Low Complexity Algorithms for Automatic Modulation Classification Based on Machine Learning

A Thesis

Presented in Partial Fulfilment of the Requirements for the

Degree of Master of Science

with a

Major in Electrical and Computer Engineering

in the

College of Graduate Studies

University of Idaho

by

Mohanad Abu-Romoh

Major Professor: Zouheir Rezki, Ph.D.

Committee Members: Lyudmyla Barannyk, Ph.D.; Dennis Sullivan, Ph.D.,P.E.

Department Administrator: Joseph Law, Ph.D.,P.E.

May 2019

Authorization to Submit Thesis

This thesis of Mohanad Abu-Romoh, submitted for the degree of Master of Science with a major in Electrical and Computer Engineering and titled "Low Complexity Algorithms for Automatic Modulation Classification Based on Machine Learning," has been reviewed in final form. Permission, as indicated by the signatures and dates given below, is now granted to submit final copies to the College of Graduate Studies for approval.

Major Professor: _____Date_____
Zouheir Rezki, Ph.D.

Committee Members: _____Date_____
Dennis Sullivan, Ph.D.,P.E.

_____Date_____
Lyudmyla Barannyk, Ph.D.

Department Administrator: _____Date_____
Joseph Law, Ph.D.,P.E.

Abstract

In this thesis, we discuss two different approaches to modulation classifiers: we first propose a hybrid method for automatic modulation classification that lies in the intersection between likelihood-based and feature-based classifiers. Specifically, the proposed method relies on statistical moments along with a maximum likelihood engine. We show that the proposed method offers a good trade-off between classification accuracy and complexity relative to the Maximum Likelihood (ML) classifier. Furthermore, our classifier outperforms state-of-the-art machine learning classifiers, such as genetic programming-based K-nearest neighbor (GP-KNN) classifiers, the linear support vector machine classifier (LSVM) and the fold-based Kolmogorov-Smirnov (FB-KS) algorithm. In the second part of thesis, we propose a distribution-based modulation classifier using neural networks. We show that our proposed classifier outperform state-of-the-art classifiers, even when the pool of possible candidate modulations are unknown to the receiver.

Acknowledgements

I would like to express my sincerest gratitude to my advisor Dr. Zouheir Rezki for supporting my research, for his motivation, patience, and immense knowledge. His guidance helped me through all the steps that lead to succeeding in research and the writing of this thesis.

I would also like to thank the rest of my thesis committee: Prof. Dennis Sullivan, and Dr. Lyudmyla Barannyk, for their insightful comments and encouragement.

I thank my fellow labmates. Without their precious support and guidance, all the discussions we had and their help in simulations, it would not be possible to conduct this research. Also I thank my friends who supported me through the time of my enrollment in my beloved university, University of Idaho.

Last but not the least, I would like to thank my family for supporting me spiritually during my academic career, and their immense support that led to the writing of this thesis.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1 Background

Automatic modulation classification (AMC) is the intermediate step between signal detection and demodulation. The term "Automatic" is used to indicate that the classification process is done automatically, as opposed to earlier modulation classifiers where signals are processed manulally with the aid of signal observation and processing equipment [1]. The problem of AMC was motivated by the its numerous applications in both civilian and military fields. In civilian applications, we can find AMC used in rate adaptation, where the transmitter exploits the channel conditions by choosing a modulation scheme that provides more throughput (e.g. QAM-64) for strong channels, and more reliability (e.g. QPSK or BPSK) for weak channels. In military applications, AMC can be used to identify and correctly modulate alien signals, as well as in threat analysis. AMC has attracted researchers' attention over the last few decades. The nature of the problem

1.2 Applications

Nowadays, communication systems rely heavily on higher order modulations to achieve high data rate transmission. In wired transmission, the communication channel is relatively stable, and the classifier does not need to adapt to the changes in the communication channel. As a result, the channel estimation is done once at the system initialization. In wireless transmission, the channel undergoes different types of fading at different time instants, which promotes the need to continuously estimate the channel state information (CSI) in order to accurately classify the modulation of the signal. In Civilian applications, it is common to see AMC used in Link adaptation, which is the process of adaptively selecting the type of modulation based on the acquired channel

state information. Communication system can benefit from Link adaptation by choosing a higher-order modulation for strong channels (AWGN or slow-fading channels), and a lower-order modulation for weak channels (fast-fading channels), to increase the overall communication capacity.

In military applications, AMC is essential in many different strategies. AMC can be implemented to correctly detect and demodulate alien signals and to recover the transmitted signal. This can be further exploited to properly transmit a jamming signal, which are high power signals to override adversary communications in the same frequency band, and prevent the communication between units.

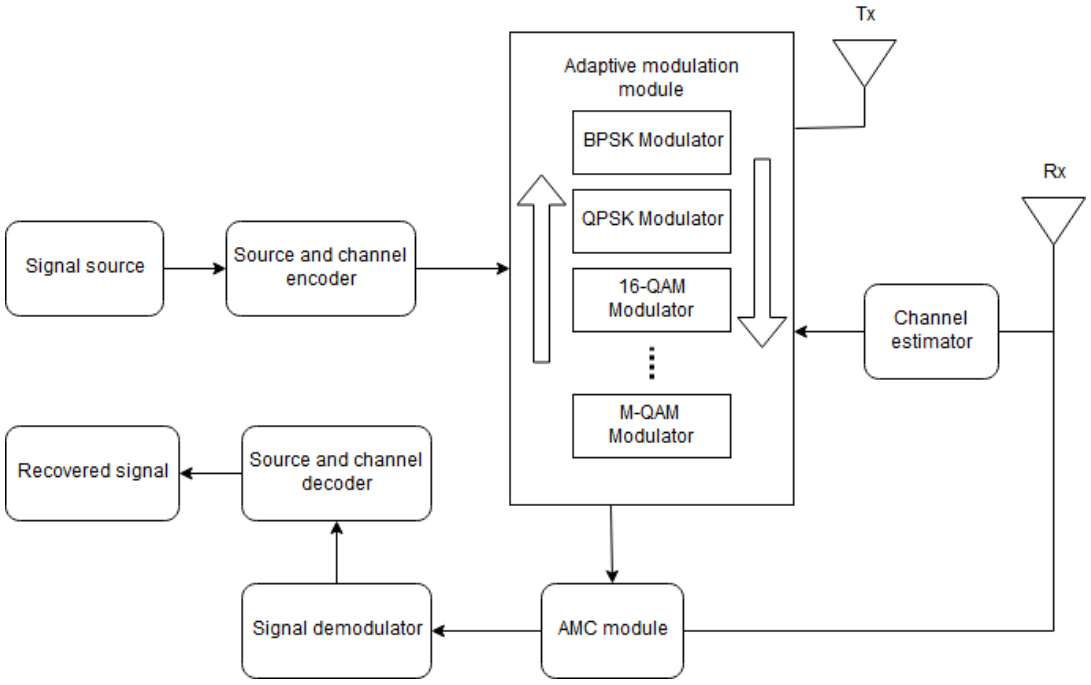


Figure 1.1: AMC in Link Adaptation system

1.3 Likelihood-based Classifiers

Likelihood-based classifiers are the most popular amongst modulation classifiers. LB classifiers are by far the most optimal in terms of classification accuracy when the channel model and channel parameters are perfectly known to the receiver. LB classifiers work

first by defining a likelihood rule and set of hypothesis (H_i : modulation ω_i is sent), then maximizing the likelihood rule over all the set of hypothesis. despite their optimality, LB classifiers suffer from two critical drawbacks: their computational complexity, and the reliance on knowing the model of the signal as well as the channel estimation at the receiver (CSI-R). Many approaches had been proposed to lower the computational complexity of LB classifiers, as well as the adaptability in non-cooperative channels. The most common LB Approach is the Maximum Likelihood classifier, which is an upper bound of the performance of modulation classifiers in terms of classification accuracy. Some of alternative approaches are the average likelihood ratio test (ALRT), generalized likelihood ratio test (GLRT) and hybrid likelihood ratio test (HLRT). In this chapter, we will mainly focus on the Maximum Likelihood approach.

