Analysis of Spectrum Occupancy Using Machine Learning Algorithms

Freeha Azmat, Yunfei Chen, *Senior Member, IEEE*, and Nigel Stocks

Abstract

In this paper, we analyze the spectrum occupancy in cognitive radio network (CRN) using different machine learning techniques. Both supervised techniques (naive Bayesian classifier (NBC), decision trees (DT), support vector machine (SVM), linear regression (LR)) and unsupervised algorithm (hidden markov model (HMM)) are studied to find the best technique with the highest classification accuracy (CA). A detailed comparison of the supervised and unsupervised algorithms in terms of the computational time and classification accuracy is performed. The classified occupancy status is further utilized to evaluate the blocking probability of secondary user for future time slots, which can be used by system designers to define spectrum allocation and spectrum sharing policies. Numerical results show that SVM is the best algorithm among all the supervised and unsupervised classifiers. Based on this, we proposed a new SVM algorithm by combining it with fire fly algorithm (FFA), which is shown to outperform all other algorithms.

Index Terms

Fire fly algorithm, hidden markov model, spectrum occupancy and support vector machine.

I. INTRODUCTION

A cognitive radio network (CRN) is composed of two types of users, namely, the licensed primary users (PU's) and the unlicensed secondary users (SU's). The core idea behind CRN is to allow unlicensed user's access to the licensed bands in an opportunistic manner to avoid interference with the licensed users. To achieve this, a realistic understanding of the dynamic usage of the spectrum is required. The spectrum measurement is an important step towards the realistic understanding of the dynamic spectrum usage. Various spectrum measurement campaigns covering a wide range of frequencies have been performed [1]. These spectrum measurement studies have found significant amount of unused frequency bands in the case of normal usage due to the static spectrum regulations. This has led researchers to understand the spectrum occupancy characteristics in depth for exploiting the free spectrum.

A. Problem definition

Many studies have been performed to understand the occupancy statistics. For instance, the statistical and spectral occupation analysis of the measurements was presented in [2] in order to study the traffic density in all frequency bands. In [3], auto-regressive model was used to predict the radio resource availability using occupancy measurements in order to achieve uninterrupted transmission of secondary users. In [4], the occupancy statistics were utilized to select the best channels for control and data transmission purposes, so that less time is required for switching transmission from one channel to the other in the case when the PU appears. Further, in [5], [6], the bandwidth efficiency was maximized by controlling the transmission power of cognitive radio using spectrum occupancy measurements.

In [7], different time series models were used to categorize specific occupancy patterns in the spectrum measurements. In [8], a novel time-varying statistical model for spectrum occupancy is proposed that uses real time wireless frequency measurements for predicting the arrival rate of PUs in each frequency bin by assuming the Poisson distribution on the arrival rates of PUs and the exponential distribution on idle durations. All of the aforementioned works have evaluated the spectrum occupancy models by using conventional probabilistic or statistical tools. These tools are often limited due to assumptions required to derive their theories. For example, one has to determine whether the value is a random variable or a random process in order to use the probabilistic and statistical tools. On the other hand, machine learning (ML) is a very powerful tool that has received increasing attention recently [9]. The machine learning algorithms are often heuristic, as they don't have any prerequisites or assumptions on data. As a result, in many cases, they provide higher accuracy than conventional probabilistic and statistical tools. There are very few works on the use of ML in spectrum occupancy. The ML works related to cognitive radio (CR) in [10]- [24] discussed cooperative spectrum sensing and spectrum occupancy variation. However, in this paper, we aim to provide a comprehensive investigation on the use of ML for analyzing spectrum occupancy. The motivation is that different ML algorithms are often suitable for different types of data. Thus, one needs to try different ML algorithms in order to find the one that suits the spectrum data best, not just one ML algorithm.

B. Contributions

The contributions of this work can be listed as follows:

1. We propose the use of ML algorithms in spectrum occupancy study. Both supervised and unsupervised algorithms are used. In [10] and [11], ML was used for cooperative spectrum sensing. However we use ML for spectrum occupancy modelling that may be used in all CR operations, including spectrum management, spectrum decision and spectrum sensing. In [12], authors have discussed call-based modelling for analyzing the spectrum usage of the dataset collected from the cellular network operator. However, we use ML to model spectrum occupancy in time slots for all important bands.

