure the radar capability of detecting the targets. The radar equation expresses the relationship between range of a radar and system hardware characteristics, target parameters and region of interest [2]. For an isotropic antenna which radiates transmitter power P_T in all directions uniformly, the power density at a distance R is calculated as in equation (2) where surface area is $4\pi R^2$ with an imaginary sphere of radius R.

$$Power\ density\ at\ a\ range\ R\ from\ an\ isotropic\ antenna = \frac{P_t}{4\pi R^2} \tag{2}$$

If radar has a directive antenna, maximum gain G of antenna is defined as in equation (4).

$$G = \frac{Maximum\ power\ density\ radiated\ by\ a\ directive\ antenna}{Power\ density\ radiated\ by\ an\ isotropic\ antenna} \tag{3}$$

Using equations (2) and (3) together, the power density at the target from a directive antenna with a transmitting gain G is in (4).

$$Power\ density\ at\ range\ R\ from\ a\ directive\ antenna = \frac{P_t G}{4\pi R^2} \tag{4}$$

For radar receiver, the echo signal is usable in the equations. Power density returned to the radar from a target is determined by RCS of the target that is denoted by σ.

$$Reradiated\ power\ density\ \ back\ at\ the\ radar = \frac{P_t G}{4\pi R^2}\ \frac{\sigma}{4\pi R^2} \tag{5}$$

Received signal power is calculated in equation (6) as effective area A_e times reradiated power density back at the radar where A_e is physical area times antenna aperture efficiency:

$$P_r = \frac{P_t G}{4\pi R^2}\ \frac{\sigma}{4\pi R^2}\ A_e\ = \frac{P_t G A_e \sigma}{(4\pi)^2 R^4} \tag{6}$$

In the case the same antenna is used for both transmission and reception, from the antenna theory, transmit gain is calculated as in equation (7) where ρ_a is antenna aperture efficiency and λ is wavelength:

$$G = \frac{4\pi A_e}{\lambda^2} = \frac{4\pi \rho_a \mathrm{A}}{\lambda^2} \tag{7}$$

Maximum range of radar R_{max} is the maximum distance to detect the target and can be calculated when received signal power Pr equals to minimum detectable signal power S_{min}.

$$R_{max} = \left[\frac{P_t G A_e \sigma}{(4\pi)^2 S_{min}}\right]^{1/4} = \left[\frac{P_t G^2 \lambda^2 \sigma}{(4\pi)^3 S_{min}}\right]^{1/4} = \left[\frac{P_t {A_e}^2 \sigma}{4\pi\lambda^2 S_{min}}\right]^{1/4} \tag{8}$$

The available thermal noise power generated at the input of a receiver of bandwidth B_n at a temperature T, where k is Boltzmann constant, is calculated as

$$Available\ thermal\ noise\ power = kTB_n \tag{9}$$

In order to extract S_{min} from equation (8), noise figure is written as:

$$F_n = \frac{Noise\ out\ of\ practical\ receiver}{Noise\ out\ of\ ideal\ receiver\ at\ T_0} = \frac{N_{out}}{kT_0BG_a} = \frac{S_{in}/N_{in}}{S_{out}/N_{out}} \tag{10}$$

$$S_{in} = \frac{kT_0BF_nS_{out}}{N_{out}} \tag{11}$$

If the minimum detectable signal S_{min} is that value of S_{in} which corresponds to the minimum detectable signal-to-nose ratio at the output of IF, $(S_{out}/N_{out})_{min}$, then

$$S_{min} = kT_0BF_n\left(\frac{S_{out}}{N_{out}}\right)_{min} \tag{12}$$

Exchanging equation (12) in equation (8), we reached to R_{max} as shown in equation (13)

$$R_{max} = \left[\frac{P_t G A_e \sigma}{(4\pi)^2 kT_0BF_n(\frac{S}{N})_{min}}\right]^{1/4} \tag{13}$$

If P_{av} is desired to use in the equation in the case that transmitter waveform is a train of rectangular pulses of width τ and constant pulse repetition period $T_p = \frac{1}{f_p}$:

$$P_{av} = \frac{P_t \tau}{T_p} = P_t \tau f_p \tag{14}$$

Assuming integration of pulses is carried out, for n-pulse integration integration efficiency for post detection integration is defined as:

$$E_i(n) = \frac{(\frac{S}{N})_1}{n(\frac{S}{N})_n} \tag{15}$$

By taking system losses such as antenna losses, beamshape loses, propagation effects and signal processing losses into account, where L_s is system loss, L_f is fluctuation loss, radar equation becomes as in equation (16)

$$R_{max}{}^4 = \frac{P_{av} G A \sigma n E_i(n) \rho_a F^4 e^{-2aR_{max}}}{(4\pi)^2 k T_0 F_n (B\tau) f_p (\frac{S}{N})_1 L_f L_s} \tag{16}$$

Regarding the radar equation as in equation (16), one of the most important missions of radar systems is detection of interested targets. Information from targets of interest is contained in the radar pulses which may also include undesired clutter signals, receiver noise or unintentional jamming. Threshold detection technique is usually used for optimal performance of target detection in the presence of unintentional clutter [5]. Detections are obtained after applying a precomputed and structured threshold to the signals with complex valued magnitudes received by receiver. The signals received by radar receiver become detections if their magnitude is above threshold level while signals with lower magnitudes are not tagged as detections as seen in Figure 2.4. Since the detection stage is a statistical process, decisions made by threshold include a finite probability of error which occurs as false alarms [5]. Unless false alarm detections are directed in a complex algorithm, these may cause false alarm tracks that decrease radar reliability. The aim of this thesis is to propose a way to

decrease false alarm rate of UAV tracks which are generated in detection and tracking stages of radar.

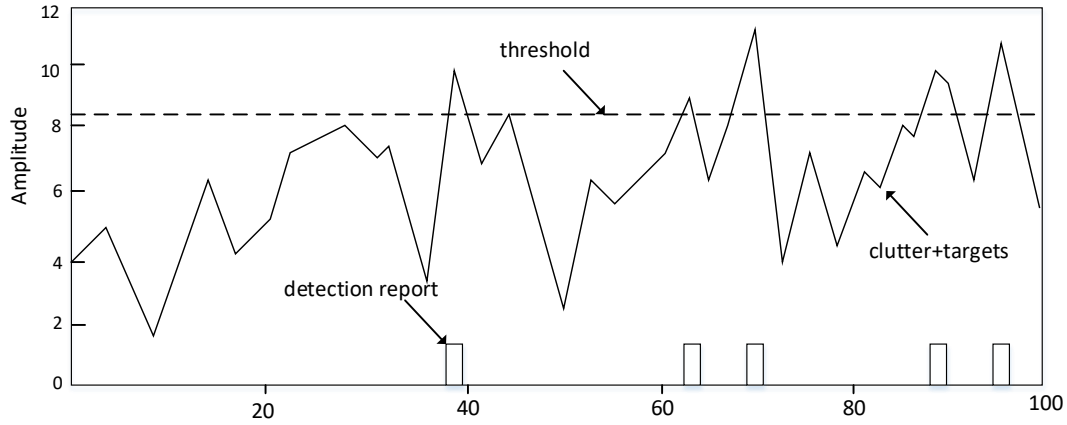

Figure 2.4 Detection of targets and clutter using threshold

Radar detection performance is calculated depending on probability of detection P_D, probability of false alarm P_{FA}, threshold level, S/N value, statistical characteristics of target and noise echoes where detection threshold level is determined taking signal to noise ratio to obtain desired P_{FA} [4]. Threshold level is monitored and controlled in some radar applications, while some radar systems have specialized circuits of adjusting threshold level automatically to reach desired P_{FA}. This specialized constant false alarm (CFAR) signal processing technique which may cause a loss which is often a couple of dBs have derivations such as CA-CFAR, SOCA-CFAR, GOCA-CFAR and Adaptive CFAR [5]. In this thesis context, track information which are obtained from detections by applying CFAR detection technique, are examined.

Table 2.3 False Alarm examples depending on radar parameters

	Radar-1	Radar-2	Radar-3	Radar-4	Radar-5
P_{FA}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-8}
B	1 MHz	1 MHz	100 MHz	100 MHz	100 MHz
R_W	Continuous	150 km	Continuous	1.5 km	Continuous
PRF		500 Hz		500 Hz	
T_{FA}	1/second	0.5/second	100/second	0.5/second	1/second
t_{FA}	1 sec	2 sec	0.01 sec	2 sec	1 sec

As seen in Table 2.3, depending on the system parameters false alarms occur because of noise or clutter. Clutter is all of undesired radar returns or signals received from objects such as earth surface, sea, atmosphere, chaff, rain, snow and other than the target which radar intends to detect [6]. Although SNR or SIR exists in the radar equation as in (16), SCR is also usable as the ratio of the target RCS to clutter RCS in the equation in case interference is taken as surface or atmospheric clutter [5].

2.3. UAV Definition and Detection With Radar

UAV (acronym for Unmanned Air Vehicle) is simply defined as a vehicle that flies with no pilot on board. UAVs can be classified by mission, weight, payload capacity, operational altitude, velocity, rotor number or RCS. In a study, Korchenko suggests a generalized classification of UAVs using 16 fundamental features which are aircraft applications, type of control system, flight rules, airspace classification, aircraft types, wing types, take-off/landing, aircraft engine types, fuel system, fuel tank types, number of exploitations, category (according to the weight and maximum range of flight UAV), flight radius, flight altitude and aircraft functions [7].

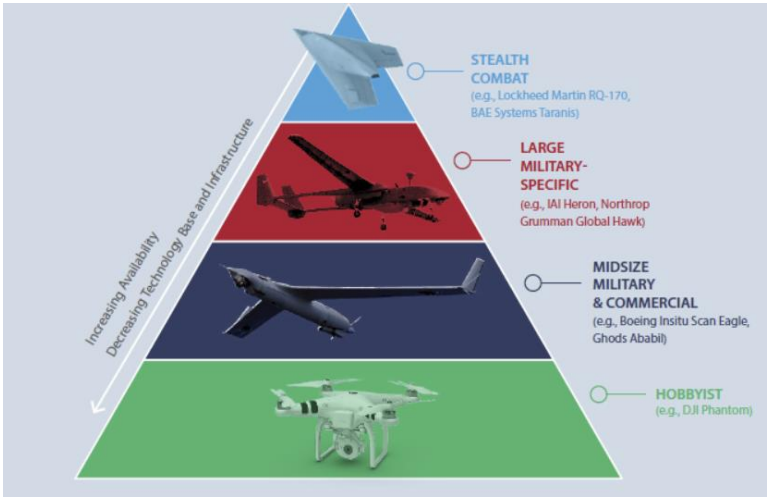

Figure 2.5 UAV Classification Pyramid [11]

Sayler groups UAVs into hobbyist drones, Midsize Military and Commercial drones, Large Military-Specific drones and Stealth Combat drones as Figure 2.5 illustrates a classification pyramid which was introduced by Sayler to classify UAVs according to their usage and technological complexities [11]. Hobbyist type has the least complexity and cheapest drones while midsize military and commercial type has more complexity and cost. Large military-specific drones are less available in the market and stealth combat drones are at the top of pyramid with least availability and the most complexity. This thesis focuses on mini and micro UAVs which are two classes defined by NATO's UAV classification as shown in Table 2.4 [8].

Table 2.4 UAV Classification [8]

Class	Category	Operational Altitude	Mission Radius (km)	Payload Capacity (kg)
UAV Class-1 (<150 kg)	Micro (< 2 kg)	< 90 m (300 ft)	5	0.2 – 0.5
	Mini (2-20 kg)	< 900m (3000 ft)	25	0.5 – 10
	Small (<150 kg)	< 1500 m (5000 ft)	50-100	5 -50
UAV Class-2 (150-600 kg)	Tactical	< 3000 m (10000 ft)	200	25 – 200

Lee et al. examines a drone detection and identification system which captures images of the unknown drones by a camera and identifies the drone model type by matching the image to the samples in the database [35]. Having a camera instead of radar in his system, Lee uses Deep Convolutional Neural Network technique to match the unknown drone image to the ones in the database [35].

Choi et al. proposes a study to classify drones by using deep learning algorithms of which inputs are images based on simulated micro-Doppler signatures [36]. For this

aim, micro-Doppler signatures are simulated by changing the number of rotors, rotor speed and orientation and transformed into images to be used in a convolutional neural network. Choi simulates micro-Doppler signatures from several drones with varying number of rotors as given in Figure 2.6 [36].

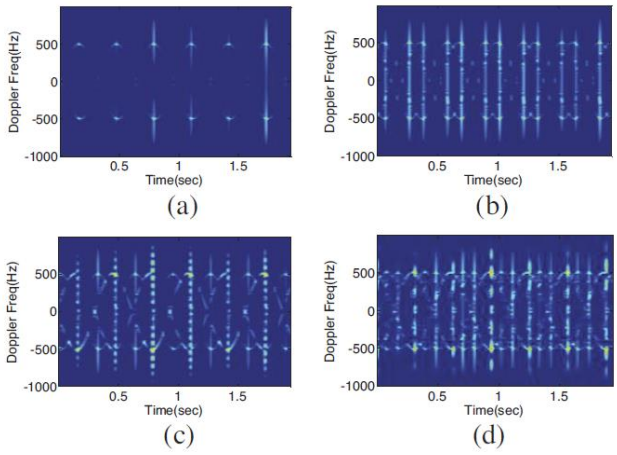

Figure 2.6 Simulated Micro-Doppler signatures of drones with a)1 rotor, b)2 rotors, c)3 rotors, d)4 rotors[35]

De Quevedo examines a DJI Phantom 4 commercial micro drone that is used for both privately and professionally for its detection and RCS measurement with a FMCW X-Band radar [9]. In his experiments, he uses outdoor-recorded data, collected with DJI Phantom 4 and calculated RCS in X-band of the micro UAV as in Figure 2.7 [9]. In this thesis, the same model drone is used for detection, tracking and classification purposes.