1.3.1 Maximum Likelihood classifiers

Likelihood evaluation is equivalent to the calculation of probabilities of observed signal samples belonging to the models with given parameters. In a maximum likelihood classifier, with perfect channel knowledge, all parameters are known except the signal modulation. Therefore, the classification process can also be perceived as a maximum likelihood estimation of the modulation type where the modulation type is found in a finite set of candidates [2]. The main focus will be to derive the likelihood function mainly in the AWGN channel.

We start by defining a set of Hypothesis H_i such that,

$$H_i : \text{ observed signal sample } y(n) \text{ belonging to the modulation } \omega_i.$$

Given that the likelihood of the each hypothesis is equal to the probability of the signal sample $y(n)$ being observed in the AWGN channel modulated with M, then the following equation holds.

$$\mathcal{L}(H_j|\boldsymbol{Y},\sigma) = p(\boldsymbol{Y}|H_j,\sigma) \tag{1.1}$$

$$\mathcal{L}(H_j|\boldsymbol{Y},h) = \prod_{i=1}^{N}\sum_{k=1}^{M_j}\frac{1}{M_j}\frac{1}{\pi\sigma^2}\exp\bigg(-\frac{\big|Y(i)-S_{jk}\big|^2}{\sigma^2}\bigg), \tag{1.2}$$

where (1.2) is derived using the complex form PDF of received signal in AWGN channel. $\boldsymbol{Y} = [Y(1),\ldots,Y(N)]$ is the received vector, M_j is the cardinality of modulation ω_j, and S_{jk} is k-th point of ω_j, for $k = 1,\ldots,M_j$, and $j = 1,\ldots,K$. The latter equation can sometimes written in the form of a log-likelihood for analytical convenience, resulting in the the following expression:

$$\begin{aligned}\log\mathcal{L}(H_j|\boldsymbol{Y},h) &= \log\bigg(\prod_{i=1}^{N}\sum_{k=1}^{M_j}\frac{1}{M_j}\frac{1}{\pi\sigma^2}\exp\bigg(-\frac{\big|Y(i)-S_{jk}\big|^2}{\sigma^2}\bigg)\bigg)\\ &= \sum_{i=1}^{N}\log\bigg(\frac{1}{M_j}\frac{1}{\pi\sigma^2}\sum_{k=1}^{M_j}\exp\bigg(-\frac{\big|Y(i)-S_{jk}\big|^2}{\sigma^2}\bigg)\bigg).\end{aligned} \tag{1.3}$$

The likelihood function can also be derived from PDF of different aspects of received signals. As an example, we can first derive the PDF for the real part of the received signal in AWGN channel, and then construct the likelihood function shown in the equation:

$$\mathcal{L}(H_j|\boldsymbol{Y},h) = \prod_{i=1}^{N}\sum_{k=1}^{M_j}\frac{1}{M_j}\frac{1}{\sqrt{\pi\sigma^2}}\exp\bigg(-\frac{\big|\Re\{Y(i)\}-\Re\{S_{jk}\}\big|^2}{\sigma^2}\bigg). \tag{1.4}$$

1.4 Feature-based Classifiers

Feature-based classifiers are the most common approaches of designing modulation classifiers. Instead of dealing with the entire sequence of the received signal, the classifier can reduce the computational demand by using a smaller set of key features. The choice of the features used in classification should be adapted to the type of the problem. For

example, if the pool of possible modulations used by the transmitter contains phase modulation, then using a feature that is a function of phase is desirable. One example of such feature is standard deviation of the absolute phase σ_{ap}, defined as follow:

$$\sigma_{ap} = \sqrt{\frac{1}{N}\left(\sum_{i=1}^{N} \phi_{NL}(i)^2\right) - \left(\frac{1}{N}\sum_{i=1}^{N} |\phi_{NL}(i)|\right)^2}, \tag{1.5}$$

where ϕ_{NL} denotes the non-linear part of the phase for the i-th symbol.

1.5 Machine Learning-based Classifiers

One of the drawbacks of LB and FB classifiers is the need of prior knowledge of the pool of possible modulations used by the transmitter, as well as their statistical characteristics. This prior knowledge may not be always available. As an example, in military applications, where the classifier's task is to classify the modulation of different alien signals, the classifier can not acquire enough information about the possible modulations used by the transmitter, rendering the classifier unable to use the LB or FB approach. One other major drawback of FB classifiers is the need to design the decision trees with multiple thresholds.

In order to overcome these issues, research has been focused toward designing various machine learning algorithms. First, The machine learning engine can learn the structure of the testing data set without the need of an exact model. Second, the machine learning engine can can reduce the dimension of feature set, and provide decision trees that are simpler and easier to implement.

In this section we are going to introduce some of the most widely-used machine learning algorithms, namely, the K-nearest neighbor (KNN), linear support vector machines (LSVM) and genetic programming (GP).

1.5.1 K-nearest Neighbor classifiers

The K-nearest neighbor classifier had been widely used to solve many different classification problems. KNN works by assigning a class to a testing sample if the majority of the K nearest reference samples also belong to that class. In modulation classification problem, KNN can be used after converting the reference signals into a set of features and establishing the feature space.

The classifier evaluates the distances between the testing signal and the reference signals. The most common distance metric is the Euclidean distance. Assume a given feature set $\mathbf{F} = \{F_1, F_2, \ldots, F_L\}$, where L is the number of features, the Euclidean distance between two feature sets, $\mathbf{F}^A$ and $\mathbf{F}^B$ can be calculated as:

$$D(\mathbf{F}^A, \mathbf{F}^B) = \sqrt{\sum_{i=1}^{L} [F_i^A - F_i^B]^2}. \tag{1.6}$$

After the distances between the test signal and the all the reference signals are evaluated and recorded, the classifier proceeds with sorting them and selects the minimum K reference signals with the minimum distances.

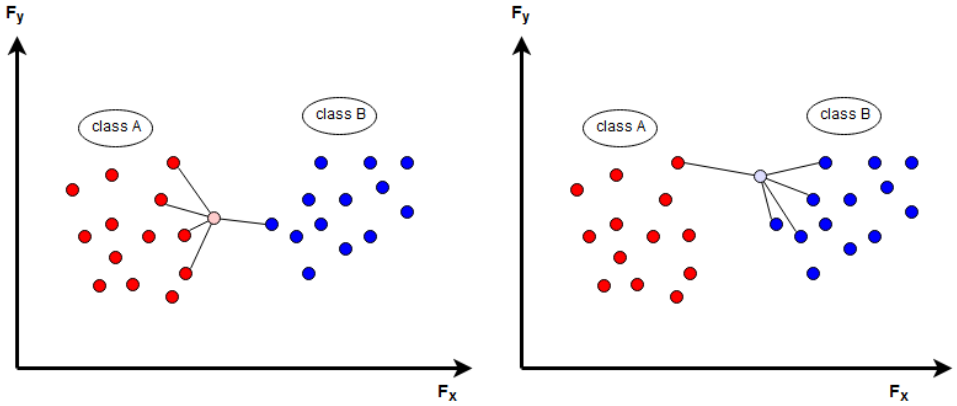


Figure 1.2: Two-class feature space K-nearest neighbor, $K = 5$.