2. We have utilized four supervised algorithms, naive Bayesian classifier (NBC), decision trees (DT), support vector machine (SVM), linear regression (LR), and one unsupervised algorithm, hidden markov model (HMM), to classify the occupancy status of time slots. The classified occupancy status is further utilized for evaluating the blocking probability. In [13], HMM was

used to predict the channel status. Our supervised algorithms and modified HMM all perform better than HMM. In [24], LR was used to investigate the spectrum occupancy variation in time and frequency. Our approach outperforms LR as well.

3. We propose a new technique that combines SVM with fire fly algorithm (FFA) to outperforms all supervised and unsupervised algorithms.

The rest of the paper is organized as follows: Section II explains the system model, followed by the detailed explanation of classifiers in Section III. The numerical results and discussion are presented in Section IV.

II. System Model

A. Measurement setup and data

We have measured the data from 880 MHz to 2500 MHz containing eight main radio frequency bands for approximately four months (6th Feb-18th June 2013) at the University of Warwick using radiometer. The eight bands are: 880-915 MHz, 925-960 MHz, 1900-1920 MHz, 1920-1980 MHz, 1710-1785 MHz, 1805-1880 MHz, 2110-2170 MHz and 2400-2500 MHz. The number of the frequency bins in each band varies. For example, the band 925-960 MHz contains 192 frequency bins, each occupying a bandwidth of 0.18 MHz, while the band 1710-1785 MHz contains 448 frequency bins, each occupying a bandwidth of 0.167 MHz. The data is arranged in a two dimensional matrix (t_i, f_j) for each band; where each row t_i represents the measured data at different frequencies in one minute while each column f_j represents the data at different time instants of each frequency bin. As we have measured the data for four months which constitute 131 days (188917 minutes), the numbers of rows are 188917 while the number of columns varies according to the number of the frequency bins in a particular band.

B. SU Model

In a network of licensed users, SU is allowed to access the licensed band without causing any harmful interference to the PU. Let i denote the time slot and j denote the frequency bin, where $i = 1, 2, ..n$, $j = 1, 2, ...k$, n represents the total number of time slots and k represents the total number of frequency bins. Using energy detection [15], if $y^i(j)$ is the sample sensed at the i^{th} time slot in the j^{th} frequency bin. One has

$$y^i(j) = x^i(j) + w^i(j) \tag{1a}$$

$$or \qquad y^i(j) = w^i(j) \tag{1b}$$

where $x^i(j)$ represents the received PU signal and $w^i(j)$ represents the additive white Gaussian noise (AWGN) with zero mean and variance σ_w^2. Each sample is compared with a threshold (γ). The selection of γ is very important because small values of γ will cause false alarms while large values will miss spectrum opportunities. The computation of γ was explained in [16]. In our approach, the threshold is dynamic and its selection is explained in Section IV-B. The spectrum status is given as

$$S^i(j) = \begin{cases} 1, & y^i(j) > \gamma \\ 0, & y^i(j) < \gamma. \end{cases} \tag{2}$$

The occupancy for the ith time slot is defined as

$$OC^i = \frac{\sum_{j=1}^{k} S^i(j)}{k}. \tag{3}$$

For example, a three-minute interval for the band 880 - 890 MHz having 9 frequency bins as shown in Fig.1, where each bin occupies 1 MHz. For each frequency bin, $S^i(j)$ is decided. Once $S^i(j)$ is evaluated, the occupancy OC^i is calculated using (3). It is observed that more frequency bins are occupied for the first minute than for the second and third minutes, so SU has less chance to transmit in the first minute. Following the discussion above, we need to set the criteria for quantifying this chance based on the occupancies.