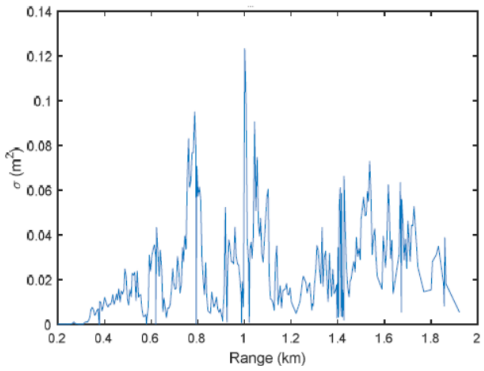

Figure 2.7 RCS Measurement of DJI Phantom 4 with X-band FMCW Radar [9]

In another study, Ritchie uses a Vector Network Analyser with a wideband Satimo quad ridge reference horn antenna in an anechoic chamber to analyze RCS measurements of DJI Phantom drone in several frequency bands such as L-band, S-band and C-band [10]. He continues his research with outdoor experiments to produce micro-doppler spectrograms of DJI phantom drone using a pulsed Doppler radar system [10].

In another study, Hofele presents an algorithm which uses the preprocessed time signals as inputs for automatic radar targets classification by using the recorded spectrum and cepstrum data of the radar SPEXER 2000 to classify pedestrians, vehicles, drones and helicopters [37].

2.4. Track Information Obtained By Radars

Track information context that acquired by a Pulsed Doppler radar which is used in this thesis is given in Table 2.5. With the more track properties obtained by a radar, it is more possible to separate the real UAV target from the non-UAV targets.

Table 2.5 Track information acquired by radars

Kinematic Property	**Track Info**	**Characteristic Features**
Range	Timestamp	SNR
Azimuth	Track ID	RCS
Velocity	Beam ID	Microdoppler signature
Acceleration	Status	
Elevation Angle	Class	
Height	Update Status	

CHAPTER 3

CLASSIFICATION METHODS BACKGROUND

3.1. Introduction

Chapter 2 gives a description of detection and tracking of mini/micro UAVs with Pulse Doppler radars. Tracking stage of radar extracts information of drone to the classification stage in order to separate UAV from non-UAV targets. In classification stage, specialized algorithms are applied to classify radar track information as given in Table 2.5. In Chapter 3, target classification in radars is defined and discussed in detail. Briefly, classification means that a decision is carried out for a sample in which class it belongs by using previous samples and their classes. Variety of applications such as visual recognition, speech recognition or fingerprint identification mechanisms get benefits of pattern recognition and classification algorithms. In addition to various applications, many electronical warfare systems classify detected objects using different methods. As a branch of electronical warfare systems, some radar applications apply target classification software. Radar target classification methods are based on data which previously obtained from environment in order to use the classes of targets in radar missions.

Duda et al. summarizes sub-problems of pattern classification as extracting features, noise, overfitting, mereology, difficulties on model selection, prior knowledge of classes, missing features, segmentation, context, invariances, evidence pooling, costs and risks of classification and computational complexity [12]. Artificial neural networks which provides solution models for problems like pattern, recognition and forecasting are generally divided into three groups as supervised methods, unsupervised methods and reinforcement learning [13]. Classification and regression

are two examples to supervised learning methods while clustering and principal component analysis are unsupervised learning methods.

Main motivation of machine learning is investigating how computers can improve their performance based on data [17]. Machine learning which has common applications with statistics, game theory and optimization is needed when tasks are too complex to program and beyond human capabilities [23]. Supervised and unsupervised learning algorithms are combined with semi-supervised learning and active learning algorithms to construct machine learning discipline. In this chapter, supervised and unsupervised learning algorithms, pattern recognition techniques, advantages and disadvantages of these methods are proposed.

3.2. Unsupervised Learning

Unsupervised learning methods are purely based on the observation of raw input data without information or errors from previous data set and useful for unlabeled data processing before supervised learning [14]. Unsupervised learning model includes unsupervised word since the model is able to learn and shape information with no error signal to compute the potential solution [13]. Unsupervised learning techniques are widely used in variety of fields such as patients grouping, online shoppers grouping and online viewers of web sites.

As explained in reference [13], application of unsupervised learning algorithm is based on three phases. These are competition, cooperative and adaptive phases. In competition phase, the input pattern x, also called neuron, which has the minimum inner product of synaptic weight w is selected from all input patterns. Cooperative phase is the layer that the selected neuron is centered and cooperative neurons are gathered in a neighborhood topology. In adaptive phase, selected neuron group's individual values of discriminant function are increased by adjusting synaptic weight functions. Repeating these phases synaptic weight vectors get their final and stable state to be coherent with input patterns in the means of unsupervised learning [13].

3.2.1. Clustering

Clustering is a learning algorithm that finds a proper and valid organization of data. By collecting and grouping similar objects in a data set, a cluster aims to set of similar entities to be separated from the other clusters. Everitt et al. explains that clusters are combination of high density related data while the other clusters are low density data regions and distance from one point to another in cluster is less than distance between the point in that cluster and a point in another cluster [15].

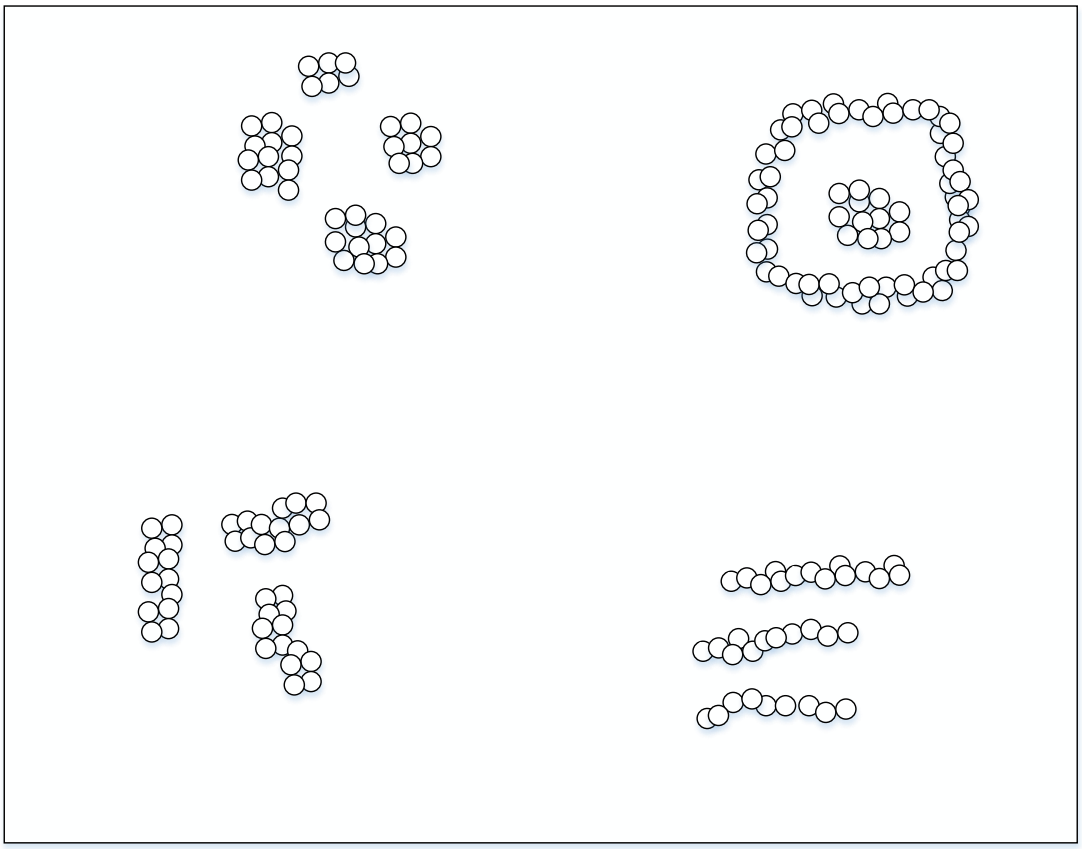

Figure 3.1 Clusters in two dimensions

Figure 3.1 illustrates an example of clustering set of points in two dimensions. Usually points in a high dimensional space are examined in this method to find the similarities, regularities and properties which are useful to group input data. A distance measure is determined and used to define the similarity of the input data points. Several

applications such as image compression, bioinformatics and online trade websites use clustering algorithms.

3.2.1.1. Partitional Clustering Methods

Partitional clustering methods also referred as partitioning methods or nonhierarchical methods which are often used in engineering applications aim to generate a single partition of the data to group. The input data is required to be in the form of a pattern matrix in the context of these methods with the advantage of being appropriate for large-scaled data applications [16]. Given a data set, $X = [x_1, x_{2,} \dots, x_n]$ and the number of clusters to form, k, partitioning algorithm divides data set into k partitions ($k \leq n$). Each partition represents a cluster which is formed to optimize a criterion in order to generate similarity with points in cluster and dissimilarity with other clusters [17]. K-Means algorithm is one of the most common partitional clustering algorithms. Supposing that the data set $X = [x_1, x_{2,} \dots, x_n]$ contains n objects in Euclidean space and partitioning algorithm distributes data set into k clusters $C_1, C_{2,} \dots, C_k$ with the assumptions of $C_i \subset D$ and $C_i \cap C_j = \emptyset$ for ($1 \leq i\ ,\ \ j \leq k$). In K-means clustering algorithm, each cluster's center is represented by the mean value of the objects in the cluster. This is initiated by arbitrarily choosing k objects from $X\ as$ the initial cluster centers, then each object is reassigned to the most similar-behavior cluster, based on the mean value of the points in the cluster. At the next step, calculating the mean value of the points for each cluster, the clusters are updated [17]. The cluster quality is measured by the sum of squared error, E, between all points in C_i and the center points c_i defined in (22) where p is the point in space representing a given object:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i}^{k} dist(p, c_i)^2 \tag{17}$$

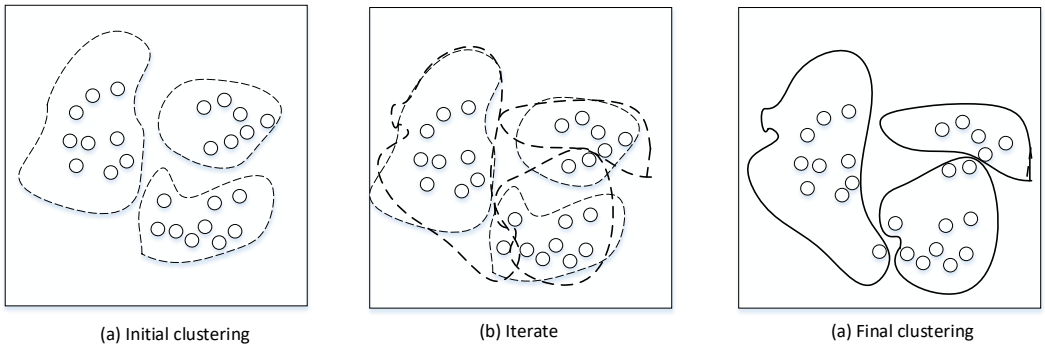

Figure 3.2 K-means clustering steps

As illustrated in Figure 3.2, the algorithm tries to minimize the error function in (17) and iteration continues until the final clustering. Burney et al. states that K-means clustering method is an algorithm that quickly group data based on predefined feature vectors and initial centroids although estimating centroids from data is sometimes expensive to compute and not guaranteed to give the best result [18].

3.2.1.2. Hierarchical Clustering Methods

Hierarchical clustering methods create hierarchical groups from input data objects. These methods may be divided into three categories as algorithmic methods, probabilistic methods and Bayesian methods. Algorithmic methods define clusters by treating objects as deterministic structures and using deterministic distances between these objects while probabilistic methods use probabilistic models to determine clusters and fit them to the models. On the other hand, Bayesian models aims to find a distribution of possible clustering structures according to given data set [17]. Algorithmic methods are agglomerative, divisive and multiphase methods. As illustrated in Figure 3.3, agglomerative method uses a bottom-up strategy by taking each object as a cluster firstly then merges it into larger clusters in contrast to divisive methods that have a top-down strategy and splits objects from initial root cluster [17]. Both methods are iterative and user may limit iteration specifying a desired number of cluster. Multiphase methods are carried out by using clustering features such as trees or dynamic modeling which bring the advantage of scalability and improve the previous clustering steps. However, sometimes algorithmic methods are difficult to

apply. This occurs when distance measure may be difficult to select in some cases or data may be observed partially causing computation not conductable. Since algorithmic methods use heuristic steps, for some cases decision of good merging or splitting and optimization goal of resulting cluster hierarchy may be challenging [17].