The selection of K should follow a number of simple rules. Ideally, K should not be

divisible by the number of classes, and should be large enough to avoid false-classification caused by outliers. Moreover, the value of K should be less than the total number of reference signals available. The K reference samples with minimum distances are the basis of the classification, as the classifier decides in favor of the class with majority of samples belonging to the set of K-nearest neighbors.

1.5.2 Linear Support Vector Machine Classifiers

Linear Support Vector Machine is another classification algorithm that uses the multi-dimensional feature space. It works by finding a hyperplane that separates the reference samples into two different classes on each side of the hyperplane. The hyperplane is optimized by maximizing the distance to the signal samples of each class. LSVM have a linear kernel that is defined as follows

$$K(\boldsymbol{F}, \boldsymbol{w}) = \boldsymbol{F}^T \boldsymbol{w}, \tag{1.7}$$

where $\boldsymbol{F} = \{F_1, F_2, \ldots, F_n\}$ is the vector with input features $F_1, \ldots, F_n$, and w is the weight vector that needs to be optimized. The decision rule in LSVM classifiers can be formulated as follows

$$\hat{H}_i = \begin{cases} H_1, & g(F) = F^T w + w_0 \geq 0 \\ H_2, & g(F) = F^T w + w_0 < 0 \end{cases} \tag{1.8}$$

The hyperplane $g(F) = F^T w + w_0 = 0$ is found using an iterative optimization process, which is exercised to solve the following problem:

$$\begin{aligned} &\text{maximize} \quad D(w, w_0) = \frac{2}{||w||}^2 \\ &\text{subject to} \quad y_i(w^T x_i + w_0) \geq 1, \quad i = 1, 2, \ldots, L, \end{aligned} \tag{1.9}$$

where y_i is a class indicator for the i-th feature, such that, $y_i = 1$ indicates that

sample i belongs to class A, and $y_i = -1$ indicates that sample i belongs to class B.

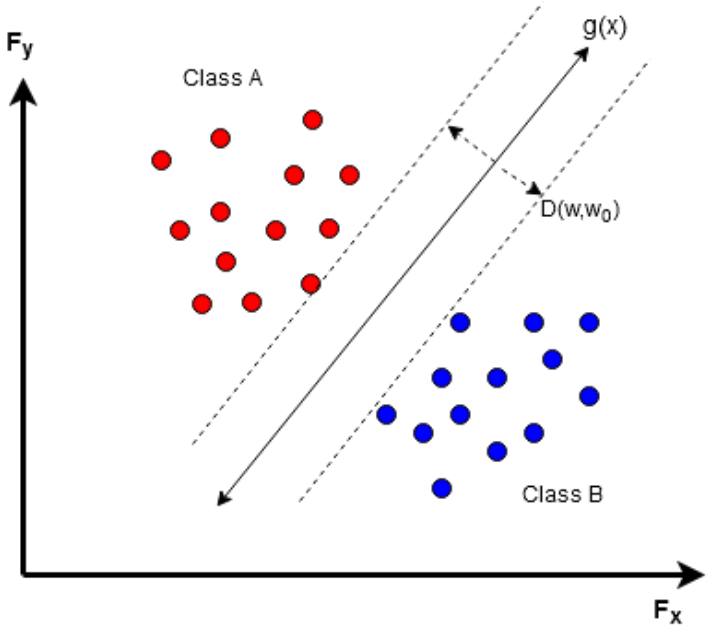


Figure 1.3: Two-class feature space with linear support vector machine.

Compared with the KNN classifier, the SVM classifier only needs to use the training signal when establishing the separating hyperplane. Once the hyperplane is optimized, there is no need to involve the training signal in any sort of further calculation. The benefit is that the computation needed at the testing stage is relatively inexpensive compared with KNN. However, the SVM classifier is most natural for two-class classification. There are implementations of a multi-class classification using SVM. However, the implementation is much less intuitive than in the two-class case.

CHAPTER 2

MOMENTS-BASED CLASSIFICATION

2.1 Motivation

Most modulation classification algorithms suffer from a clear trade-off between computational complexity and classification performance. In ML classification, which provides the upper limit in terms of classification performance, the computational complexity is significantly demanding. On the other hand, FB-based classifiers are less computationally demanding, but they fall behind in term of classification accuracy. This motivated us to look for a classification algorithm that combines between the simplicity of FB-based classifiers and the accuracy of ML classifiers. Motivated by the fact that moments of the received signal are easy to compute and can provide a simple way to automatically classify the modulation of the transmitted signal, we propose a hybrid method for automatic modulation classification that lies in the intersection between likelihood-based and feature-based classifiers. The proposed classifier uses statistical moments via an ML engine and provides a good accuracy-complexity trade-off. It is worth noting that moments were chose over other statistical features due to a number of reasons: calculating moments is relatively simple, which in turn lowers the complexity of the system. Moments also offer robustness to AWGN noise by averaging the signal over its symbols.

In this study, our main contribution is to propose an algorithm that obtains the optimal accuracy from a set of moments while maintaining a significantly lower computational complexity than the ML classifier and a competitive complexity with the state-of-the-art. Note that we do not claim optimality in the choice of moments used for classification but we claim optimality in combining a given set of moments, where optimality is in terms of probability of correct classification.

This chapter is divided into four sections. In the system model section, we discuss the channel effects on the received signal and its interaction with the signal model. In

the next section, we present our proposed algorithm, and derive the moment-based likelihood maximization classifier. In the results section, We present a comparison between our method and state-of-the-art methods. We also show that the proposed method offers a good trade-off between classification accuracy and complexity relative to the ML classifier. Finally, we show that our classifier outperforms state-of-the-art machine learning classifiers, such as genetic programming-based K-nearest neighbor (GP-KNN) classifiers, the linear support vector machine classifier (LSVM) and the fold-based Kolmogorov-Smirnov (FB-KS) algorithm.

2.2 System Model

Regardless of the transmitter setting and modulation selection, the transmitted signals are subject to the same channel conditions. Here we give a signal model that includes a majority of the channel effect a single wireless radio frequency may encounter. The received signal is given by:

$$Y(i) = Ae^{j(2\pi f_o iT+\theta_i)} \sum_{l=-\infty}^{\infty} X(l)p(lT - iT)h(iT - lT + \epsilon_T T) + Z(i), \tag{2.1}$$

where A is the amplitude of the signal, f_o is carrier frequency offset, T is the symbol duration, $h(\cdot)$ is the baseband channel effect, $p(\cdot)$ is the pulse shaping, θ_i is the phase jitter varying from sample to sample, $X(l)$ is the symbol sequence, ϵ_T is the timing error, and $g(i)$ is the additive white gaussian noise (AWGN).

Under common circumstances, assumptions are made that the pulse shaping and channel response is known to the receiver. Therefore, the signal model after matching the filter could be simplified as:

$$Y(i) = Ae^{j(2\pi f_o iT+\theta_i)} X(i) + Z(i), \tag{2.2}$$

equation 2.2 can be further simplified such that the baseband waveform at the receiver can be modeled as:

$$Y(i) = hX(i) + Z(i), \tag{2.3}$$

We assume a quasi-static channel model where h is constant during T and changes randomly during the next coherence time T. Furthermore, we assume perfect channel state information at the receiver (CSI-R) as in [3–6]. We assume that, at the beginning of classification, the receiver has acquired CSI via training or via the sequence itself.