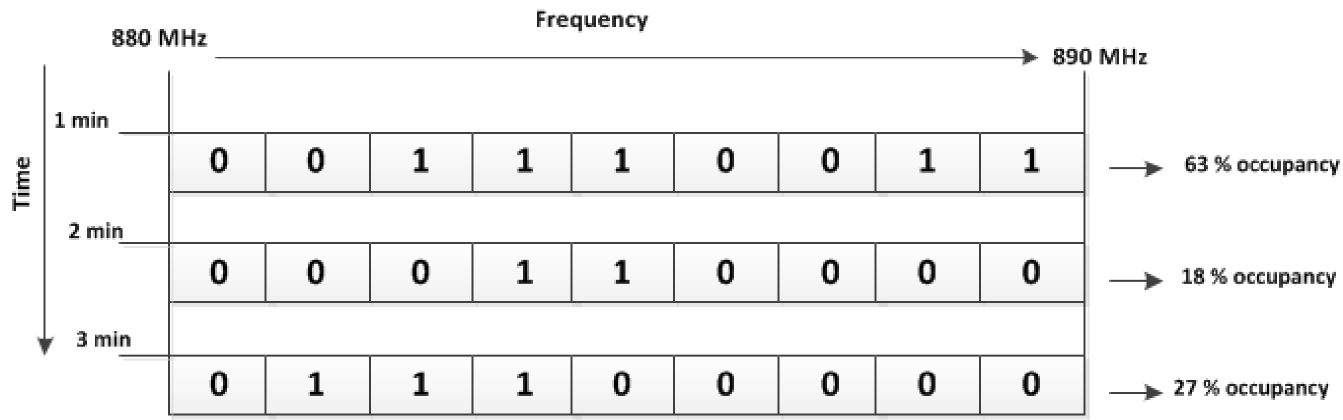


Fig. 1. Occupancy for different time slots in the band.

C. PU Model

As per our approach, the status of PU (P^i) for the i^{th} time slot can be decided using the following rules:

$$P^i = \begin{cases} 1, & OC^i > U_{oc} \quad (Condition\, 1) \\ 1, & L_{oc} <= OC^i <= U_{oc} \quad AND \quad con^i < B \quad (Condition\, 2) \\ 0, & L_{oc} <= OC^i <= U_{oc} \quad AND \quad con^i >= B \quad (Condition\, 3) \\ 0, & OC^i < L_{oc} \quad (Condition\, 4) \end{cases} \tag{4}$$

where U_{oc} and L_{oc} represent the maximum and minimum values of occupancy for all n time slots, con^i represents the number of consecutive free frequency bins in the ith time slot and B represents the maximum value of con^i, when PU is considered present. Each condition is explained as follows:

1. Condition 1 and Condition 4: The values of U_{oc} and L_{oc} vary with the frequency band, the

day and the threshold. Our test show that U_{oc} should not be less than 75% and L_{oc} should not be greater than 40%. For fixed frequency band and day, we have evaluated U_{oc} and L_{oc} for different thresholds in Section IV-B. In order to guarantee PU protection and ensure SU transmission, when the values of OC^i lie in the range between L_{oc} and U_{oc}, further criterion is applied.

2. Condition 2 and Condition 3: It is difficult to apply Condition 1 and Condition 4 when $L_{oc} <= OC^i <= U_{oc}$, so we evaluate con^i for each time slot. If $con^i > B$ for $L_{oc} <= OC^i <= U_{oc}$, there exists at least B consecutive free frequency bins in the ith time slot; thus SU can transmit and vice versa when $con^i > B$. The value of B is selected to provide PU protection. This will be explained in Section IV-B.

D. *Machine Learning Framework for SU and PU Model*

ML constructs a classifier to map $\mathbf{S}^i$ to P^i, where $\mathbf{S}^i = [S^i(1), S^i(2), ..S^i(k)]$ represents the feature vector and P^i is the corresponding response to the feature vector. There are two steps for constructing a classifier:

1) Training: Let $\mathbf{S}^i_{train} = [S^i(1)_{train}, S^i(2)_{train}..., S^i(k)_{train}]^T$ denote the training spectrum status and P^i_{train} represent the training PU status for the ith time slot, respectively, where $i = 1, 2, ..n_1$ and n_1 represents the number of training time slots fed into the classifier.

2) Testing: Once the classifier is successfully trained, it is ready to receive the test vector for classification. Let $\mathbf{S}^i_{test} = [S^i(1)_{test}, S^i(2)_{test}..., S^i(k)_{test}]^T$ denote the testing spectrum status and P^i_{test} represent the testing PU status for the ith time slot, respectively, where $i = n_1{+}1, n_1{+}2, ..n_2$ and n_2 represents the length of testing sequence. It is assumed that $n = n_1{+}n_2$. For our proposed approach, the matrix of size $n * k$ is divided into 15% training data matrix of size $n_1 * k$ and 85% testing data matrix of size $n_2 * k$. The value P^i_{test} is not used during the testing but as a reference for computing the classification error.