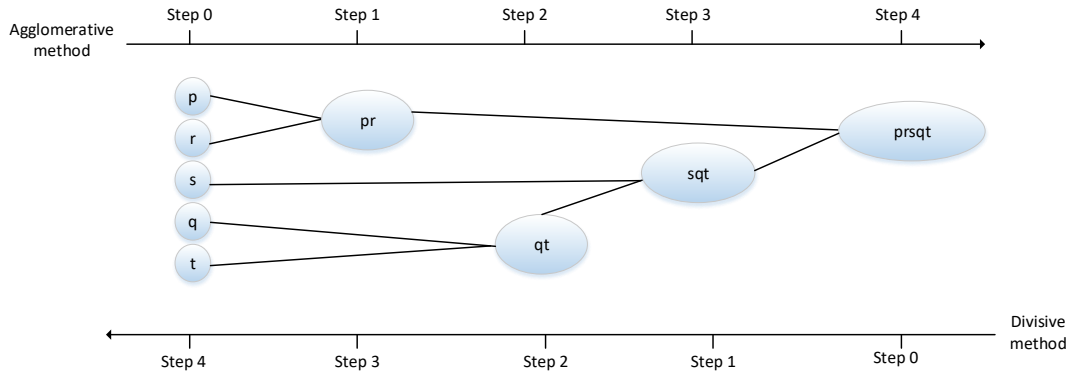

Figure 3.3 Agglomerative and Decisive Hierarchical Clusters

In order to decide which objects involve in the next steps of cluster, four distance measurement criteria are listed below:

Minimum distance : $dist_{min}(C_i, C_J) = \min\{|p - p'|\} \quad , \; p \in C_i \; , \; p' \in C_j$ (18)

Maximum distance : $dist_{max}(C_i, C_J) = \max\{|p - p'|\} \quad , \; p \in C_i \; , \; p' \in C_j$ (19)

Mean distance : $dist_{mean}(C_i, C_J) = |m_i - m_j| \quad , \; p \in C_i \; , \; p' \in C_j$ (20)

Average distance : $dist_{avg}(C_i, C_J) = \frac{1}{n_i n_j} \sum_{-\infty}^{\infty} |m_i - m_j| \, , p \in C_i \, , \; p' \in C_j$ (21)

In equations (18), (19), (20) and (21) distance criteria are given where $|p - p'|$ is the distance between two objects $p \; and \; p'$, m_i is the mean for cluster, C_i, n_i is the number of objects in C_i. For variety of algorithmic hierarchical clustering problems, convenient one from distance criteria in (18-21) is selected and used in an easy and efficient way [17].

Probabilistic hierarchical clustering methods aim to estimate the generative model which is assumed to adopt common distribution functions, such as Gaussian

distribution or Bernoulli distribution as accurately as possible using the observed data objects to be clustered. By finding the parameters of distribution functions for which the model best fits the observed data set, task of learning for generative model is achieved [17]. Let $X = [x_1, x_{2,} \dots, x_n]$ is 1-D point data set and generated by a Gaussian distribution which is defined in equation (22) where μ is mean and σ is variance:

$$\mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{22}$$

The probability that a point $x_i \in X$ is then generated by the model is

$$P(x_i|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{23}$$

Consequently, the likelihood that X is generated by the model is

$$L(\mathcal{N}(\mu, \sigma^2): X) = P(X|\mu, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \quad . \tag{24}$$

To find mean and variance parameters, the likelihood is maximized where $max\,\{L(\mathcal{N}(\mu, \sigma^2): X)\}$ is called the maximum likelihood:

$$\mathcal{N}(\mu_0, {\sigma_0}^2) = argmax\,\{L(\mathcal{N}(\mu, \sigma^2): X)\} \tag{25}$$

For a data set partitioned into m clusters $C_1, C_2, \dots, C_m$, where $P()$ is the maximum likelihood, clustering quality Q is:

$$\mathrm{Q}(\{C_1, C_2, \dots, C_m\}) = \prod_{i=1}^{m} P(C_1) \quad . \tag{26}$$

For two merged clusters in hierarchical clustering, $\prod_{i=1}^{m} P(C_i)$ is constant for any pair of clusters. Thus, for given clusters C_1 and C_2, the distance between them is

$$\text{dist}(C_i, C_j) = -\log \frac{P(C_1 \cup C_2)}{P(C_1)P(C_2)} \quad (27)$$

In the concept of hierarchical clustering algorithms, graphs, dendrograms and proximity matrices are used for visualization and problem explanations.

3.2.1.3. Density Based Methods

Density based clustering methods are methods that model clusters as dense regions in data set, aim to find core points with dense neighborhoods to identify convex regions accurately. An example for these methods is density-based clustering based on connected regions with high density. Using a fixed neighborhood size parameter ε and a neighborhood density threshold b, it is decided whether a neighborhood is dense or not. If the ε-neighborhood of a point has at least b points, a new cluster C is created and the point is added to C. The whole data set is visited by applying these steps iteratively and final clustering structure is reached [17]. Another example for density-based methods is the method of ordering points to identify the clustering structure. This method uses a hierarchical technique which creates a graph showing the clusters of different densities and outputs a cluster ordering [19]. The last example for density-based clustering methods is the method of clustering based on density distribution functions. This method is considered as a generalization of several clustering methods and uses kernel density estimation to effectively reduce the influence of noise [17].

3.2.1.4. Grid Based Methods

Similarly to the density based clustering techniques, grid-based clustering methods are useful for large multidimensional data sets using a grid data structure which is independent of input data points. Bandyopadhyay summarizes the grid-based clustering method application as 5 steps in [19]:

- Forming the grid architecture by mapping the data space into grids.

- Calculating the grid density of each cell by using all points within a grid.

- Separating the grids according to their densities.

- Finding the position of cluster centers by using separated grids.

- Traversing the neighbor grids.

As an example, statistical information grid-based clustering (STING) method seperates data space into layers and cells in a hierarchical and recursive way and aims to find statistical parameters of these layers. Mean, maximum and minimum values of each cell is calculated and stored by partitioning each cell at a high level in order to construct the cells at the next level [17]. Another method is an Apriori-like subspace clustering method which focuses on finding density-based clusters in pre-formed subspaces. This method aims to identify search spaces using monotonicity of dense cells regarding the dimensionality and form convenient clusters [17].

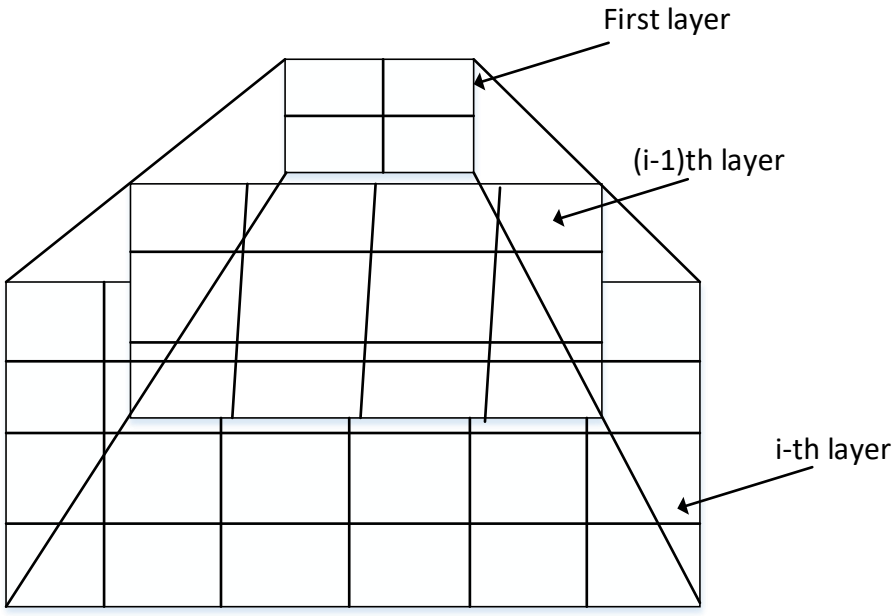

Figure 3.4 Structure of grid-based clustering

3.2.2. Component Analysis Methods

Principal component analysis (PCA) is a method of data decorrelation computing orthogonal basis vectors for input data which is also assessed as a correlation-based clustering method in some sources. PCA is a beneficial method for signal processing applications that has a data-driven approach. The main motivation of this method is to

reduce the dimension of a data set consisting of many interrelated variables to a smaller number of dimensions. Principle components are introduced as a new uncorrelated set of variables for this task in order to present the data in all input variables with decreased set of variables [20]. As Smith explains in [21], PCA is used with statistical background information in many fields such as image compression and computer vision. For that reason, to carry out an analysis using this method requires following steps:

- Having a proper data set,

- Calculating the mean and subtracting it from each of the data dimensions,

- Calculating the covariance matrix,

- Calculating the eigenvectors and eigenvalues of the covariance matrix,

- Choosing components and forming a feature vector,

- Deriving the new data set [21].

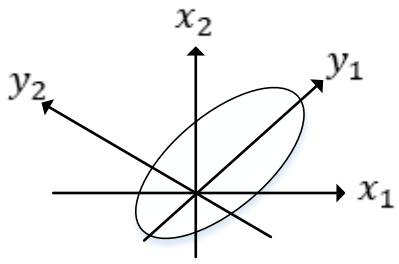

Figure 3.5 An example of PCA using Y1 and Y2 vectors

PCA is usually achieved with three possible methods such as eigenvalue decomposition, singular value decomposition and expectation minimization algorithm. Eigenvalue decomposition and singular value decomposition methods require matrix calculations which may bring some difficulties for large scale data sets [21]. Roweis states in [22] that since eigenvectors of covariance matrix brings computational costs, expectation-minimization algorithm is useful to obtain principle components of the data without calculating covariance matrix.

3.3. Supervised Learning Methods

Supervised learning which is a branch of machine learning algorithms aims to discover the relationship model between input and output. The supervision term of this method is a result of training structure with the labeled points or occasions. In a supervised learning method, there is a set of variables referred as inputs or predictors that have influence on one or more outputs that also are called responses or dependent variables and the goal is to predict the output values by using the inputs [23]. Supervised learning techniques are often used in variety of disciplines such as data mining, information technology, plant control, forecasting and robotics.

Osisanwo et al. examines a real-world problem applying supervised machine learning as illustrated in Figure 3.6 and proposes a standard formulation of the supervised learning task as learning the mapping function and creating a classifier [25]. As Osisanwo explains, function to learn maps a vector into classes and classifier uses examples in training set to classify new instances.

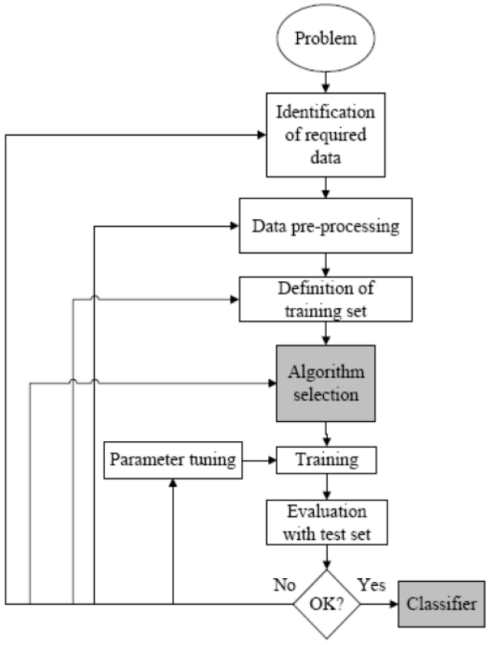

Figure 3.6 Process summary of supervised machine learning [25]

3.3.1. Regression

Regression learning models map input data set into a real-value domain providing usually quantitative continuous responses. Haykin explains in his book that regression model includes the scenario below [27]:

- There is a dependent variable in the set of random variables, referred as response

- The remaining random variables, referred as regressor, which explain the statistical behavior of the response

- An additive error term, referred as expectational error, is included by the dependence formula of the response on the to present the uncertainties

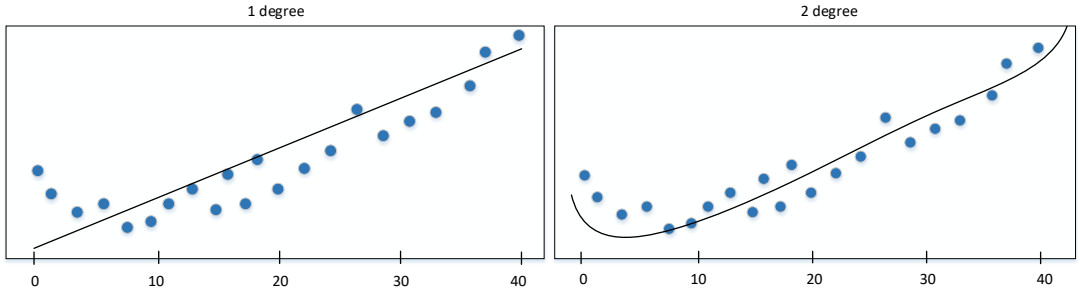

Figure 3.7 Examples of a) linear regression, b) polynomial regression

As some regression examples given in Figure 3.7, regression learning techniques are used in some prediction problems for stock market prices, age of a viewer watching a video, amount of an element in the body or temperature at a building [26].

3.3.1.1. Linear Regression Methods

To define, linear regression is a statistical method that models the relationship between input data and real valued output. For the input data set $X = [x_{1,} \dots, x_n]$, the goal of this method is to find a linear function that best approximates the relationship between variables [24]. Linear regression model aims to model the relationship between regressor and response as given in Figure 3.8.