2.3 Proposed Method

Our proposed approach is based on maximizing the likelihood that a modulation has been used at the transmitter, but using a finite number of moments only, instead of a received sequence of symbols of length N, $\boldsymbol{Y}$. We evaluate the likelihood as a function of the moments of the received sequence. Let $\tilde{\mathcal{M}}$ be the set of empirical moments of Y such as:

$$\tilde{\mathcal{M}} = \{\tilde{M}_{p_1q_1}, \tilde{M}_{p_2q_2}, \ldots, \tilde{M}_{p_Lq_L}\}, \tag{2.4}$$

where L is the cardinality of $\tilde{\mathcal{M}}$. $\tilde{M}_{p_iq_i}(\boldsymbol{Y})$ is the empirical moment used as an approximation of the exact moment $M_{p_iq_i}(\boldsymbol{Y})$. $\tilde{M}_{pq}$ can be obtained by:

$$\tilde{M}_{pq} \triangleq \frac{1}{N}\sum_{i=1}^{N} Y(i)^{p-q}Y^*(i)^q. \tag{2.5}$$

Note that the empirical moment $\tilde{M}_{pq}$ converges to the statistical moment M_{pq} for large N due to the law of large numbers. Once the moments $\tilde{M}_{p_i,q_i}$ are evaluated, the ML engine proceeds by formulating the hypothesis testing problem as follows: H_j : Modulation ω_j is used, for $j = 1, \ldots, K$. Since the received symbols $Y(i)$ in (2.5) are independent and identically distributed (i.i.d.), then, conditioned on the channel h, the central limit theorem can be invoked to show that the probability density function (pdf)

of each conditional moment $f_{\tilde{M}_{pq}|H_j}(m|H_j)$ converges to a Gaussian distribution as the number of samples N increases, i.e.,

$$\lim_{N\to\infty} \Pr\left(\tilde{M}_{pq} \geq x|H_j\right) = Q\left(\frac{x-\mu_{pq}^{(j)}}{\sigma_{pq}^{(j)}}\right), \tag{2.6}$$

where $Q(\cdot)$ is the Q-function, and $\mu_{pq}^{(j)}$ and $\sigma_{pq}^{(j)}$ are the mean and the standard deviation of the empirical moment $\tilde{M}_{pq}$ given the hypothesis H_j. The use of the CLT is justified since a typical sequence length N is larger than or equal to 128. The summands in (2.5) are i.i.d.; hence, the mean of $\tilde{M}_{pq}$ can be obtained as:

$$\begin{aligned}
\mu_{pq}^{(j)} &\triangleq E[\tilde{M}_{pq}|H_j, h] \\
&= E\Big[\frac{1}{N}\sum_{i=1}^{N} Y(i)^{p-q}Y^*(i)^q \Big| H_j, h\Big] &(2.7)\\
&= E\Big[Y^{p-q}Y^{*q}\Big| H_j, h\Big] \quad \text{for } j = 1,\ldots,K, &(2.8)
\end{aligned}$$

where (2.7) follows from the definition of empirical moments in (2.5), $Y \triangleq Y(i)$ and (3.10) follows from that $Y(i), i = 1,\ldots,N$ are i.i.d. since the channel is memoryless (c.f. subsection 3.2). To completely define the pdf of $\tilde{\mathcal{M}}|H_j$, we also need its covariance matrix $\mathbf{K}^{(j)}$ of size $L \times L$. The entries $K_{mn}^{(j)}$ of the covariance matrix $\mathbf{K}^{(j)}$ can therefore be obtained by:

$$K_{mn}^{(j)} = E[(\tilde{M}_{p_m q_m} - \mu_{p_m q_m}^{(j)})(\tilde{M}_{p_n q_n} - \mu_{p_n q_n}^{(j)})^*|H_j, h]. \tag{2.9}$$

computing the covariance matrix in 2.9 can be further simplified as shown in (appendix A).

The joint conditional pdf $f_{\tilde{\mathcal{M}}|H_j}(\tilde{\mathcal{M}}|H_j)$ of moments (with each moment itself being normally distributed) can simply be obtained by the general equation of the multivariate

complex Gaussian distribution:

$$f_{\tilde{\mathcal{M}}|H_j}(\tilde{\mathcal{M}}|H_j) = \frac{1}{\pi^L \left|\mathbf{K}^{(j)}\right|} \exp\Big\{ -(\tilde{\mathcal{M}} - \boldsymbol{\mu}^{(j)})(\mathbf{K}^{(j)})^{-1}(\tilde{\mathcal{M}} - \boldsymbol{\mu}^{(j)})^H \Big\}, \tag{2.10}$$

where $\left|\mathbf{K}^{(j)}\right|$ is the determinant of $\mathbf{K}^{(j)}$, and $\boldsymbol{\mu}^{(j)} = [\mu^{(j)}_{p_1 q_1}, \ldots, \mu^{(j)}_{p_L q_L}]$. We know from the Bayesian rule that the likelihood function can be written as:

$$L(H_j|\tilde{\mathcal{M}}) = f(H_j|\tilde{\mathcal{M}}) = f(\tilde{\mathcal{M}}|H_j)\frac{f(H_j)}{f(\tilde{\mathcal{M}})}. \tag{2.11}$$

We are interested in maximizing the probability of correct classification. This can be achieved by maximizing the likelihood function using the pdf function derived in (2.10). The classifier decides that hypothesis H_j is correct if the likelihood function $L(H_j|\tilde{\mathcal{M}})$ is maximized for all $j = 1, \ldots, K$. Assuming that all the candidate modulations are equally probable, maximizing $L(H_j|\tilde{\mathcal{M}})$ boils down to maximizing $f(\tilde{\mathcal{M}}|H_j)$ over the set of all hypotheses.

2.3.1 Numerical Results and performance analysis

In this section, we present a variety of simulation experiments to demonstrate the performance of our proposed algorithm. For our method, we use five moments, namely:

$$\tilde{\mathcal{M}} = \{\tilde{M}_{40}, \tilde{M}_{42}, \tilde{M}_{61}, \tilde{M}_{63}, \tilde{M}_{82}\}. \tag{2.12}$$

The SNR γ in dB is defined as $\gamma = 10\log_{10}(E_s/\sigma^2)$, where E_s is the signal energy. We classify between five different modulations given in: $\Omega = \{\text{BPSK}, \text{QPSK}, \text{8PSK}, \text{16QAM}, \text{64QAM}\}$. The selection of these modulations is mainly because they are included in the IEEE 802.11a standard. All results are based on 10,000 Monte Carlo trials for each modulation, i.e., 50,000 trials for modulation set Ω. We express the average probability of correct

Table 2.1: Comparison of moments for higher order modulations.

	64QAM	256QAM	Absolute Value of Relative Difference (%)
$M_{4,0}$	-0.619	-0.6047	2.3%
$M_{4,2}$	1.3810	1.3953	1%
$M_{6,1}$	-1.2980	-1.2890	0.69%
$M_{8,4}$	4.1910	3.9630	5.4%
$M_{12,6}$	17.3775	15.2692	12.3%
$M_{16,8}$	68.9592	86.7237	25.8%

classification, P_c, as the average of individual probabilities of correct classification, i.e.,

$$P_c = \frac{1}{K}\sum_{m=1}^{K} \Pr\left(\underset{i=1,\ldots,K}{\operatorname{argmax}} L(H_j|\tilde{\mathcal{M}}) = m|H_m \right). \tag{2.13}$$