3) Classification Accuracy (CA): Let P^i_{eval} denote the PU status determined by the classifier for the ith time slot. The classifier categorizes the testing vector $\mathbf{S}^i_{test}$ as 'occupied class' (i.e., $P^i_{eval} = 1$) or 'unoccupied class' (i.e., $P^i_{eval} = 0$). Therefore, the PU status is correctly determined, when P^i_{eval}= P^i_{test}, giving $CA^i = 1$. The misdetection occurs when $P^i_{eval} = 0$ and $P^i_{test} = 1$, while false alarm occurs when $P^i_{eval} = 1$ and $P^i_{test} = 0$, both giving $CA^i = 0$.

E. Blocking Probability

Let $\mathbf{P}^i_{eval}$ be a vector evaluated by each classifier, and P^i_{eval} represent the presence/ absence of PU for the i^{th} time slot. When $P^i_{eval} = 0$, SU is allowed to utilize the i^{th} time slot. Define out_{su} as the minimum value of consecutive free time slots required by SU for transmission. The blocking probability measures the chance when SU cannot find out_{su} consecutive free time slots for transmission given as [17]:

$$P(SU_{blocking}) = 1 - P(SU_{transmit}) \tag{5a}$$

where

$$P(SU_{transmit}) = \sum_{c=1}^{C} P(FB_c) \tag{5b}$$

where FB_c is the block of free consecutive time slots of length out_{su}, $c = \{1, 2, ..C\}$ and C represents the total number of free blocks in $\mathbf{P}^i_{eval}$. The probability for a free block starting at index, say r, in $\mathbf{P}^i_{eval}$ is evaluated using the following equation

$$P(FB_c) = \prod_{i=r}^{r+out_{su}} OC^i. \tag{5c}$$

III. Proposed Algorithms

In the proposed approach, five machine learning algorithms are utilized to predict the future PU status using the occupancy data, which is a function of time, frequency and threshold. Among them, four are supervised learning algorithms: NBC, DT, SVM and LR, while one is an unsupervised algorithm, HMM. The motivation to use five different algorithms is to find the best machine learning algorithm for predicting future status as they have different characteristics.

A. Naive Bayesian Classifier

A naive Bayesian classifier is a generative model based on the Bayes theorem. It is also called 'independent feature model' because it does not take dependency of features into account. The feature vector for the ith time slot in our model contains all the samples which are independent of each other, since every feature represents a specific frequency bin. For example, the status vector of the ith time slot is given as $\mathbf{S}^i = S^i(1), S^i(2), S^i(3), .., S^i(k)$, where $S^i(1)$ is independent of $S^i(2)$. However, the response variable in our approach i.e. PU status (P^i) is a dependent variable which is affected by each frequency bin.

The probability of $\mathbf{S}^i$ belonging to the class P^i evaluated using the Bayes theorem is formally defined as [18]

$$p(P^i, \mathbf{S}^i) = p(P^i) * p(\mathbf{S}^i|P^i). \tag{6}$$

When $P^i = 0$, $\mathbf{S}^i$ will be classified as 'idle' class, while when $P^i = 1$, $\mathbf{S}^i$ will be classified as 'occupied' class. The goal is to find the class with the largest posterior probability in the classification phase. The classification rule is given as

$$classify(\hat{\mathbf{S}^i}) = argmax_{\mathbf{S}^i}\{p(P^i, (\hat{\mathbf{S}^i})\} \tag{7}$$

where $\hat{\mathbf{S}^i} = \{S^i\hat{(1)}, S^i\hat{(2)}...S^i\hat{(k)}\}$. NBC is sensitive to the choice of kernel and the prior probability distribution of classes. This will be explained in Section IV-B.