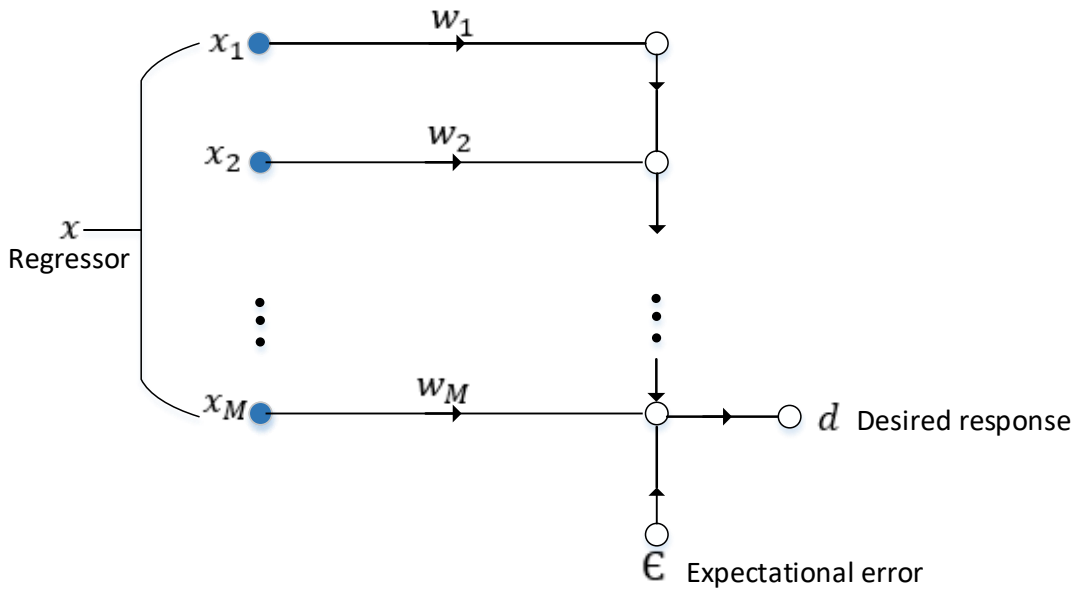

Figure 3.8 An environment modelling with linear regression

Shalev-Shwarts states that in linear regression, hypothesis class of predictors is simply the set of linear functions which is given in Eq.(28).

$$L_d = \{ x \mapsto \langle w, x \rangle + \varepsilon \} : w \in \mathbb{R}^d , \varepsilon \in \mathbb{R} \tag{28}$$

Supposing that $X = [x_{1,} \dots, x_n]^T$ is the regressor and d is the corresponding response, the regression model can be proposed as

$$d = \sum_{j=1}^{M} w_j x_j + \varepsilon = w^T x + \varepsilon \tag{29}$$

where $w_{1,} \dots, w_m$ denote the set of unknown parameters and ε is the expectational error. In Equation (29), M is called as model order and the parameter vector is defined as $w = [w_{1,} \dots, w_m]^T$ of which dimensionality is the same as x. Now, regression problem can be stated as estimating the unknown parameter vector w with the given correlation matrix of the regressor X and variance of the corresponding desired response D while assuming that means of both X and D are zero [27]. Bayesian methods as explained by Hastie, Maximum Likelihood Estimation method as

explained by Murphy and Maximum-A-Posteriori method as explained by Haykin can be used for the solution of this regression problem [23,26,27].

3.3.1.2. Logistic Regression Methods

In linear regression models explained in previous chapter, a fitting problem may occur when a straight line is fit to a binary response [28]. To solve this problem, regression model can be changed to logistic model by using hypothesis class associated with the logistic function which is also called sigmoid function and given in Eq. (30) [24].

$$H_{sig} = \emptyset_{sig} \circ L_d = \{ x \mapsto \emptyset_{sig}(\langle w, x \rangle) \} : w \in \mathbb{R}^d$$

$$\emptyset_{sig}(z) = \frac{1}{1 + \exp(-z)} \tag{30}$$

Starting with defining the logistic function, logistic regression methods use the following steps as stated by James in [28]:

- Fitting logistic regression models using one of the various techniques,

- Estimating the regression coefficients,

- Making predictions multiple times iteratively if needed.

Murphy examines that after model specification step, to estimate the coefficients some algorithms such as MLE, steepest descent, Newton's method, iteratively reweighted least squares or variable metric methods can be used.

3.3.1.3. Regression Trees

Decision trees are efficient hierarchical data structures using nonparametric model parameters which bring the cost of calculating the distances from the given input to all training instances. Regression trees are constructed from a given labeled training set like classification trees. Alpaydın examines that regression trees differ from classification trees with impurity measurement [29].

Defining number of nodes is m, subset of X reaching node m is X_m,

$$d_m(x) = \begin{cases} 1, & x \in X_m : x \text{ reaches node m} \\ 0, & otherwise \end{cases} \quad (31)$$

Using mean square error from the estimated value where $g_m(x)$ is the estimated value of node m and $N_m = X_m = \sum_t^{\infty} d_m(x^t)$

$$E_m = \frac{1}{N_m} \sum_t^{\infty} (r^t - g_m)^2 d_m(x^t) \quad (32)$$

In a node, mean of the required outputs of instances reaching the node g_m is

$$\frac{\sum_t^{\infty} d_m(x^t) r^t}{\sum_t^{\infty} d_m(x^t)} \quad (33)$$

Then, E_m corresponds to the variance at m. If at a node, the error is acceptable, then a leaf node is created and it stores the $g_m$value. If the error is not acceptable, data reaching node m is split further such that the sum of the errors in the branches is minimum. At each node, the attribute that minimizes the error is desired, and recursion is continued.

As James explains that regression tree is built following the steps below [28]:

- Recursive binary splitting is used to grow a large tree on the training data, stopping only when each terminal node has fewer than some minimum number of observations.

- Apply cost complexity pruning to the large tree in order to obtain a sequence of best subtrees, as a function of α.

- Use K-fold cross validation to choose α. That is, divide the training observations into K folds. For each k=1,….,K:

- Repeat Steps 1 and 2 on all but the kth fold of the training data and evaluate the mean squared prediction error on the data in the left-out kth fold, as a function of α.

- Average the results for each value of α, and pick α to minimize the average error.

- Return the subtree from Step 2 that corresponds to the chosen value of α.

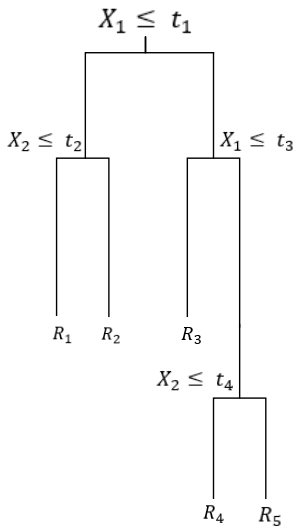

Figure 3.9 An example of regression tree

3.3.2. Classification

For the cases that the response variable is quantitative, regression methods are used as stated in 3.3.1. On the other hand, in order to predict a qualitative response which is referred as categorical for an observation classification process is more convenient to use [28]. In varying sources, logistic regression method is stated as a classification method for this property of response type.

The classification methods aim to learn a mapping from inputs x to outputs y, where $y \in \{1, \dots, C\}$ and C is the number of classes. In the case $C = 2$, classification is named as binary classification. If $C > 2$, classification is called as multiclass classification. Classification tasks can be called as generalization task since they may include predictions of the first-seen inputs. Thus, training set which is an important part of generalization process provides the criteria to define classes. In addition, classification methods generally use probability concepts to define the problem and present the best guess. Denoting the probability distribution over possible labels for a training set D by $p(y|x, D)$, length vector of C is represented by this probability. In binary classification, $p(y = 1|x, D) + p(y = 0|x, D) = 1$. Alpaydın states that the

empirical error is the proportion of training instances where predictions of hypothesis class, h not matching the required values given in training set D is in Eq.(34)[29]:

$$h(x) = \begin{cases} 1, & if\ h\ classifies\ x\ as\ a\ positive\ example \\ 0, & if\ h\ classifies\ x\ as\ a\ negative\ example \end{cases} \tag{34}$$

where D contains N such examples and t indexes different examples in the set and

$$x = [x_1 x_2]\ , r = \begin{cases} 1, & if\ x\ is\ a\ positive\ example \\ 0, & if\ x\ is\ a\ positive\ example \end{cases}\ , D = \{x^t, R^t\}$$
$$E(h|\ D) = \sum_{t=1}^{N} 1(h(\ x^t)\ \neq r^t\ ) \tag{35}$$

Classification methods are used in many applications such as email spam filtering, speech recognition, image classification, face recognition, optical character recognition and medical diagnosis.

3.3.2.1. Simple Classifiers

In classification methods, predicting discrete classes or posterior probabilities which are between zero and one is desired. One simple approach to classify is that constructing a discriminant function that directly assigns each input vector to a specific class while another approach is modeling a conditional probability distribution and using it for optimal decision [30]. Bayes classifiers, Linear discriminant analysis (LDA) and Quadratic Discriminant Analysis (QDA) are simple methods used for dimensionality reduction in classification.

3.3.2.1.1. Bayes Classifier

For the case that number of classes is two, Bayes decision theory is useful to define a simple classifier and to minimize the error function given in Eq. 35 [26]. Bayes rule is applied to a classifier to classify a feature vector:

$$p(X = x|Y = y) = \frac{p(X = x, Y = y)}{p(Y = y)} = \frac{p(X = x)p(Y = y|X = x)}{\sum_{x'}^{N} p(X = x')p(Y = y|X = x')} \tag{36}$$

Bayes classifier assigns each test point to the most possible class if predictor values are given. A test observation is assigned with x_0 to the class j for which $f_j(x) = \Pr(Y = j|X = x_0)$ is largest. If there only two possible classes, Bayes classifier results class one if $\Pr(Y = 1|X = x_0) > 0.5$, and class two otherwise. Since Bayes classifier chooses the class j for which $\Pr(Y = j|X = x_0)$ is maximum, the lowest possible test error rate at $X = x_0$ is $1 - max_j \Pr(Y = j|X = x_0)$ where the expectation averages the probability over all possible values of X [28]. For a given training sample x, assuming there are K mutually exclusive classes, C_i, where $i = 1, \dots, K$, $p(x|C_i)$ is the probability of seeing x as the input when it is known to belong class C_i. The posterior probability of class C_i, multiplication of prior probability and likelihood divided by evidence is

$$P(C_i|x) = \frac{p(x|C_i)P(C_i)}{p(x)} = \frac{p(X = x)p(Y = y|X = x)}{\sum_{x'}^{N} p(X = x')p(Y = y|X = x')} \quad (37)$$

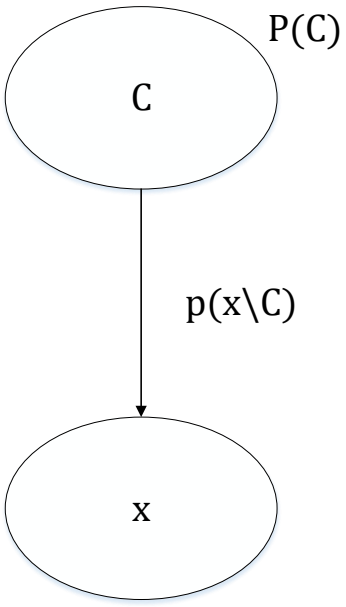

Figure 3.10 A simple variation of Bayesian network for classification

For minimum error, Bayes classifier chooses the class with the highest posterior probability [29].

3.3.2.1.2. Linear Discriminant Analysis

LDA is a popular and stable classification method if number of response classes is more than two and classes are well-separated. The simplest representation of a linear discriminant function is obtained by taking a linear function of the input vector so that $y(x) = w^T x + w_0$ where w is called a weight vector and w_0 is bias. An input vector X is assigned to class C_1 if $y(x) \geq 0$ and to class C_2 otherwise. The relation $y(x) = 0$ defines the decision boundary which is a $(D-1)$-dimensional hyperplane within the D-dimensional input space [30]. Assuming that $f_j(x)$ is Gaussian distributed and each class density is modeled as multivariate Gaussian

$$f_j(x) = \frac{1}{(2\pi)^{\frac{p}{2}} |\sum k|^{\frac{1}{2}}} \, e^{\frac{-1}{2}(x-\mu_k)^T \sum k^{-1}(x-\mu_k)} \tag{38}$$

Hastie states that LDA arises in the special case when assuming the classes have a common covariant matrix $\sum k = \sum \forall k$. In comparing two classes k and l, the log-ratio of those is an equation linear in x [23].

$$\log \frac{\Pr(G = k|X = x)}{\Pr(G = l|X = x)} = log \frac{f_k(\mathrm{x})}{f_l(\mathrm{x})} + log \frac{\pi_k}{\pi_j} \tag{39}$$

$$= log \frac{\pi_k}{\pi_j} - \frac{1}{2}(\mu_k + \mu_l)^T \sum^{-1} (\mu_k - \mu_l) + x^T \sum^{-1} (\mu_k - \mu_l)$$

The equal covariance matrices provides the normalization factors and quadratic part in the exponents to cancel. This linear log-odds function implies that the decision boundary between k and l- the set where $\Pr(G = k|X = x) = \Pr(G = l|X = x)$-is linear in x; in p dimensions a hyperplane. If the space is divided into regions that are classified as class 1, class 2, etc. these regions will be separated by hyperplanes [23]. James states that training data is used to estimate the parameters of the Gaussian distributions since they are not known in practice. $\pi_k = \frac{N_k}{N}$, $where\ N_k$ is the number of class-k observations :

$$\mu_k = \sum_{g_i=k}^{N} \frac{x_i}{N_k} \tag{40}$$

$$\sum = \sum_{k=1}^{K} \sum_{g_i=k}^{K} (x_i - \mu_k)(x_i - \mu_k)^T/(N-K) \tag{41}$$

Quadratic Discriminant Analysis (QDA) is used if $\sum k$ are not equal and cancellations do not occur; in particular the pieces quadratic in x remain. The estimates for QDA are similar to LDA, except that separate covariance matrices must be estimated for each class [26].