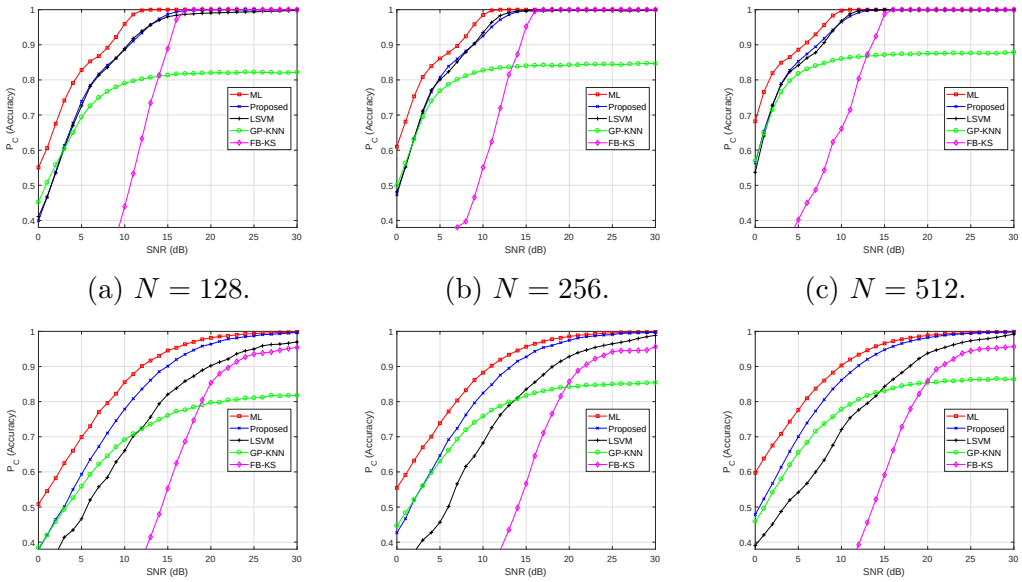
We simulate P_c to measure the performance of five classifiers: the ML classifier, the proposed classifier, the well-known LSVM classifier [7], the GP-KNN classifier [8], and the fold-based Kolmogorov-Smirnov (FB-KS) classifier in [3].

Note that cumulants are polynomial functions of moments [5]; hence, while the CLT holds for moments, it does not hold in general for cumulants since polynomials of Gaussian random variables are not necessarily Gaussian. In addition, we note that a set of moments of $\boldsymbol{Y}$ and a set of cumulants of $\boldsymbol{Y}$ of the same size and order carry the same amount of information about the modulation to be classified. To see this, let H be a random variable whose support is the set of modulations Ω and $SS(\boldsymbol{Y})$ denote the sufficient statistic of $\boldsymbol{Y}$. Consider the mutual information between a sequence of moments $\{M_{p_1q_1}, \ldots, M_{p_Lq_L}\}$ and the modulation hypothesis H, denoted by $I(H; M_{p_1q_1}, \ldots, M_{p_L,q_L})$, and that of a sequence of cumulants $\{C_{p_1q_1}, \ldots, C_{p_Lq_L}\}$ and H, where $p_i = p_{i-1} + 1$ for $i = 2, \ldots, L$, and $p_1 = 1$. Then,

$$I(H; \boldsymbol{Y}) = I(H; SS(\boldsymbol{Y})) \tag{2.14}$$

$$= I(H; M_{p_1q_1}, \ldots, M_{p_Lq_L}) \tag{2.15}$$

$$= I(H; f(M_{p_1q_1}, \ldots, M_{p_Lq_L})) \tag{2.16}$$



(a) $N = 128$. (b) $N = 256$. (c) $N = 512$.

(d) $N = 128$. (e) $N = 256$. (f) $N = 512$.

Figure 2.1: Comparison between the probability of correct classification versus SNR (dB) of five different classification methods under different channel assumptions: AWGN (a)–(c), and normalized Rayleigh fading (d)–(f), for various sequence lengths. The modulations used are BPSK, QPSK, 8PSK, 16QAM and 64QAM.

$$= I(H; C_{p_1 q_1}, \ldots, C_{p_L q_L}), \tag{2.17}$$

where (a) follows because sufficient statistics preserve mutual information; (b) follows from that $\{M_{p_1 q_1}, \ldots, M_{p_L q_L}\}$ are in general not a sufficient statistic, with equality if and only if $\{M_{p_1 q_1}, \ldots, M_{p_L q_L}\}$ are a sufficient statistic of $\boldsymbol{Y}$; (c) follows if and only if $f : \mathbb{R}^L \to \mathbb{R}^L$ is a one-to-one mapping; and (d) follows due to the fact that cumulants are indeed one-to-one functions of moments since there is a one-to-one relation between the moment-generating function $M(t) = E[e^{tY}]$ and the cumulant-generating function $K(t) = \log E[e^{tY}]$. Hence, a sequence of moments and a sequence of cumulants of the same order and size carry the same information about H.

Figure 2.1 shows the accuracy of correct classification for different sequence lengths in the AWGN channel and the normalized Rayleigh fading channel. First, as shown by Figures 2.1a–2.1c, in the AWGN channel, at SNRs higher than or equal to 3 dB,

our proposed classifier performs similarly to the LSVM classifier and outperforms the GP-KNN and FB-KS classifiers by considerable margins. But, for a sequence length of only 128 modulated symbols (in Fig.2.1a), our proposed classifier achieves a classification accuracy of 100% at an SNR of 18 dB, while the LSVM classifier achieves a 100% accuracy at 30 dB. Further, when normalized Rayleigh fading is considered, our proposed method outperforms the LSVM, the GP-KNN, and the FB-KS classifiers over all SNR values, as shown by Figures 2.1d–2.1f. For a sequence of 128 symbols (in Fig.2.1(d)), our proposed classifier achieves a classification accuracy of 90% at 15 dB, while the LSVM classifier achieves the same accuracy at 20 dB.

The reason our algorithm outperforms the LSVM is twofold. First, when two families of modulations, namely QAM-type and PSK-type, need to be classified, there is no clear linear hyperplane that would separate these two classes in the feature space. Second, the performance of the LSVM classifier is sensitive to mismatch between the SNRs of the training and testing data. Thus, when fading is considered, the difference between the training and testing SNRs increases, thus deteriorating the accuracies of the LSVM classifier, as illustrated by Figures 3.3(d)–3.3(f). On the other hand, our proposed classifier uses moments and the likelihood function in (2.11), whose computation does not require any training. Hence, our proposed method still performs well and classifies with higher accuracies than the LSVM, the GP-KNN, and the FB-KS classifiers in the presence of fading.

2.3.2 Complexity analysis

The number of different operations required by different classifiers is listed in Table 2.2. The implementation of the ML classifier requires exponential and logarithmic operations of order N, unlike our proposed, the LSVM, the GP-KNN, and the FB-KS classifiers. Furthermore, the proposed, the LSVM, the GP-KNN, and the FB-KS classifiers all offer lower computational complexities in terms of numbers of additions and multiplications

Table 2.2: Complexity comparison between the ML, Proposed, LSVM, GP-KNN, and FB-KS classifiers. Recall that p_i is the order of the moment $M_{p_i q_i}, i = 1, \ldots, L$. Note that R represents the number of training samples, the approximate equality $\approx$ denotes the order of an expression, and the FB-KS classifier also requires $2N \sum_{j=1}^{K} \sqrt{M_j}$ computations of the error function $erf(\cdot)$.