B. Decision Trees

Decision tree builds classification or regression models in the form of a tree structure. The decision trees used in this approach are classification trees whose leaf represents the class labels. Unlike NBC, it can handle feature interactions and dependencies. In DT, the decision is made on each internal node, which is used as a basis for dividing the data into two subsets while leaf nodes represent the class labels (in the case of classification trees) or the real numbers (in the case of regression trees). Data come in the form:

$$(\mathbf{S}^i, P^i) = (S^i(1), S^i(2), S^i(3).., S^i(k), P^i). \tag{8}$$

where P^i is the dependent variable representing the class label of the ith time slot. The class labels P^i are assigned by calculating the entropy of the feature, as [19]

$$Entropy(t) = -\sum_{id=0}^{Z} p(id|t)\ log_2 p(id|t). \tag{9}$$

Where $p(id|t)$ denote the fraction of records belonging to class id at a given node t and Z represents the total number of classes. In our approach, $Z = 1$. The smaller entropy implies that all records belong to the same class. It will be discussed in Section IV-C on how fraction of records per node affects the classification accuracy of DT.

C. *Support Vector Machines*

SVM is a discriminative classifier with high accuracy. The phenomenon of over fitting [1] often happens in DT, however SVM tends to be resistant to over-fitting and can be used for online learning [2]. There are two types of classifiers in SVM: linear SVM for separable data [3] and non-linear SVM for non-separable data . Linear SVM is used in our approach.

The training feature and response vectors in our system is represented as $D = (P^i, \mathbf{S}^i)$ where $P^i \in \{0, 1\}$. Following the definition of linear SVM, the two classes are separated by defining a hyper plane H, represented as $\mathbf{w}.\mathbf{S}^i = \rho$, where $\mathbf{w}$ represent the normal vector and ρ represents the constant separating occupied and idle classes ($P^i \in \{0, 1\}$) given as [22]:

$$P^i = +1 \quad when \quad \mathbf{w}.\mathbf{S}^i > \rho \qquad (Occupied\, class) \tag{10a}$$

[1]It is a condition when the ML model fits the training set very well but fails to generalize to the unseen examples [20].

[2]It is learning scenario in which training data is provided one example at a time, as opposed to the batch mode in which all examples are available at once. [21].

[3] Two sets of points A and B are linearly separable if there exists n real numbers $w_1, w_2, w_3, .., w_n$, such that every point $a_i \in A$ satisfies $\sum_{i=1}^{n} w_i.a_i > \rho$ and every point $b_i \in B$ satisfies $\sum_{i=1}^{n} w_i.b_i < \rho$, where ρ represents the constant separating two sets A and B [20].

$$P^i = 0 \quad when \quad \mathbf{w}.\mathbf{S}^i < \rho \qquad (Idle\, class) \tag{10b}$$

Two margins are defined on both sides of H to maximize the gap between two classes. The length of the margins is controlled by a parameter called box constraint Box_{ct}. We will evaluate the optimal value of Box_{ct} using a bio-inspired technique i.e. FFA in our approach.

D. SVM with Fire Fly Algorithm

In FFA, let X be a group of fire flies, $X = [l_1, l_2, ..l_X]$, initially located at specific positions $a_X = [a_{l_1}, a_{l_2}, ..a_{l_X}]$. Each fire fly moves and tries find a brighter fire fly, which has more light intensity than its own. The objective function $f(x)$ used for evaluating the brightness of the fire fly in our approach is the classification accuracy i. e. $f(x) = CA(a_X)$. When a fire fly, say l_1 finds another brighter fire fly l_2 at another location having more intensity compared to its own, it tends to move towards fire fly l_2. The change in position is determined as [23]

$$a_{l_1}^{v+1} = a_{l_1}^{v} + \beta_0 e^{-\psi_{l_1 l_2} r_{l_1 l_2}^2}(a_{l_2}^{v} - a_{l_1}^{v}) + \alpha(rand - 0.5) \tag{11}$$

where v represents the number of iterations, a_{l_1} and a_{l_2} represents the position of fire fly l_1 and l_2 respectively, α, β_0 and $\psi_{l_1 l_2}$ are constants, $rand$ is a uniformly distributed random number and $r_{l_1 l_2}$ represents the euclidean distance between l_1 and l_1. For our approach, the starting positions of the X fire flies are initialized, while the position of each fire fly represents the value of box constraints Box_{ct}.