3.3.2.2. Nearest Neighborhood

A non-parametric classification model is the model in which the parameters space does not grow with the amount of training data. Nearest neighborhood classification method is an example for usage of non-parametric classifiers. Hastie examines that nearest neighbor prediction rule is a powerful prediction method like linear model fit by least squares [26]. One of the most popular examples for nearest neighborhood methods is K-Nearest neighbor (KNN) classifier. In this method, K points in the training set are selected which are nearest to the test input x_0 and represented by N_0. The members of each class in the K-points set are counted and the empirical fraction is returned as estimate [26]. The conditional probability for class represented by j as the fraction points in N_0 whose response values are equal to :

$$Pr(Y = j|X = x_0) = \frac{1}{\mathrm{K}} \sum_{i \in \mathrm{N}_0}^{\infty} I(\mathrm{y}_i = j) \tag{42}$$

Indicator function in Eq.42 is defined as follows:

$$I(\mathrm{y}_i = j) = \begin{cases} 1, & if\ \mathrm{y}_i = j\ is\ true \\ 0, & if\ \mathrm{y}_i = j\ is\ false \end{cases} \tag{43}$$

As the next step, KNN applies Bayes rule and classifies test inputs x_0 to the class with largest probability. According to the book by James, KNN can often provides classifiers that are close to the optimal Bayes classifier with the fact that the impact of choosing number K on KNN classifier is drastic [28]. The comparison of two KNN classifiers with $K = 1$ and $K = 100$ is illustrated in Figure 3.11.

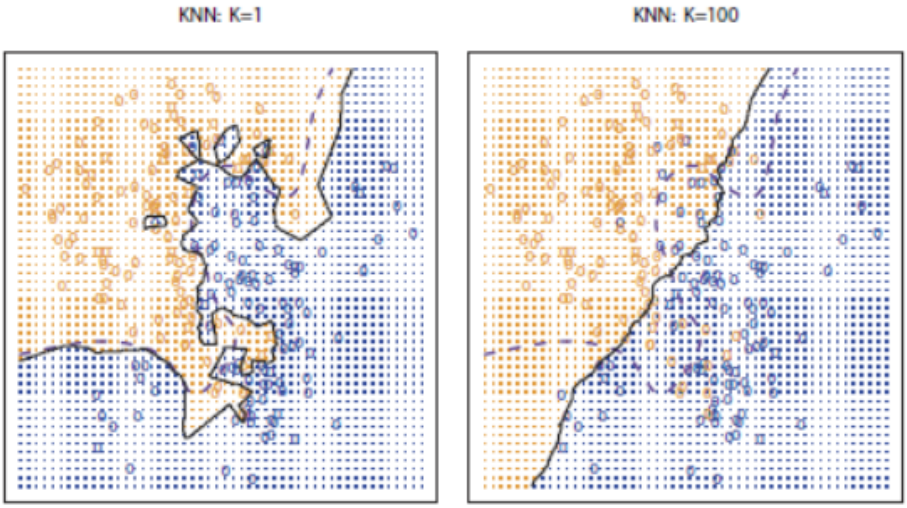

Figure 3.11 Comparison of K-NN classifiers with K=1 and K=100 [28]

KNN method is an example of instance-based learning which uses usually Euclidean distance as distance metric, while other metrics can also be used [26].

3.3.2.3. Naïve Bayes

Naïve Bayes is a simple, useful classification method in the case that feature space dimension is high and density estimation is not necessary [28]. For a given class, $G = j$, features X_k are assumed to be independent in Naïve Bayes model as illustrated in Eq.(44)

$$f_j(X) = \prod_{k=1}^{d} f_{jk}(X_k) \qquad P[X = X_k|Y = y] = \prod_{i=1}^{d} P[X_i = x_i|Y = y] \tag{44}$$

With the assumption given in Eq.(44) and using the Bayes rule, the Bayes optimal classifier can simply be written as in Eq.(45) [24]

$$h_{Bayes}(X_k) = argmax\, P[Y = y|X = X_k] \\ = argmax\, P[Y = y]P[X = X_k|Y = y] \\ = argmax\, P[Y = y]\prod_{k=1}^{d} P[X_i = x_i|Y = y] \quad (45)$$

Using Eq.(45), the number of parameters to estimate is reduced to 2d+1 and this estimation can be made by using the maximum likelihood principle. The resulting classifier is referred as Native Bayes classifier, in which the number of parameters to learn is significantly decreased with the assumption in Eq.(44). [24]

The reason for this method to be called as naïve, is that independency of the features is very unlikely while they do not have to be necessarily conditional on the class label [26].

In addition, James states that the individual class-conditional marginal densities f_{jk} in Eq.(44) can each be estimated separately using 1D-kernel density estimators while an appropriate histogram estimate can be used in the case $X_j$of X is discrete [28].

3.3.2.4. Deep Learning

Deep learning is a specific powerful method of machine learning techniques. For millions of data involving data sets that are difficult to solve by simple methods require complicated operations. Bengio examines that for the deep architectures which represents high level abstractions involving multiple levels of non-linear operations are the subjects of deeper models and more complicated learning algorithms [31]. Deep learning methods may use several algorithms and data structures depending on the need following some applications of unsupervised learning methods. Duda states that before classification using labeled data, unsupervised methods are useful to obtain features and build the structure [12]. Although variety of supervised unsupervised algorithms supply efficient results for learning problems, deep architectures mean more efficiency and compact solutions for highly-varying functions. Compact term for a function means that having few computational elements as possible. For a deep network, depth of architecture refers to the longest path from an input node to an

output node or number of layers while the set of computational elements is the set of computations artificial neurons can make [31].

Deep learning methods can be separated into two branches to cover the applications using deep architectures as deep generative methods and deep neural networks.

3.3.2.4.1. Deep Generative Methods

Deep generative methods are simply divided into three categories as direct, undirect and mixed models. Three examples for each models are illustrated in Figure 3.12.

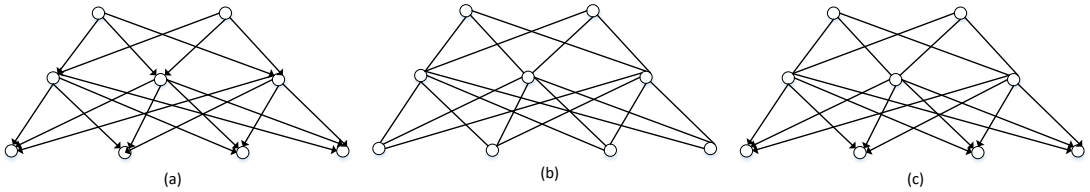

Figure 3.12 Deep generative models examples: (a) Direct, (b)Undirect, (c)Mixed.

Deep directed models also referred as deep directed networks use bottom layers that contains observed data points and hidden layers as illustrated in Figure 3.12(a). Model structure can be constructed by choosing the number and size of layers manually or using non-parametric Bayesian methods or boosting tree structures [26].

Deep undirected models have some advantages over the directed models such that performing efficient calculations since all the nodes in each layer are conditionally independent of each other while these models have a more difficult training section because of partition function [26]. An example of undirected models is Deep Boltzmann machine which is considered as a particular energy-based model and has a couple of hidden layers.

Deep mixed models are considered as partially directed and partially undirected. Supposing that a deep belief network with three hidden layer is used, it may have top two layers acting as an associative memory and the other layers generate the output [26]. Deep belief networks which are graphical models learning to extract a deep hierarchical representation of the training data model the joint distribution between

observed vector x and the l hidden layers h^k where $x = h^0$and $P(h^{l-1}, h^l)$ is the visible-hidden joint distribution [32]:

$$\mathrm{P}(x, h^1, \dots, h^l) = (\prod_{k=0}^{l-2} P(h^k | h^{k+1}))\ P(h^{l-1}, h^l) \tag{46}$$

3.3.2.4.2. Deep Neural Networks

Deep neural networks are generally considered in two separated classes as deep multi-layer perceptrons and deep auto-encoders also referred as auto-associators. Deep neural networks contain convolutional layers and sampling layers and each layer includes topographic structure.

Deep multi-layer perceptrons are used in neural networks with usually back-propagation along with the optimization algorithms and unsupervised learning methods [26].

Deep auto-encoders consisting input layer, output layer and hidden layers have encoders and decoders. The input is encoded by a trained by auto-encoder in order to reconstruct input data from output data [31].

3.3.2.5. Support Vector Machines (SVM)

Support vector machine (SVM) is basically a learning machine which constructs a hyperplane as the decision surface that maximizes the margin of separation between positive and negative examples [27]. Regression and classification methods use SVMs in a variety of models. One of the simplest versions of support vector classifiers is a maximal margin classifier which is intuitive and simple although it is not applicable for most data sets [28]. Support vector machines are usually used for accommodating linear and nonlinear decision boundaries in presence of multiple classes.

A hyperplane is a $(p-1)$-dimensional flat subspace in a p-dimensional space which is defined as equation where $X = (X_1, X_2, \dots X_p)^T$ is a point on the hyperplane:

$$f(x) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p = 0 \tag{47}$$

Supposing that a $n\,x\,p$ data matrix X consisting of n training observations in p −dimensional space and y_i represent classes

$$x_1 = \begin{pmatrix} x_{11} \\ \cdot \\ x_{1p} \end{pmatrix}, \ldots, x_n = \begin{pmatrix} x_{n1} \\ \cdot \\ x_{np} \end{pmatrix} \tag{48}$$

A separating hyperplane separates the training observations according to their class labels and has the property for all $i = 1, \ldots, n$ that

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} > 0 \tag{49}$$

In the case a separating hyperplane exists, a test observation can be assigned a class depending on which side of the hyperplane it is located [28]. The maximal margin classifier uses the maximal margin hyperplane that is used for separating and located farthest from the training observations. The training observations which are equidistant from the maximal margin hyperplane lying along in the width of margin are referred as support vectors. Support vectors are vectors in p-dimensional space and support directly the maximal margin hyperplane if separating hyperplane exists. However, in many cases a hyperplane that almost separates the classes is used if it is not possible to separate classes using a hyperplane [28].

A support vector machine is also referred as a kernel method since inner-product kernel between a support vector x_i and a vector x from the input data space is used while the support vectors consists of a small subset of data points extracted from the training sample [27]. Murphy states in his book that SVMs are combination of kernels and loss functions with the property of being appropriate for binary classification and unnatural from a probabilistic point of view [26].

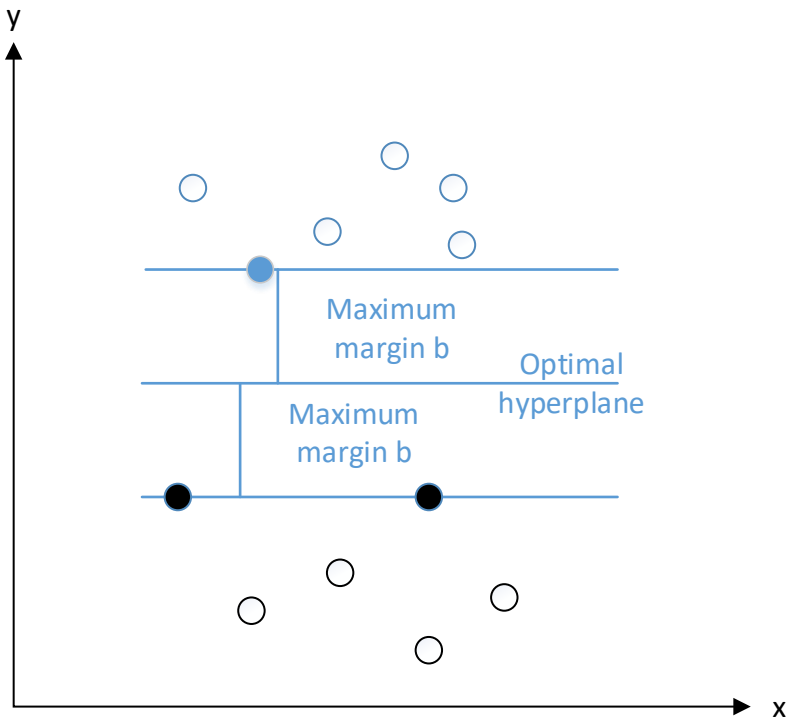

Figure 3.13 An example of optimum hyperplane with maximum margins

The soft margin hyperplanes are used for the support vector classifiers in order to be more robust to individual data points and to classify the most of the training input points [28]. For the nonlinear decision boundaries Kernel functions are derived to the optimal hyperplanes. Linear support vector classifiers are represented as below in Eqn. 50 where $\alpha_i, \dots, \alpha_n$ and β_0 are parameters to estimate in the linear classifier function and S is the collection of indices :

$$f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle \mathrm{x}, \mathrm{x}_i \rangle = \beta_0 + \sum_{i \in S}^{\infty} \alpha_i \langle \mathrm{x}, \mathrm{x}_i \rangle \tag{50}$$

In the nonlinear classifier calculation, kernel function $K(\mathrm{x}, \mathrm{x}_i{}')$ which quantifies the similarity of two observations is used [23] and kernel functions are in the various forms as given below:

$$K(\mathrm{x}, \mathrm{x}_i') = \sum_{j=1}^{p} \mathrm{x}_{ij}\mathrm{x}_{i'j} \tag{51}$$

$$Polynomial\ kernel\ with\ degree\ d\colon K(\mathrm{x}, \mathrm{x}_i') = \left(1 + \sum_{j=1}^{p} \mathrm{x}_{ij}\mathrm{x}_{i'j}\right)^d \tag{52}$$

$$Nonlinear\ function\colon f(x) = \beta_0 + \sum_{i \in S}^{\infty} \alpha_i K(\mathrm{x}, \mathrm{x}_i) \tag{53}$$

$$Radial\ kernel\colon K(\mathrm{x}, \mathrm{x}_i') = \exp(-\zeta \sum_{j=1}^{p} (\mathrm{x}_{ij}\mathrm{x}_{i'j})^2) \tag{54}$$

$$Neural\ network\colon K(\mathrm{x}, \mathrm{x}_i') = \tanh(\kappa_1\langle \mathrm{x}, \mathrm{x}'\rangle + \kappa_2) \tag{55}$$

An example of using radial kernel is illustrated in Figure 3.14.