	Multiplications	Additions	Exponentials	Logarithms	Comparisons
ML	$NK + 3NK \sum_{j=1}^{K} M_j \approx 10^6$	$3NK \sum_{j=1}^{K} M_j \approx 10^6$	$NK \sum_{j=1}^{K} M_j \approx 10^5$	$NK \approx 10^3$	$K - 1 = 4$
Proposed	$3N \sum_{i=1}^{L} (p_i - 1) \approx 10^4$	$N \sum_{i=1}^{L} (p_i + 1) \approx 10^4$	0	0	$K - 1 = 4$
LSVM	$(K-1)N/2 \approx 10^3$	$(K-1)N/2 \approx 10^3$	0	0	$N \log N \approx 10^4$
GP-KNN	$3N \sum_{i=1}^{L} (p_i - 1) \approx 10^4$	$N \sum_{i=1}^{L} (p_i + 1) + RK \approx 10^4$	0	0	$RK \log(RK) \approx 10^3$
FB-KS	0	$8N \sum_{j=1}^{K} \sqrt{M_j} \approx 10^5$	0	0	$10N \approx 10^4$

required than the ML classifier. The number of multiplications required by the five algorithms discussed is a linear function of N, i.e., $O(N)$, but their slopes with respect to N differ significantly. In the ML classifier, the slope depends on the set of modulations to be classified and the cardinality of each modulation. Moreover, the number of multiplications and additions required by the LSVM classifier scales linearly with the number of modulations, K, whereas in case of the proposed method, it does not. This suggests that for large K, the complexity of our proposed method will be significantly lower than that of LSVM. The implementation of the LSVM algorithm also requires a sorting process of the vector of features of size N (i.e., requires $O(N \log(N))$ comparison operations), in addition to the complexity required to extract these features. On the other hand, our method and the ML classifier only require $K - 1$ comparisons. It can be noticed that the computational complexity of our proposed classifier does not depend on the cardinality of the modulation in making its decision as is the case in the ML classifier. Furthermore, the complexity of our method can be reduced by adapting the number of moments used for classification with the SNR. Indeed, at low SNR values, it is enough to use a single moment in order to maintain low complexity of the proposed classifier since at low SNRs the upper bound on the performance (ML classifier) is poor anyway; hence, using multiple moments will only provide marginal accuracy gains, but will enhance the complexity extensively. At high SNR values, it is favorable to increase the number of moments used (five in our case) to maintain a good balance between accuracy and complexity.

2.4 Conclusion

We proposed a modulation classifier that uses the moments of the received signal to estimate the likelihood of each modulation being sent. The motivation of choosing moments is that they are easy to extract from the signal and have relatively low computational demands. In terms of probability of correct classification, we were able to achieve accuracies close to those of the ML classifier at high SNR values using only moments of the received signal. The proposed method outperforms the state-of-the-art machine learning classification algorithms, in AWGN and fading channels at all SNR values of practical interest. Our complexity analysis shows a significant complexity reduction compared with the ML classifier, and a competitive complexity with that of the LSVM classifier. Therefore, the proposed classifier offers an excellent trade-off between accuracy and complexity.

CHAPTER 3

DISTRIBUTION-BASED CLASSIFICATION USING NEURAL NETWORKS

3.1 Motivation

Despite their optimality when the signal model and channel state information is know at the receiver, Likelihood based classifiers suffer from a major drawback: they require prior knowledge of the pool of modulations sent by the transmitter, as well as accurate estimation of channel state information. In practical application, it is unlikely that the classifier can acquire such prior knowledge.

In order to solve this problem, extensive research is directed toward machine learning algorithms capable of classifying the signal by its modulation without requiring a signal model. This can be achieved by training the classifier to detect specific patterns in received signals and build its decision based on them. In this Section, we propose our novel approach for modulation classification using Neural Networks. We will show that our method achieves comparable classification accuracy compared to Maximum-Likelihood classifiers even in the absence of prior knowledge of the pool of modulations used. We also show that our method outperforms state-of-the-art algorithms using Machine Learning in all Signal-to-Noise ratios. In part II, We extend our work and propose an algorithm that automatically builds the regions using training data set. We show that our self-constructed regions achieves comparable results with our manual-built regions.

3.2 System Model

The baseband waveform at the receiver can be modeled as $Y(i) = hX(i) + Z(i)$, for $i = 1, \ldots, N$, where $Y(i)$ is the received signal at time instance i, $X(i)$ is the transmitted signal, $Z(i)$ is a complex white Gaussian noise with power density σ^2, and h is the channel

gain during coherence time $T = NT_s$, where T_s is the symbol duration. We assume that the channel undergoes slow fading such that h is relatively constant during the coherence time T.

3.3 Back-propagation Neural Networks

Despite their high performance, likelihood-based classifiers run into the problem of their reliance on perfect knowledge of the system model, as well as the channel state information. Back-propagation Neural Networks is a powerful tool to deal with the case of unknown channel information. The BPNN classifier has several layers. The first layer is the input layer. Each node in this layer receives the input signals and transmits them to the next layer. The layers in the middle are called hidden layers, and they receive the data from the previous layer and transmit the calculated results to the next layer. A typical BPNN usually has one or two hidden layers. The nodes in the output layer process the received data and yield the network's outputs. There are many functions for the nodes to process the received data, and the most commonly used is the Sigmoid function.

$$S(x) = \frac{1}{1 + e^{-x}} \tag{3.1}$$

For classification problems, the cross-entropy loss function is given by

$$L = \sum_{i=1}^{M} t_i \ln(y_i) \tag{3.2}$$

3.4 Distribution-based classification

Consider n regions in the complex plane $\mathbb{C}$, with R_i denoting the number of received symbols in each region. The symbols of the received signal $\mathbf{Y}$ are distributed over the n regions with the probability mass function (pmf) $p(\mathbf{R}|H_j, \Gamma)$, given by:

$$p(\mathbf{R}|H_j, \gamma) = \binom{L}{R_1, R_2, \ldots, R_n} p_1^{R_1} p_2^{R_2} \ldots p_n^{R_n}, \tag{3.3}$$

where $\mathbf{R} = \langle R_1, R_2, \ldots, R_n \rangle$, and p_i is the probability that a specific symbol is located in the i-th region, given by (3.8). It is worth noticing that $\sum p_i = 1$ for regions that are exhaustive and non-intersecting.

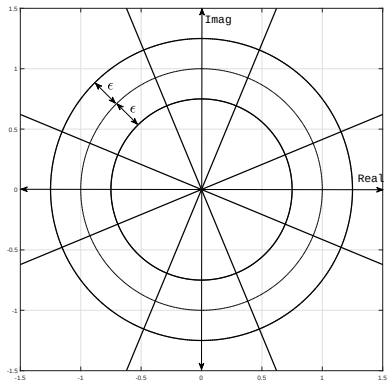


Figure 3.1: Proposed regions for classifying modulations in the set Ω

We define the optimization parameters $\epsilon = \langle \epsilon_1, \epsilon_2, \ldots \rangle$ which alter the shape and dimensions of our regions. In order to optimize these parameters, a likelihood rule is derived and optimized by maximizing it over all values of ϵ. The likelihood function can be formalized as:

$$\mathcal{L}(H_j|\mathbf{R}, \gamma) = p(\mathbf{R}|H_j, \gamma), \tag{3.4}$$

where γ is the received signal's SNR. This likelihood is only a function of the SNR and

$$p_i = \int_{\phi_l(\theta)}^{\phi_u(\theta)} \int_{r_l(\epsilon)}^{r_u(\epsilon)} \frac{r}{M\pi\sigma^2} \sum_{k=0}^{M} \exp\left(\frac{(r\cos\theta - \Re\{A_{jk}\})^2 + (r\sin\theta - \Im\{A_{jk}\})}{\sigma^2} \right)^2 dr d\theta \tag{3.7}$$

for $1 \leq i \leq n$.