E. Linear Regression

The flexibility of linear regression to include mixture of various features in different dimensions e. g. space, frequency, time and threshold as a linear combination is the main motivation of using it for modeling in this approach. The linear regression model for our approach is given by [24]:

$$P^i = e_0 + e_1 S^i(1) + e_2 S^i(2) + ... + e_k S^i(k) = e_0 + \sum_{j=1}^{k} e_j S^i(j). \tag{12}$$

where the class label P^i is represented as a linear combination of parameters $e_1, e_2, , e_k$ and features $(S^i(1), S^i(2), .., S^i(k))$ in the ith time slot. The stepwise-linear regression is used in this approach. In each step, the optimal term based on the value of defined 'criterion' is selected. The 'criterion' can be set as the sum-of-squares error (SSE), deviance, akaike information criterion (AIC), Bayesian information criterion (BIC) or R-squared etc. SSE is used in this approach. It is observed from (12), that the computational time for evaluating the response of the model linearly increases with the number of frequency bins/ predictors involved. So we need to select an appropriate number of predictors for linear regression.

F. Hidden Markov Models

It is an unsupervised algorithm for modeling the time series data. The motivation to use the unsupervised algorithm is that it does not need the training phase. In HMM, the sequence of states can be recovered by an analysis of the sequence of observations. The set of states and observations are represented by U and G given as, $U = (u_1, u_2, ...u_N)$, $G = (g_1, g_2, ...g_M)$, where u_1 and u_2 represent the states when $P^i = 0$ and $P^i = 1$, respectively. The observations g_1 and g_2 represent the value of OC^i corresponding to each P^i. HMM is defined as

$$\lambda = (C_h, D_h, \pi) \tag{13}$$

where the transition array C_h is the probability of switching from state u_1 to state u_2 given as [25], $C_h = [c_{12}] = P(q_t = u_2 | q_{t-1} = u_1)$. The D_h is the probability of observation g_1 being produced from state, $D_h = [d_{1,2}] = P(o_t = g_{1,2} | q_t = u_2)$ and π is the initial probability array, $\pi = P(q_1 = u_2)$.

HMM has two main steps. In the first step, the sequence of observations $O = (o_1, o_2, ...o_T)$, transition probability matrix C_h and emission probability matrix D_h are utilized to find the probability of observations O given hmm model λ using ([25], Eq.13) as, $P(O|\lambda) = \sum_Q P(O|Q, \lambda)P(Q|\lambda)$, where $Q = (q_1, q_2, ...q_T)$ and $P(O|Q, \lambda) = \prod_{t=1}^{T} P(o_t|q_t, \lambda) = g_{q_1}(o_1) * g_{q_2}(o_2)..g_{q_T}(o_T)$. The probability of the state sequence is given as $P(Q|\lambda) = \pi_{q_1} c_{q_1 q_2} c_{q_2 q_3} ... c_{q_{T-1} q_T}$. In the second step, the hidden state sequence, that is most likely to have produced an observation is decoded using the viterbi algorithm. The most likely sequence of states Q_L generated using the viterbi algorithm is matched with the expected fixed state sequence Q to compute classification accuracy. HMM can be also be supervised by adding two extra steps as

Step(a): Use the initial guesses of C_h and D_h to compute Q and O, that are used for computing $P(O|\lambda)$ in forward algorithm

Step(b): Use O, D_h and C_h in Step(a) to estimate the transition probability matrix $C_{h'}$ and emission probability matrix $D_{h'}$ using maximum likelihood estimation [26].

The $C_{h'}$ and $D_{h'}$ collectively form the estimated HMM model (λ_e) that can be further used for evaluating $P(O|\lambda)$ and Q_L using the forward algorithm and the Viterbi algorithm respectively.

IV. Numerical Results and Discussion

In order to analyze the occupancy of the eight bands, the statistics of data in all bands from 880 to 2500 MHz are presented in Section IV-A. The classification criteria are explained in Section IV-B. The selection of the best parameters for each model using the classification criteria are discussed in Section IV-C. The classification models with the optimal parameters are compared to find the best classifier in terms of the CA, defined as $CA = \frac{\text{No. of correct classfications}}{\text{Total number of test samples}}$

A. Statistics of Data

The CDF plot is shown in Fig.2 which gives the summarized view of all power ranges for the eight bands. It can be observed from Fig.2 that the eight bands can be categorized into two main groups. Group A includes 925 - 960 MHz, 1805 - 1800 MHz and 2110 - 2170 MHz while Group B has five bands: 880 - 915 MHz, 1710 - 1785 MHz, 1900 - 1920 MHz, 1920 - 1980 MHz and 2400 - 2500 MHz. Group A and Group B are different given as:

1) Average Power: Group A bands have higher value of the average power than Group B bands. The results are shown in Table I.