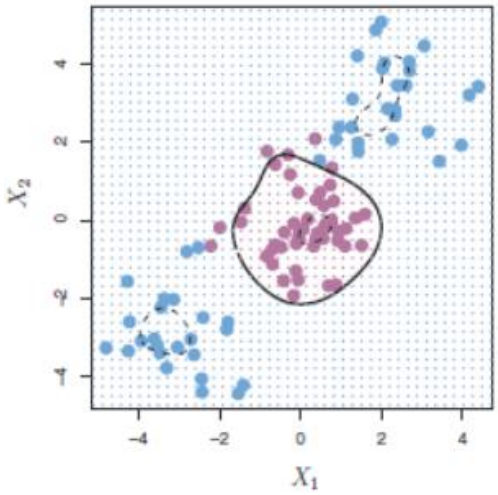

Figure 3.14 Radial kernel example [28]

Kernel function $K(\mathrm{x}, \mathrm{x}_i)$ is symmetric about the center point x_i attaining its maximum value at $\mathrm{x} = \mathrm{x}_i$ while total volume under the surface of the function $K(\mathrm{x}, \mathrm{x}_i)$ is a constant [27]. The optimal decision hyperplane in the output space is expressed as

$$\sum_{i=1}^{N_s} \alpha_i d_i K(\mathrm{x}, \mathrm{x}_i) = 0 \tag{56}$$

Kernel selection for the most accurate SVM can change according to the input data with the fact that Kernel usually makes a small difference in resulting accuracy in practice [17]. However, SVM has a benefit such that being independent of the dimensionality of the transformed space and the complexity of the resulting classifier is characterized by the number of support vectors [12]. Assuming that N_s is the total number of support vectors for n training patterns, the expected value of the generalization error rate is bounded as

$$\epsilon_n[\text{error rate}] \leq \frac{\epsilon_n[N_s]}{n} \tag{57}$$

For $K > 2$ classes, two approaches for applying SVMs are one-versus-one and one-versus-all procedures. One-versus-one approach uses $\mathrm{K\,x\,(K-1)/2}$ SVMs. Each of SVMs compares a pair of classes. Classifying a test observation using each of classifiers, the number of times that the test observation is assigned to each of the K classes. Assigning the test observation to the most frequently assigned class in (2-out-of-K) pairwise classifications is the final classification in this procedure [28]. Comparing one of the K classes to the remaining $K-1$ classes each time K SVMs are fitted in one-versus-all approach. Assuming $\beta_{0_k}, \beta_{1_k}, \ldots, \beta_{\mathrm{p}_k}$ denotes the parameters that result from fitting an SVM comparing to the kth class to the remaining classes, and x^* denotes a test observation, the observation is assigned to the class for which $\beta_{0_k} + \beta_{1_k}\mathrm{x_1}^* + \beta_{2_k}\mathrm{x_2}^* + \cdots + \beta_{\mathrm{p}_k}\mathrm{x}_p{}^*$ is largest. This amounts to a high level of confidence that the test observation is in kth class [28].

CHAPTER 4

DATA COLLECTION

4.1. Data Collection For Classification

Based on the background information that is given in Chapter 2, in the context of this thesis, a pulsed Doppler radar tracking data is used for classification and UAV is aimed to be separated from clutter data by using proper classification technique. This radar data is initially modified to be in the form such that track information includes the necessary information for using in classification stage. Radar tracks should be labeled with the associated classes.

As given in Chapter 2, Pulse Doppler radar tracking process follows reception of target signals, digitalization and processing of signal and forming the detection. Blind range of radar is taken into consideration along with the line of sight for usable detection reports including drone information. Following detection, tracking of target is accomplished in the design rules and restrictions depending to the radar mission and designer decisions. Targets can be classified using the detection information or tracking information. In this thesis, Pulsed Doppler radar tracking data is used for classification and this radar data is initially modified to be in the form such that track report includes the necessary information for classification purposes. Radar track reports are obtained in a period of radar rotation and they should be labeled with the associated classes such as UAV and Not UAV. Track reports with the selected information from the track data structure given in Table 2.5 are saved in the suitable form for classification training.

As well as UAV track data, non-UAV tracks are obtained with the data collection setup. To increase the variation of clutter data, setup is established in three different clutter environments. These clutter environments are city environment, a suburban

environment with wide line of sight and an environment at a tower near to highway. The experiment setup including Pulsed Doppler radar and DJI Phantom 4 drone are established in these environments to collect the varying data for efficient training and test sets.

4.2. Experiment Setup

Based on the background information that is given in Chapter 2, in the context of this thesis, UAV is aimed to be separated from non-UAV target in the field of interest by using proper classification technique which is selected from given methods in Chapter 3. For the purpose of classification, radar is operated in tracking mode to gather track data. For this purpose, radar should operate in tracking mode and the setup including radar and UAV should be established appropriately. Supplying the proper setup, flight scenarios of UAV are defined with the only condition that it flies in the radar line of sight both horizontally and vertically to gather track data.

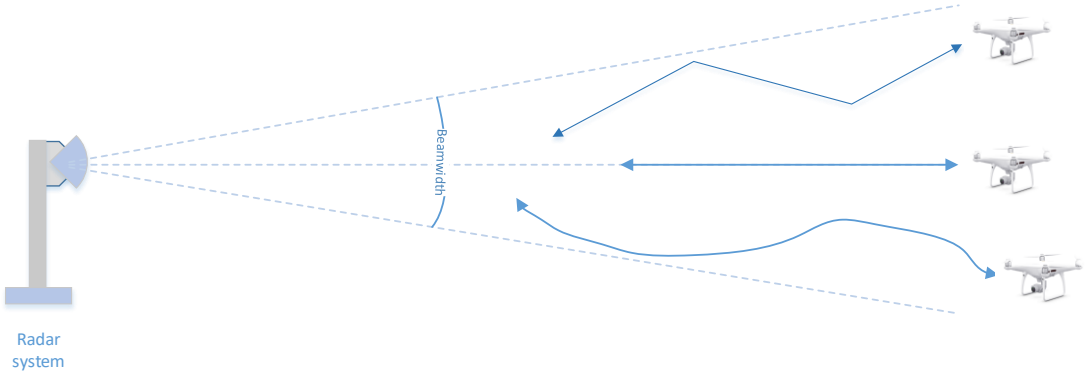

Figure 4.1 UAV flight scenarios in the radar field of view

As illustrated in Figure 4.1 drone moves in the radar line of sight with a velocity component radial to the radar. Radial velocity makes a doppler shift for radar which uses this shift in signal processing stage. Circular flight or hanging in the air of drone is not a source of Doppler shift for Pulsed Doppler radar. For this reason, the flight scenarios include incoming and outgoing motions of drone. In order to use both in training and test data sets, flight scenarios contain drone motions in varying velocity,

acceleration and height. Maneuvering movements are also useful to train drone in varying aspects of RCS.

In the data collection section of this thesis, the track information gathered by the radar contains the parameters according to the radar located at the origin. Therefore, drone was taken off at the nearest point to the radar in order to compare the track data with the data gathered by GPS module of drone. Besides, drone samples were gathered for every single specified run with its track ID in order to reach a realistic classification with the aim of specification of drone motion.

4.3. UAV Preparation

DJI Phantom drone used in the experiments which is subject to classification purpose is able to move in horizontal and vertical directions in the air. As one of the most popular hobbyist drones, this drone gives a good chance to model UAV behaviors.

Figure 4.2 DJI Phantom 4 as a hobbyist drone [11]

With the plastic propellers on the body, it has the capability of reaching a maximum height level and the maximum speed in the battery life. As given in Table 2.5, DJI Phantom 4 drone characteristic features that give the opportunity for complicated maneuvers are important for classifying purposes.

Table 4.1 DJI Phantom 4 Specifications [9]

Weight (propellers and battery included)	1380 g
Diagonal length (propellers not included)	350 mm
Maximum speed	20 m/s (72 km/h)
Maximum flying height (above sea level)	6000 m
Positioning system	GPS / GLONASS
Maximum flying time (battery life)	28 minutes

CHAPTER 5

CLASSIFICATION OF UAV AND NON-UAV TARGETS

In this chapter, classification is carried out using the most appropriate and efficient method which is selected by comparing the best success rates of classification methods with chosen features. In order to reach the best performance with the minimum number of features from a wider feature set, confusion matrix is helpful for assessment of model success.

Obtaining the drone and non-drone track data in the clutter from the setup explained in Chapter 4, data is prepared for classification. Non-drone data obtained from several environments and drone track data gained from DJI Phantom 4 UAV are combined and subjected to the classification. The first step of the classification is determining the model to be used in this study by taking the processor capability and data set width into account. In this study, features are initially determined for the determination of appropriate classification methods. Thus, performance analysis of these methods is done with the determined features for the best success rate. Murphy et al. explains that model selection approaches such as computing misclassification rate, calculating generalization error and generating validation sets can be used [26].

In this study, data obtained from the radar is separated into two groups as training set and validation set for the aim of picking the best model. Cross validation used in this thesis includes five folds. One of these folds is used for test, and the rest of data is used for training data. Then, this procedure is repeated for all of the folds and the success of the model is calculated by the average of all these five runs of which a schematic is given in . Confusion matrix which consists number of samples located in each class gives the amount of samples predicted as drone for real drone data and the

samples labeled as not-drone for real not-drone data. The success rates are calculated by the ratio of numbers in the rows of confusion matrix.

Figure 5.1 Five folds-cross validation used in performance analysis

5.1. Feature Extraction

Features are determined from a larger feature set for the aim of obtaining the best performance with the minimum number of features. Feature set consists the information which is taken from drone and not-drone tracks of a Pulsed Doppler radar. In his book, Jain explains that feature selection methods include several approaches such as exhaustive search, branch-and-bound search, best individual feature, sequential forward selection, sequential backward selection, plus l-take away-r selection and sequential forward floating selection [33]. In this thesis, sequential forward floating approach is used for feature selection similarly to the way explained in [34].

For the selected methods such as complex tree, linear discriminant analysis, logistic regression, linear SVM, quadratic SVM, cubic SVM, fine Gaussian SVM and fine K-NN feature selection using sequential forward floating approach is given below. Features with the number are given with F and its number shortly in the tables.

nD/nD is the short notation of prediction of not-drone for not-drone data, D/D is short notation of prediction of drone for real drone data and average is the mean value of nD/nD and D/D since the amount of data for these classes is almost equal.

The explanations of the tables from 6 to 14 are given as:

- Light green: Single feature that is decided to be used since it increases the success rate when combined with the previous combination of features.
- Red: Combination of features that is decided to be used in the method since it has the highest success rate in the combinations for the method.
- Dark green: Combination of features that has better success rate than the combination in the one step before. (The first combination is compared with the single feature having the highest success rate.)
- Yellow dashed line: Unused single features that decreases the success rate of the combination in the one step before.

- Complex Tree: 6 features (nD/nD: 97.68%, D/D: 99.52%, Average: 98.6%)

Table 5.1 Complex Tree method feature selection

Features	nD/Nd	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	67.58%	32.42%	96.54%	3.46%	82.06%	•
Standard deviation of velocity (F2)	75.86%	24.14%	92.29%	7.71%	84.07%	•
Mean Acceleration (F3)	76.56%	23.44%	94.43%	5.57%	85.49%	•
Standard deviation of acceleration (F4)	72.23%	27.77%	98.62%	1.38%	85.42%	•
Mean RCS (F5)	58.05%	41.95%	96.62%	3.38%	77.33%	•
Standard deviation of RCS (F6)	22.08%	77.92%	95.34%	4.66%	58.71%	-
Mean Height (F7)	74.01%	25.99%	88.11%	11.89%	81.06%	•
Standard deviation of height (F8)	14.25%	85.75%	99.27%	0.73%	56.76%	-
F3+F4	76.02%	23.98%	98.40%	1.60%	87.46%	
F3+F4+F2	78.43%	21.57%	8.36%	1.64%	88.39%	
F3+F4+F2+F1	96.85%	3.15%	99.2%	0.8%	98.02%	
F3+F4+F2+F1+F7	97.68%	2.32%	99.44%	0.56%	98.56%	
F3+F4+F2+F1+F7 +F5	97.68%	2.32%	99.52%	0.48%	98.6%	
F3+F4+F2+F1+F7 +F5+F6	97.68%	2.32%	99.52%	0.48%	98.6%	-
F3+F4+F2+F1+F7 +F5+F8	97.62%	2.38%	99.52%	0.48%	98.57%	-

Best success rate for complex tree method is reached using Feature-1, Feature-2, Feature-3, Feature-4, Feature-5 and Feature-7.

- Linear Discriminant Analysis: 7 features (nD/nD: 67.84%, D/D: 99.15%, Average: 83.49%)

Table 5.2 Linear Discriminant Analysis feature selection

Features	nD/Nd	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	37.39%	62.61%	92.35%	7.65%	64.87%	•
Standard deviation of velocity (F2)	44.76%	55.24%	96.88%	3.12%	70.82%	-
Mean Acceleration (F3)	52.93%	47.07%	98.44%	1.56%	75.68%	•
Standard deviation of acceleration (F4)	52.33%	47.67%	99.8%	0.2%	76.06%	•
Mean RCS (F5)	55.63%	44.37%	91.52%	8.48%	73.57%	•
Standard deviation of RCS (F6)	0.08%	99.92%	100%	0%	50.04%	•
Mean Height (F7)	32.32%	67.68%	83.84%	16.16%	58.08%	•
Standard deviation of height (F8)	0.05%	99.95%	100%	0%	50.02%	•
F4+F3	59.09%	40.91%	98.9%	1.1%	78.99%	
F4+F3+F5	65.03%	34.97%	98.9%	1.1%	81.96%	
F4+F3+F5+F2	63.21%	36.79%	97.47%	2.53%	80.34%	-
F4+F3+F5+F1	67%	3%	97.56%	2.44%	82.28%	
F4+F3+F5+F1+F7	67.26%	32.74%	99.15%	0.85%	83.20%	
F4+F3+F5+F1+F7 +F6	67.84%	32.16%	99.13%	0.87%	83.48%	
F4+F3+F5+F1+F7 +F6+F8	67.84%	32.16%	99.15%	0.85%	83.49%	

Best success rate for linear discriminant analysis method is reached using Feature-1, Feature-3, Feature-4, Feature-5, Feature-6, Feature-7 and Feature-8.