---

the regions, with the later being a function of the different optimization parameters, so, in order to maximize the Likelihood function, it is required to optimize the regions' parameters. This problem can be expressed mathematically as:

$$\begin{aligned} \epsilon_{opt}(\gamma) &= \underset{\epsilon_1,\epsilon_2,\ldots}{\operatorname{argmax}} P_c(\gamma) \\ &= \underset{\epsilon_1,\epsilon_2,\ldots}{\operatorname{argmax}} \frac{1}{K} \sum_{m=1}^{K} \Pr\left(\underset{j=1,\ldots,K}{\operatorname{argmax}} \mathcal{L}(H_j|\mathbf{R},\gamma) = m | H_m \right), \end{aligned} \tag{3.5}$$

Where $P_c(\gamma)$ is the average probability of correct classification at a certain SNR. Now the optimization parameters are known, we can proceed into training the neural network using a number of training data at each SNR. Since the BPNN classifier is designed to process the values of the data close to 1, we need to normalize the received signal as:

$$\tilde{y}(n) = \frac{y(n)}{\sqrt{E[y^2]}} \tag{3.6}$$

3.4.1 Known CSI-R

Here we assume that, at the beginning of classification, the receiver has acquired Channel State Information (CSI) via training or via the sequence itself. Given the CSI-R assumption, the fading channel can be seen as an AWGN, simply by multiplying the received signal y with $\frac{h^*}{|h|^2}$. Thus, embedding the channel coefficient in the noise. The received signal now has the form $y'(n) = x(n) + \frac{z(n)}{h}$, where $y'(n)$ is just affected by

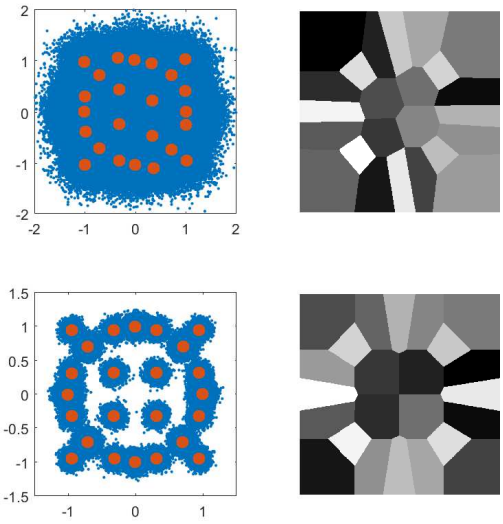


Figure 3.2: Self built regions

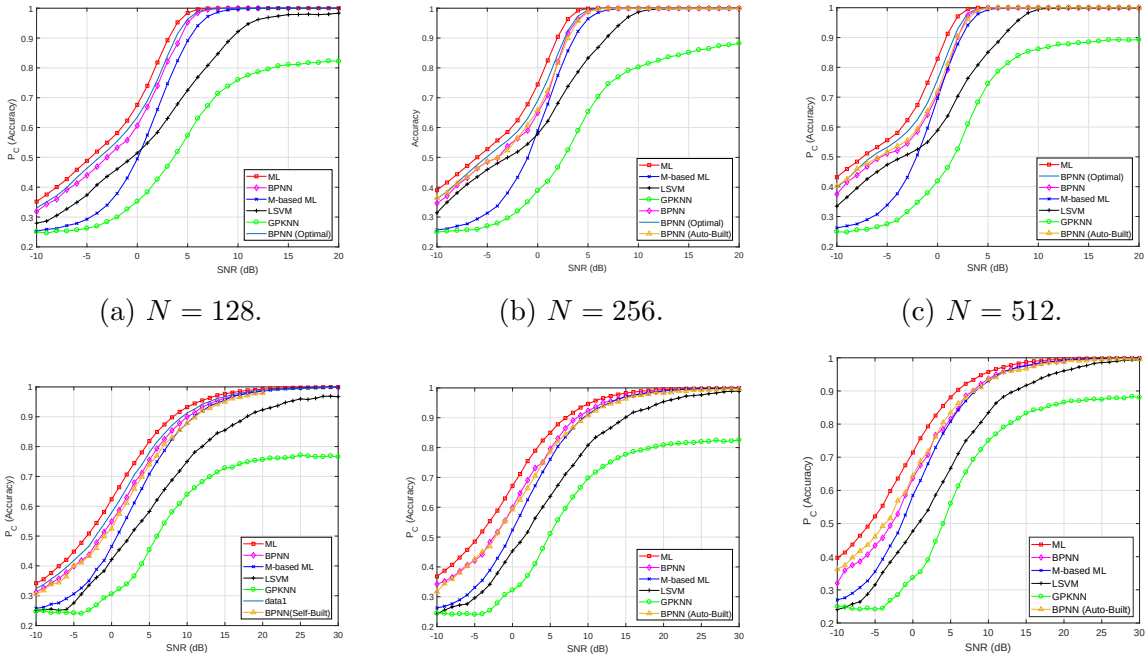
AWGN noise. This model is isomorphic to the AWGN channel model, i.e., for $z' = \frac{z(n)}{h}$, $z' \sim \mathcal{CN}(0, \sigma^2/|h|^2)$. Therefore, the assumptions made for classifying signals in AWGN channel can also be used to classify signals undergoing fading channel.

3.4.2 Unknown CSI-R

Here we assume that the receiver has no knowledge of the CSI. Unlike what has been discussed previously where CSI is known, the receiver has to estimate the CSI in order to be able to classify effectively. A number of algorithms were proposed for blind estimation of CSI-R at the receiver, including []; In this section we propose a simplified algorithm that is tailored to our classifier in the absence of CSI. We start by estimating the magnitude of h:

$$E[y^2] = E[(hx + n)^2] = E[h^2] + E[n^2], \tag{3.8}$$

The phase component can be estimated by minimizing $\sum_{i=1}^{128}[(\phi_a + \Phi_s) - \phi_d]^2$.



(a) $N = 128$. (b) $N = 256$. (c) $N = 512$.

(d) $N = 128$. (e) $N = 256$. (f) $N = 512$.

Figure 3.3: Comparison between the accuracy of correct classification versus SNR (dB) of four different classification methods under different channel assumptions: AWGN (a)–(c), and Rayleigh fading (d)–(f), and various sequence lengths. The modulations used are BPSK, QPSK, 16QAM and 8PSK.

3.5 Conclusion

We proposed a modulation classifier that uses the moments of the received signal to estimate the likelihood of each modulation being sent. The motivation of choosing moments is that they are easy to extract from the signal and have relatively low computational demands. In terms of probability of correct classification, we were able to achieve accuracies close to those of the ML classifier at high SNR values using only moments of the received signal. The proposed method outperforms the state-of-the-art machine learning classification algorithms, in AWGN and fading channels at all SNR values of practical interest. Our complexity analysis shows a significant complexity reduction compared with the ML classifier, and a competitive complexity with that of the LSVM classifier. Therefore, the proposed classifier offers an excellent trade-off between accuracy and complexity.

References

[1] Z. Zhu and A. K. Nandi, *Automatic Modulation Classification: Principles, Algorithms, and Applications*. John Wiley & Sons, Ltd, 2014, pp. 1–17.

[2] ——, *Likelihood-based Classifiers*. John Wiley & Sons, Ltd, 2014, pp. 35–47. [Online]. Available: http://dx.doi.org/10.1002/9781118906507.ch3

[3] F. Wang, O. A. Dobre, C. Chan, and J. Zhang, "Fold-based Kolmogorov–Smirnov modulation classifier," *IEEE Signal Process. Lett.*, vol. 23, no. 7, pp. 1003–1007, Jul. 2016.

[4] Z. Zhu, M. W. Aslam, and A. K. Nandi, "Genetic algorithm optimized distribution sampling test for M-QAM modulation classification," *Signal Process.*, vol. 94, pp. 264 – 277, 2014.

[5] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Trans. on Commun.*, vol. 48, no. 3, pp. 416–429, Mar 2000.

[6] S. Huang, Y. Yao, Z. Wei, Z. Feng, and P. Zhang, "Automatic modulation classification of overlapped sources using multiple cumulants," *IEEE Trans. on Veh. Technol.*, vol. PP, no. 99, pp. 1–1, 2016.