2) Standard Deviation (σ): We evaluated the maximum standard deviation in frequency bins (σ^j) and time slots (σ^i) over the period of four months using k frequency bins. It is observed that Group A bands have higher values of both σ^i and σ^j compared to Group B bands as shown in Table I.

3) Periodicity: Due to stochastic noise and non-linearities, it is impossible to find the pure periodic and stationary signal in real time. However we found reasonably periodic structures in some bands by computing the relationship between occupancy and frequency, given as ($OC^j = \frac{\sum_{i=1}^{n} S^i(j)}{n}$). We found that Group B bands can be classified as periodic bands while Group A bands do not have this property. One band from each group is shown in Fig.3.

The periodicity may be caused by the usage pattern. For instance, the periodicity in each band lies in their uplink/ downlink usage pattern. For instance, the band 1710-1785 MHz is an uplink band, while the aperiodic band 1805-1880 MHz is the downlink. The uplink transmits data from the mobile user to base station so that its activity is completely determined by mobile users's periodic usage pattern. On the other hand, the downlink transmits the data from base station to the mobile user so that its activity is also affected by control and broadcast channels, making it less or non periodic.

We found that OC^i and OC^j for both groups share similar trend, where OC^j is greater than OC^i for all bands. The threshold selection is an important task for analyzing both OC^j and OC^i. Therefore, we took the minimum and the maximum value of power for each frequency band and tested seven values of thresholds in this range. Each band is analyzed separately for the seven values of the threshold using the four months data. It is observed for all bands that occupancy monotonically decreases when the value of threshold increases. These results have proved that larger value of threshold will classify less samples as occupied. The effect of different threshold levels on mean occupancy for one band from each group is shown in in Fig.4.

B. Classification Criteria

This subsection studies the choice of U_{oc}, L_{oc}, con^i and B in Section II-C as shown in Fig 5. We have utilized Day1 (1-1440 min), Day 2 (1441-2448 min) and Day 5 (7200-8640 min) in Band 880-915 MHz, and four different values of threshold: $\gamma = [-102, -104, -106, -108]$ dBm. The parameters U_{oc} and L_{oc} will be selected by M_s, which represents the occupancy split that divides the data into occupied and idle classes. It varies from 0.1 to 0.9 with a step

TABLE I

STATISTICS OF FREQUENCY BANDS

Group	Band	Frequency Bins (k)	Mean Power (dBm)	σ^j	σ^i	OC^i (%)	OC^j (%)	Periodicity
A	925-960 MHz	192	-75.07	23.43	24.93	45.8	65.8	x
	1805-1880 MHz	448	-89.9338	21.054	23.64	32.45	86.21	x
	2110-2170 MHz	384	-71.1649	23.66	12.097	53.90	59.41	x
B	880-915 MHz	192	-101.0693	16.67	6.2322	57.06	79.1	✓
	1710-1785 MHz	448	-106.0201	9.2423	4.8510	57.71	66.52	✓
	1900-1920 MHz	128	-103.8796	6.4199	4.0611	62.25	75.17	✓
	1920-1980 MHz	384	-103.7396	15.58	4.18	61.48	72.12	✓
	2400 -2500 MHz	640	-103.5026	16.8873	7.9524	31.17	36.02	✓