- Logistic Regression Method: 8 features (nD/nD: 90.09%, D/D: 98.03%, Average: 94.06%)

Table 5.3 Logistic Regression method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	44.54%	55.46%	77.84%	22.16%	61.19%	•
Standard deviation of velocity (F2)	59.88%	40.12%	94.60%	5.40%	77.24%	•
Mean Acceleration (F3)	70.66%	29.34%	94.72%	5.28%	82.69%	•
Standard deviation of acceleration (F4)	71.32%	28.68%	96%	4%	83.66%	•
Mean RCS (F5)	55.72%	44.28%	91.77%	8.23%	73.74%	•
Standard deviation of RCS (F6)	0.06%	99.94%	99.95%	0.05%	50.01%	•
Mean Height (F7)	49.55%	50.45%	83.76%	16.24%	66.65%	•
Standard deviation of height (F8)	0.03%	99.97%	99.92%	0.08%	49.97%	•
F4+F3	74.22%	25.78%	95.65%	4.35%	84.93%	
F4+F3+F2	75.74%	24.26%	94.89%	5.11%	85.31%	
F4+F3+F2+F5	79.32%	20.68%	96.14%	3.86%	87.73%	
F4+F3+F2+F5+F7	80.52%	19.48%	97.37%	2.63%	88.94%	
F4+F3+F2+F5+F7 +F1	89.57%	10.43%	97.45%	2.55%	93.51%	
F4+F3+F2+F5+F7 +F1+F6	90%	10%	97.9%	2.1%	93.95%	
F4+F3+F2+F5+F7 +F1+F6+F8	90.09%	9.91%	98.03%	1.97%	94.06%	

Best success rate for logistic regression method is reached using Feature-1, Feature-2, Feature-3, Feature-4, Feature-5, Feature-6, Feature-7 and Feature-8.

- Linear SVM Method: 6 features (nD/nD: 75.94% , D/D: 99.8%, Average: 87.87%)

Table 5.4 Linear SVM method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	43.6%	56.4%	49.68%	50.32%	46.64%	•
Standard deviation of velocity (F2)	80.11%	19.89%	19.44%	80.56%	49.62%	-
Mean Acceleration (F3)	58.39%	29.34%	54.31%	45.61%	56.35%	•
Standard deviation of acceleration (F4)	77.26%	22.74%	20.68%	79.32%	48.97%	•
Mean RCS (F5)	32.18%	67.82%	53.42%	46.58%	42.8%	•
Standard deviation of RCS (F6)	60.15%	39.85%	45.6%	54.4%	52.87%	•
Mean Height (F7)	45.19%	54.81%	83.29%	16.71%	64.24%	•
Standard deviation of height (F8)	21.47%	78.53%	91.81%	8.19%	56.64%	-
F7+F8	42.23%	25.78%	84.57%	15.43%	63.4%	-
F7+F3	61.75%	38.25%	98.9%	1.1%	80.32%	
F7+F3+F6	61.78%	38.22%	98.9%	1.1%	80.34%	
F7+F3+F6+F2	62.52%	37.48%	98.09%	1.91%	80.3%	-
F7+F3+F6+F4	64.46%	35.54%	99.79%	0.21%	82.12%	
F7+F3+F6+F4+F1	67.37%	32.63%	99.75%	0.25%	83.56%	
F7+F3+F6+F4+F1 +F5	75.94%	24.06%	99.8%	0.2%	87.87%	

Best success rate for linear SVM method is reached using Feature-1, Feature-3, Feature-4, Feature-5, Feature-6 and Feature-7.

- Quadratic SVM Method: 5 features (nD/nD: 81.24%, D/D: 97.12%, Average: 89.18%)

Table 5.5 Quadratic SVM method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	21.48%	78.52%	99.88%	0.12%	60.68%	•
Standard deviation of velocity (F2)	60%	40%	95.28%	4.72%	77.64%	-
Mean Acceleration (F3)	69.04%	30.96%	96.12%	3.88%	82.58%	•
Standard deviation of acceleration (F4)	68.25%	31.75%	97.36%	2.64%	82.8%	•
Mean RCS (F5)	68.69%	31.31%	93.22%	6.78%	80.95%	•
Standard deviation of RCS (F6)	5.23%	94.77%	98.33%	1.67%	51.78%	-
Mean Height (F7)	51.17%	48.83%	81.64%	18.36%	66.4%	•
Standard deviation of height (F8)	2.22%	97.78%	91.69%	8.31%	46.95%	-
F4+F3	72.56%	27.44%	95.94%	4.06%	84.25%	
F4+F3+F5	78.71%	21.29%	96.68%	3.32%	87.69%	
F4+F3+F5+F2	75.61%	24.39%	95.50%	4.50%	85.55%	-
F4+F3+F5+F7	79.09%	20.91%	96.88%	3.12%	87.98%	
F4+F3+F5+F7+F1	81.24%	18.76%	97.12%	2.88%	89.18%	
F4+F3+F5+F7+F1 +F6	81.03%	18.97%	96.86%	3.14%	88.94%	-
F4+F3+F5+F7+F1 +F8	80.42%	19.58%	97.14%	2.86%	88.78%	-

Best success rate for quadratic SVM method is reached using Feature-1, Feature-3, Feature-4, Feature-5 and Feature-7.

- Cubic SVM Method: 8 features (nD/nD: 96.36%, D/D: 99.86%, Average: 98.11%)

Table 5.6 Cubic SVM method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	49.17%	50.83%	56.29%	43.71%	52.73%	•
Standard deviation of velocity (F2)	19.38%	80.62%	80.04%	19.96%	49.71%	•
Mean Acceleration (F3)	14%	86%	80.06%	19.94%	47.03%	•
Standard deviation of acceleration (F4)	18.54%	81.46%	76.9%	23.1%	44.72%	•
Mean RCS (F5)	31.71%	68.29%	53.42%	46.58%	42.57%	•
Standard deviation of RCS (F6)	50%	50%	49.4%	50.6%	49.7%	•
Mean Height (F7)	59.33%	40.67%	54.86%	45.14%	57.1%	•
Standard deviation of height (F8)	35.07%	64.93%	61.34%	38.66%	48.21%	•
F7+F1	85.78%	8.22%	32.19%	67.81%	58.99%	
F7+F1+F2	86.27%	29.73%	34.13%	65.87%	60.2%	
F7+F1+F2+F6	91.29%	8.71%	52.87%	47.13%	72.08%	
F7+F1+F2+F6+F8	91.44%	8.56%	74.79%	25.21%	83.11%	
F7+F1+F2+F6+F8 +F3	97.12%	2.88%	96.3%	3.7%	96.71%	
F7+F1+F2+F6+F8 +F3+F4	95.93%	4.07%	99.79%	0.21%	97.86%	
F7+F1+F2+F6+F8 +F3+F4+F5	96.36%	3.64%	99.86%	0.14%	98.11%	

Best success rate for cubic SVM method is reached using Feature-1, Feature-2, Feature-3, Feature-4, Feature-5, Feature-6, Feature-7 and Feature-8.

- Fine Gaussian SVM Method: 8 features (nD/nD: 97.39%, D/D: 100%, Average: 98.69%)

Table 5.7 Fine Gaussian SVM method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	60.92%	30.18%	92.64%	7.36%	76.78%	•
Standard deviation of velocity (F2)	73.38%	26.62%	85.37%	14.63%	79.38%	•
Mean Acceleration (F3)	74.41%	25.59%	86.59%	13.41%	80.5%	•
Standard deviation of acceleration (F4)	69.85%	30.15%	92.83%	7.17%	81.84%	•
Mean RCS (F5)	60.82%	39.18%	93.55%	6.45%	77.19%	•
Standard deviation of RCS (F6)	0.07%	99.93%	97.9%	2.1%	48.99%	•
Mean Height (F7)	72.33%	27.67%	84.02%	15.98%	78.17%	•
Standard deviation of height (F8)	17.65%	82.35%	94.22%	5.78%	55.93%	•
F4+F3	73.28%	26.72%	96.76%	3.24%	85.02%	
F4+F3+F2	74.80%	25.20%	96.95%	3.05%	85.87%	
F4+F3+F2+F7	88.97%	11.03%	97.83%	2.17%	93.4%	
F4+F3+F2+F7+F5	94.34%	5.66%	99.84%	0.16%	97.09%	
F4+F3+F2+F7+F5 +F1	97.23%	2.77%	99.93%	0.07%	98.58%	
F4+F3+F2+F7+F5 +F1+F8	97.29%	2.71%	99.95%	0.05%	98.62%	
F4+F3+F2+F7+F5 +F1+F8+F6	97.39%	2.61%	100%	0%	98.69%	

Best success rate for fine Gaussian SVM method is reached using Feature-1, Feature-2, Feature-3, Feature-4, Feature-5, Feature-6, Feature-7 and Feature-8.

- Fine K-NN Method: 6 features (nD/nD: 95.49%, D/D:99.97%, Average: 97.71%)

Table 5.8 Fine K-NN method feature selection

Features	nD/nD	nD/D	D/D	D/nD	Average	Usage
Mean Velocity (F1)	36.47%	63.53%	94.13%	5.87%	65.3%	•
Standard deviation of velocity (F2)	71.5%	28.5%	91.68%	8.32%	81.59%	•
Mean Acceleration (F3)	74.16%	25.84%	93.37%	6.63%	83.76%	•
Standard deviation of acceleration (F4)	70.76%	29.24%	96.31%	3.69%	83.53%	•
Mean RCS (F5)	57.11%	42.89%	93.76%	6.24%	75.43%	•
Standard deviation of RCS (F6)	29.27%	70.73%	89.15%	10.85%	59.21%	-
Mean Height (F7)	67.1%	32.9%	90.05%	9.95%	78.57%	•
Standard deviation of height (F8)	29.72%	70.28%	88.50%	11.50%	59.11%	-
F3+F4	74.4%	25.6%	95.88%	4.12%	85.14%	
F3+F4+F2	76.03%	23.97%	96.77%	3.23%	86.4%	
F3+F4+F2+F7	83.28%	16.72%	98.51%	1.49%	90.89%	
F3+F4+F2+F7+F5	87.69%	12.31%	99.9%	0.1%	93.79%	
F3+F4+F2+F7+F5 +F1	95.49%	4.51%	99.94%	0.06%	97.71%	
F3+F4+F2+F7+F5 +F1+F6	95%	5%	99.97%	0.03%	97.48%	-
F3+F4+F2+F7+F5 +F1+F8	94.78%	5.32%	99.97%	0.03%	97.37%	-

Best success rate for Fine K-NN method is reached using Feature-1, Feature-2, Feature-3, Feature-4, Feature-5 and Feature-7.

5.2. Performance Of UAV vs Non-UAV Classification

Success rates of appropriate classification methods is given in Table 5.9 while detailed procedure of selecting features is included in the previous chapter. Given in Equation 50, varying forms of Kernel function are included in the formulation of some methods in Table 5.9.

Table 5.9 Classification methods success rates with selected features

Method Name	Number of Features	Overall Success Rate
Complex Tree	6	98.60%
Linear Discriminant Analysis	7	83.49%
Logistic regression	8	94.06%
Linear SVM	6	87.87%
Quadratic Kernel SVM	5	89.18%
Cubic Kernel SVM	8	98.11%
Fine Gaussian Kernel SVM	8	98.69%
Fine K-NN	6	97.71%

Fine Gaussian SVM is selected for further usage with the selected eight features. Gaussian SVM is relatively more appropriate for limited processor capability than cubic kernel SVM classifier. In the Gaussian SVM method, Gaussian Kernel is used which is given below:

$$Gaussian\ kernel\text{: } K(\mathrm{x}, \mathrm{x}_i{}') = \exp(-\frac{1}{2}(\mathrm{x} - \mathrm{x}_i{}')^T \sum^{-1} (\mathrm{x} - \mathrm{x}_i{}')) \qquad (57)$$

If $\sum$ is diagonal, Equation 57 can be written as [26]:

$$K(\mathrm{x}, \mathrm{x}_i{}') = \exp(-\frac{1}{2}\sum_{j=1}^{D} \frac{1}{{\sigma_j}^2}(\mathrm{x}_j - \mathrm{x}_j{}')^2 \qquad (58)$$

In the feature selection of Gaussian SVM, as seen in Table 5.7, the best two individual success rates are 81.84% and 80.50% while performance using 8 features is 98.69%.

Fine Gaussian SVM is preferred since its advantages of being applied more easily in radar processors than cubic SVM although Gaussian SVM uses more features than Cubic SVM. Cubic SVM requires some computational load since it includes cubic terms in the formulation.

In this section, classification of performance kinematic features and RF characteristic features is observed. Classification success rate of mean and standard deviation of selected kinematic features is calculated. With a slight difference in varying environments, velocity, acceleration and height can be considered appropriate as kinematic features for the selected classification methods in this thesis.