[7] L. Han, F. Gao, Z. Li, and O. A. Dobre, "Low complexity automatic modulation classification based on order-statistics," *IEEE Trans. on Wireless Commun.*, vol. 16, no. 1, pp. 400–411, Jan 2017.

[8] M. W. Aslam, Z. Zhu, and A. K. Nandi, "Automatic modulation classification using combination of genetic programming and KNN," *IEEE Trans. on Wireless Commun.*, vol. 11, no. 8, pp. 2742–2750, August 2012.

Appendix A

The covariance matrix in 2.9 is evaluated as follows:

$$
\begin{aligned}
& K_{mn}^{(j)} \\
=& E\big[(\tilde{M}_{p_m q_m} - \mu_{p_m q_m}^{(j)})(\tilde{M}_{p_n q_n} - \mu_{p_n q_n}^{(j)})^* | H_j, h\big] \\
\overset{(a)}{=}& E\bigg[\Big(\frac{1}{N}\sum_{i=1}^{N} Y(i)^{p_m - q_m} Y^*(i)^{q_m} - \mu_{p_m q_m}^{(j)}\Big)\Big(\frac{1}{N}\sum_{i'=1}^{N} Y(i')^{p_n - q_n} Y^*(i')^{q_n} - \mu_{p_n q_n}^{(j)}\Big)^* | H_j, h\bigg] \\
\overset{(b)}{=}& \frac{1}{N} E\big[(Y^{p_m - q_m} Y^{*q_m} - \mu_{p_m q_m}^{(j)})(Y^{p_n - q_n} Y^{*q_n} - \mu_{p_n q_n}^{(j)})^* | H_j, h\big] \\
\overset{(c)}{=}& \frac{1}{N}\bigg(E\big[(Y^{p_m - q_m + q_n})(Y^{q_m + p_n - q_n})^* | H_j, h\big] - \mu_{p_m q_m}^{(j)} \mu_{p_n q_n}^{*(j)}\bigg)
\end{aligned}
\tag{3.9}
$$

where (a) follows from the definition of $\tilde{M}_{pq}$ in 2.5; (b) follows from the fact that $Y(i), i = 1, \ldots, N$ are all i.i.d.; and (c) follows from a simple algebraic manipulation.

The $\mu_{p_i q_i}$ is computed via 2.7 and 3.10. Now, consider evaluating the term $E\big[(Y^{p_m - q_m + q_n})(Y^{q_m + p_n - q_n})^* | H_j, h\big]$. This term is tedious to evaluate in general for any p_m and q_n. For our method, we use five moments to provide a good trade-off between accuracy and complexity. Namely, we use M_{40}, M_{42}, M_{61}, M_{63}, and M_{82}. Then, $K_{mn}^{(j)}$ is the covariance between moments given hypothesis H_j, for $m, n = 1, 2, \ldots, L$, where L is the number of moments used for classification (which is five in our case). Hence, the covariance matrix $\mathbf{K}^{(j)}$ is given by

$$
\mathbf{K}^{(j)} = \begin{bmatrix}
\mathrm{Var}(M_{40}) & \mathrm{Cov}(M_{40}, M_{42}) & \mathrm{Cov}(M_{40}, M_{61}) & \mathrm{Cov}(M_{40}, M_{63}) & \mathrm{Cov}(M_{40}, M_{82}) \\
\mathrm{Cov}(M_{42}, M_{40}) & \mathrm{Var}(M_{42}) & \mathrm{Cov}(M_{42}, M_{61}) & \mathrm{Cov}(M_{42}, M_{63}) & \mathrm{Cov}(M_{42}, M_{82}) \\
\mathrm{Cov}(M_{61}, M_{40}) & \mathrm{Cov}(M_{61}, M_{42}) & \mathrm{Var}(M_{61}) & \mathrm{Cov}(M_{61}, M_{63}) & \mathrm{Cov}(M_{61}, M_{82}) \\
\mathrm{Cov}(M_{63}, M_{40}) & \mathrm{Cov}(M_{63}, M_{42}) & \mathrm{Cov}(M_{63}, M_{61}) & \mathrm{Var}(M_{63}) & \mathrm{Cov}(M_{63}, M_{82}) \\
\mathrm{Cov}(M_{82}, M_{40}) & \mathrm{Cov}(M_{82}, M_{42}) & \mathrm{Cov}(M_{82}, M_{61}) & \mathrm{Cov}(M_{82}, M_{63}) & \mathrm{Var}(M_{82})
\end{bmatrix}.
$$

For instance, consider evaluating the covariance between M_{40} and M_{61} $\mathrm{Cov}(M_{40}, M_{61})$, which is obtained as follows:

$$\begin{aligned}
&\mathrm{Cov}(M_{40}, M_{61}) \\
=&E\big[(\tilde{M}_{40} - \mu_{40}^{(j)})(\tilde{M}_{61} - \mu_{61}^{(j)})^* | H_j, h\big] \\
\overset{(a)}{=}&\frac{1}{N}\bigg(E\big[Y^5 Y^{*5} | H_j, h\big] - \mu_{40}\mu_{61}^*\bigg),
\end{aligned}$$

where (a) follows from the result in (3.9). Now the term $E\big[Y^5 Y^{*5} | H_j, h\big]$ is computed as follows:

$$\begin{aligned}
&E\big[Y^5 Y^{*5} | H_j, h\big] = E\big[(hX + Z)^5 (h^* X^* + Z^*)^5 | H_j, h\big] \\
=&E\big[(h^5 X^5 + 5h^4 X^4 Z + 10h^3 X^3 Z^2 + 10h^2 X^2 Z^3 + 5hXZ^4 + Z^5) \\
&\times (h^{*5} X^{*5} + 5h^{*4} X^{*4} Z^* + 10h^{*3} X^{*3} Z^{*2} + 10h^{*2} X^{*2} Z^{*3} + 5h^* X^* Z^{*4} + Z^{*5}) | H_j, h\big] \\
=&|h|^{10} E\big[|X|^{10} \big| H_j\big] + 25|h|^8 E\big[|X|^8 |Z|^2 \big| H_j\big] + 100|h|^6 E\big[|X|^6 |Z|^4 \big| H_j\big] \\
&+ 100|h|^4 E\big[|X|^4 |Z|^6 \big| H_j\big] + 25|h|^2 E\big[|X|^2 |Z|^8 \big| H_j\big] + E\big[|Z|^{10} \big| H_j\big] \\
=&|h|^{10} E\big[|X|^{10} \big| H_j\big] + 25|h|^8 E\big[|X|^8 \big| H_j\big] E\big[|Z|^2\big] + 100|h|^6 E\big[|X|^6 \big| H_j\big] E\big[|Z|^4\big] \\
&+ 100|h|^4 E\big[|X|^4 \big| H_j\big] E\big[|Z|^6\big] + 25|h|^2 E\big[|X|^2 \big| H_j\big] E\big[|Z|^8\big] + E\big[|Z|^{10}\big]. \qquad (3.10)
\end{aligned}$$

Now, since we are given h, the expectations in (3.10) are straightforward to compute. For $Z \sim \mathcal{CN}(0, \sigma^2)$ and $n \in \mathbb{Z}$, $E\big[|X|^{2n} \big| H_j\big] = \frac{1}{M_j} \sum_{k=1}^{M_j} |S_{jk}|^{2n}$, where $S_{jk} \in \omega_j$ is the k-th modulation symbol belonging to the modulation ω_j, and $M_j = |\omega_j|$, for $j = 1, \ldots, K$. Further, since the noise variance is known at the receiver, we compute the moments of Z using the fact that $E[|Z|^{2n}] = n! \times \sigma^{2n}$. A similar procedure has been used to find the remaining entries of the covariance matrix $\mathbf{K}^{(j)}$.