5.3. Performance Improvement Method

Following the trials for the features and method selection of classification, success rate is aimed to be increased. Combination of various methods, using the same classification method multiple times and reduction of samples are some possible ways to improve the performance.

Liu suggests an algorithm that includes four SVM classifiers in series while each SVM classifier represents a level for the system [38]. As given in Figure, Liu designs a system with multiple classifiers in series and trains the data for SVM classifier in each level [38].

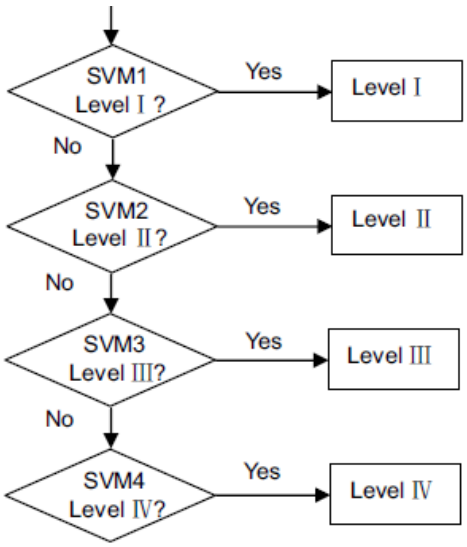

Figure 5.2 Usage of multiple SVM classifiers in series [38]

In another study, Liu et al. proposes an algorithm by nesting the methods for Multi-Class-SVM and cancelling all unclassifiable regions in the data set [39]. By comparing the success and complexity of the suggested algorithm with the One-Against-One and Fuzzy Least Squares Support Vector Machines approaches, Liu summarizes the algorithm in 4 steps [39]:

- Constructing hyperplanes in the feature space based on One-against-One approach,
- Choosing the samples in the unclassifiable region,
- Using these samples in the unclassifiable region to form hyperplanes with the same parameters,
- Repeating the previous two steps until the region is cancelled or no sample exists in that region.

Another approach for improving the performance of SVM is suggested by Lu as a sample reduction method which reduces the training samples through the Support Vector Domain Description algorithm and removes the edge points based on Euclidean Distance [40].

In the article by Yan et al., a strategy of combining Multiple SVM classifiers by using fuzzy integrals is presented. In this article, dividing the original training data set into n training sets by Bagging algorithm, SVM is applied to these training sets and outputs are combined by fuzzy integral as given in Figure 5.3 [41].

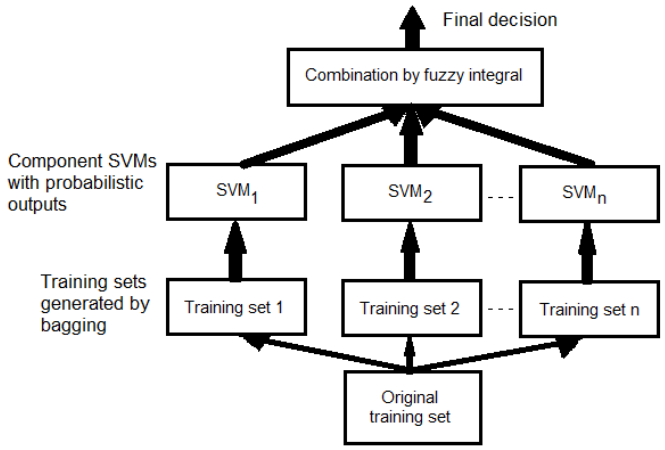

Figure 5.3 Combination of SVMs by fuzzy integral

After the trials of several classification methods with the selected features for the data obtained from the prepared setup, success rates in Table 5.9 are reached. Despite the fact that success rates are relatively high for the data used in Table 5.9, it is not guaranteed to have the same performance for varying clutter environments and drone models. In order to increase the performance which is given in Table 5.7 for the method of Gaussian SVM with eight selected features, not-drone prediction success rate is aimed to be improved since drone prediction performance is 100%.

The determined steps to increase the performance for Gaussian SVM classification method can be summarized as:

- Training the data obtained for not-drone and drone by Gaussian SVM method with eight features,
- Exporting the model of classification method used in training,
- Testing the same data by using the exported model,
- Determining the false predictions of not-drone tracks in the test results,
- Forming a new data vector by combining these samples with the drone data which is used in the training,
- Training the new data vector including the false predicted not-drone and drone samples for the second time,
- Exporting the model of classification method used in the second training.
- Testing the data with the first and second models which are obtained from the training models in order to reach the success rate.

  In order to assess success rates for the both drone and non-drone classes, the false predictions in the first stage are determined and examined for the second test stage.

  The reached performance improvement ratio with the second stage is 39.47%. This resulted ratio shows that the data classified in the first stage is better separated in the second stage with the improvement of 39.47% as given in Table 5.10.

Table 5.10 Second stage Fine Gaussian SVM success rates

Method	Number of Features	Non-Drone Success Rate	Non-Drone Error Rate	Improvement in Non-Drone Success Rate	UAV Success Rate
Gaussian Kernel SVM	8	98.69%	1.31%	%39.4	100%
2nd Stage Gaussian SVM	8	99.21%	0.79%		100%

CHAPTER 6

CONCLUSION

This thesis study shows that UAVs can be classified by SVM classification approach having the user-defined features. The conclusions obtained are given below.

- Classification methods are useful for separating UAVs from non-UAV targets using the distinctive characteristic and kinematic features.
- The selection of features from the wide feature set is dependent on the mechanical and electromagnetic properties of radar which presents target and not-drone information.
- Classification can possibly be based on track data or detection data of radar. Classifying upon track data is based on outputs of the radar tracking algorithm.
- After reviewing the literature, it is observed that many classification studies are based on analyzing kinematic features obtained from CW or Pulsed radars.
- It is observed that use of micro-Doppler signature is a very popular approach for classification problems provided that radar system is capable of using micro-Doppler signature.
- It is also observed from the experiments that classification success rate increases if data is trained and tested with more samples.
- Another observation from the experiments is that kinematic features and RF characteristic features which are used in this thesis result a significant success rate in the environment of experiments. Although features such as velocity, acceleration and height succeed separating drones from non-drones, success of these features may have a correlation with clutter of environment which is used in training.

- Experiments with classification methods that are used in this thesis showed that UAV and non-UAV tracks can be classified efficiently with the appropriate features. Experiments show that training with a target which is aimed to be classified is very effective on success rate (i.e. classification accuracy of UAV may reach 100% for UAV dependent case such that training and test set contains data from same drone). However, it may not be possible to train with all of the drone models in the market. According to the results of the experiments, training with varying types of kinematic motions increases the possibility of classifying the prospective data. It is also observed that classification of a drone with a satisfying success rate may be reached out by using training data obtained from some various types of drones.

Future Research Directions:

According to the results and observations from this thesis new future research directions are defined. The proposed future directions reported below would bring significant value to the UAV detection research if applied in the future.

- Extension of study with addition of some other clutter environments data and data from different kinds of UAVs can be examined in order to increase success rate for radar operations.
- The success rate for the selected method in this study can be examined for various types of moving targets.
- Inclusiveness of the features selected in this study can be investigated for different detection and tracking algorithms.
- The success of Gaussian SVM classifier with the determined features in this study can be investigated further for varying types of targets such as helicopters, land vehicles etc.
- Classification can be investigated for various feature selection methods.
- Effects of more varying and larger training sets can be investigated for the purpose of classification of mini/micro UAVs in the clutter presence.

REFERENCES

[1] Richards M.A., Scheer J.A., Holm W.A., Principles of Modern Radar Vol. I: Basic Principles, SciTech Publishing, 2010.

[2] Skolnik M. I., Introduction to Radar Systems, McGraw-Hill Book Company, 1981.

[3] IEEE Standard for Letter Desgnations for Radar-Frequency Bands, New York: The Institute of Electrical and Electronic Engineers, Inc., January 8, 2003.

[4] Curry C.R., Radar System Performance Modeling, Artech House, Inc., 2005.

[5] Richards M.A., Fundamentals of Radar Signal Processing, McGraw-Hill Book Company, 2005.

[6] EW and Radar Systems Engineering Handbook, Naval Air Warfare Center, Weapons Division., 1999.

[7] Korchenko A.G., "The Generalized Classification of Unmanned Air Vehicles" IEEE 2nd International Conference "Actual Problems of Unmanned Air Vehicles Developments" Proceedings, pp. 28-34, 2013.

[8] N. I. A. G. (NIAG), N. A. A. G. (NAAG) ve J. C. G. G. B. A. Defence, "Final Report of NIAG SG170 on Engagement of Low, Slow and Small Aerial Targets by GBAD", NATO, 2013.

[9] De Quevedo A.D., Urzaiz F.I., Menoyo J.G., Lopez A.A., "Drone Detection and RCS Measurements with Ubiquitous Radar", International Conference on Radar (RADAR), 2018.

[10] Ritchie M., Torvik B., “Micro-Drone RCS Analysis” IEEE Radar Conference, 2015.

[11] Sayler K., “A World of Proliferated Drones: A Technology Primer”, Center for a New American Security, CNAS, 2015.

[12] Duda R.O., Hart P.E., Stork D.G., Pattern Classification, Wiley Book Company, 2000.

[13] Sathya R., Abraham A., “Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification,” International Journal of Advanced Research in Artificial Intelligence (IJARAI), Vol. 2, No. 2, pp.34-38, 2013.

[14] Becker S., Plumbley M., “Unsupervised Neural Network Learning Procedures for Feature Extraction and Classfication,” Applied Intelligence 6, pp. 185-203, 1996.

[15] Everitt B.S., Cluster Analysis, John Wiley & Sons, Inc., New York, 1974.

[16] Jain A.K., Dubes C.R., Algorithms for Clustering Data, Prentice Hall Advanced Reference Series, 1988.

[17] Han J., Kamber M., Pei J., Data Mining Concepts and Techniques, The Morgan Kaufmann Publishers, 2012.

[18] Burney A.S.M., Tariq H., “K-Means Cluster Analysis for Image Segmentation”, International Journal of Computer Applications (IJCA), Vol:96, No:4, pp.1-8, 2014.

[19] Bandyopadhyay S., Sriparna S., Unsupervised Classification Similarity Measures, Classical and Metaheuristic Approaches, and Applications, Springer Science + Business Media, 2013.

[20] Jolliffe I.T., Principle Component Analysis, Springer Series in Statistics, 2002.

[21] Smith, L.I., "A tutorial on Principal Components Analysis", Computer Science Technical Report No. OUCS-2002-12, pp.21-22, 2002.

[22] Roweis S., "EM Algorithms for PCA and SPCA", Advances in Neural Information Processing Systems, no:10, pp.626–632, 1997.

[23] Hastie T., Tibshirani R., Friedman J., The Elements of Statistical Learning Data Mining, Inference, and Prediction, 2nd ed., Springer Series in Statistics, 2008.

[24] Shalev-Shwartz S., Ben-David S., Understanding Machine Learning From Theory to Algorithms, Cambridge Unversity Press, 2014.

[25] Osisanwo F.Y., Akinsola J.E.T., Awodele O., Hinmikaiye J.O., Olakanmi O., Akinjobi J., "Supervised Machine Learning Algorithms: Classification and Comparison," International Journal of Computer Trends and Technology (IJCTT), vol.:48, no:3, pp.128-138, 2017.

[26] Murphy K.P., Machine Learning A Probabilistic Perspective, The MIT Press, 2012.

[27] Haykin S., Neural Networks and Learning Machines, 3rd ed., Pearson Prentice Hall, 2009.

[28] James G., Witten D., Hastie T., Tibshirani R., An Introduction to Statistical Learning, Springer Series in Statistics, 2015.

[29] Alpaydin E., Introduction to Machine Learning, The MIT Press, 2004.

[30] Bishop C.M., Pattern Recognition and Machine Learning, Springer Science+Business Media, 2006.

[31] Bengio Y., "Learning Deep Architectures for AI," Technical Report 1312, Universite´ de Montreal, 2009.

[32] LISA Lab, University of Montreal, "Deep Learning Tutorial Release 0.1," , 2015.

[33] Jain A., Duin R.K., Jianchang P.W.M., "Statistical Pattern Recognition: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence vol.:20, no:1, pp.4-37, 2000.

[34] Slomka S., Gibbins D., Gray D., Haywood B., "Features for high resolution radar range profile based ship classification," 5th International Symposium on Signal Processing and its Applications, 1999.

[35] Lee D.R., La W.G., Kim H., "Drone Detection and Identification System using Artificial Intelligence," ICTC 2018, pp.1131-1133, 2018.

[36] Choi B., Oh D., "Classification of Drone Type Using Deep Convolutional Neural Networks Based on Micro-Doppler Simulation," 2018 International Symposium on Antennas and Propagation (ISAP), pp.1-2, 2018.

[37] Hafele F.X., "A New Algorithm for Automatic Radar Traget Classification Using Feature Extraction Having Special Regard to Drones," The International Radar Symposium IRS, 2019.

[38] Liu Z., Bai L., “Evaluating the Supplier Cooperative Design Ability Using a Novel Support Vector Machine Algorithm,” 2008 12th International Conference on Computer Supported Cooperative Work in Design, pp.986-989, 2008.

[39] Liu B., Hao Z., Yang X., “Nesting Support Vector Machinte For Muti-Classification,” Proceesings of the Fourth International Conference on Machine Learning and Cybernetics, pp.4220-4225, 2005.

[40] Lu S., Meng J., Cao G., “Support Vector Machine Based On A New Reduced Samples Method,” Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, pp. 1510-1514, 2010.

[41] Yan G., Ma G., Zhu L., Shi Z., “Combining Multiple Support Vector Machines Using Fuzzy Integral For Classification,” Proceedings of the Fifth International Conference on Machine Learning and Cybernetics, pp. 3438-3441, 2006.