size of 0.1. It is observed in Fig.5 that the value of CA depends on day and the value of threshold. The actual value of OC^i_{train} in (3) always lies in a certain range, $[L_s, U_s]$, where L_s represents the lowest value of OC^i_{train} and U_s represents the maximum value of OC^i_{train}. When $L_s <= M_s <= U_s$, two groups of classes $P^i = 0$ (available class) and $P^i = 1$ (occupied class) can be classified correctly. When $M_s > U_s$ or $M_s < L_s$, all the samples will be classified as one class because OC^i_{train} is a closed set whose values do not lie outside the range $[L_s, U_s]$. This explains why the $CA = 1$ for $[L_{oc}, U_{oc}] = [0.1, 0.2]$ and $[L_{oc}, U_{oc}] = [0.75, 0.9]$ while $CA < 1$ for $[L_{oc}, U_{oc}] = [0.2, 0.75]$ for Day 1 using $\gamma = -102$ dBm. Thus, the classification cannot be performed when $M_s > U_s$ or $M_s < L_s$. The optimal range is $[L_{oc}, U_{oc}] = [0.2, 0.75]$ for $CA < 1$. However, for $CA < 1$, there are four different choices of threshold available. In our proposed approach, we choose that specific value of threshold that contains the largest number of values between L_{oc} and U_{oc}. Following this, we have selected $\gamma = -102$ dBm for Day1, Day2 and Day5 as the optimal threshold which ensures the largest amount of samples between L_{oc} and U_{oc}. The $[L_{oc}, U_{oc}] = [0.2, 0.75]$ for Day 1, $[L_{oc}, U_{oc}] = [0.4, 0.85]$ for Day2 and $[L_{oc}, U_{oc}] = [0.2, 0.80]$ for Day 5 respectively. The optimal values of γ, U_{oc} and L_{oc} are further used for finding B for each day.

C. Model Performance Comparison

Following the discussion above, we have compared the performance of the algorithms in this section using 1 month data of Band 880-915 MHz. Our tests show that the number of minimum observations/node for DT can be seclected as 17, number of predictors for LR as 15, normal kernel for NBC and linear kernel for SVM. The optimal splitting range, optimal threshold and B will be selected corresponding to the data of each day.

1) Supervised VS Unsupervised Algorithms using $k = 55$: In Fig. 6(a), it is observed that the mean CA attained by LR, SVM, DT, NBC and HMM is 0.9257, 0.9162, 0.8483, 0.9493 and 0.4790 respectively. The mean computation time in each iteration by LR, SVM, DT, NBC and HMM is 350.19, 0.092, 0.0136, 0.0045, and 0.0171 seconds, respectively. Thus, NBC is the best considering the accuracy and complexity.

2) Supervised vs Unsupervised Algorithms using $K = 192$: We have compared HMM, Trained HMM, SVM, DT and NBC in Fig.6(b) for 30 days. Each iteration represents 1 day. LR is not shown as it takes an excessively long time in this case. It is observed that trained HMM performed better than HMM, but worst than DT, NBC and SVM. The mean CA attained by Trained HMM, HMM, SVM, DT and NBC is 0.6816, 0.4887, 0.8528, 0.8392, 0.7970 while the computational time for each iteration of Trained HMM, HMM, SVM, DT and NBC 0.0205, 0.09066, 0.0135, 0.0163, 0.0095 seconds, respectively. Thus, SVM is the best in this case with highest CA and shortest time.

3) SVM with Fire Fly Algorithm : So far, the best overall performance is attained by the linear SVM technique. The performance of linear SVM is affected by the value of Box_{ct} as illustrated in Section IV-C. The fire fly algorithm can be used to select the best value of Box_{ct}. We set $\alpha = 1$, $\beta_0 = 2$ and $\psi_{l_1 l_2} = 1.3$ for FFA. Fig. 7(a) depicts that 'SVM+FFA' performs better than the conventional SVM in most of the cases. The mean CA attained by SVM+FFA, SVM, DT, NBC and HMM is 0.8728, 0.8499, 0.7970, 0.8392 and 0.4822, respectively.

4) Comparison with Statistical/ Probabilistic Model: We compare the proposed ML classification framework with the model in [8]. The inputs of this model are the statistical parameters extracted from real time measurements. The outputs obtained from this model are the transmission times t_{ON} and t_{OFF}, where t_{ON} and t_{OFF} represents the busy and idle duration respectively. Based on t_{ON} and t_{OFF}, the model predicts the PUs occupancy for the testing data matrix.

Following [8], we have predicted the PU occupancy of the testing data matrix using data of one day and $k = 192$ frequency bins of Band 880-915 MHz. In order to compare the statistical model with our approach, we have transformed the (OC^i) evaluated using the statistical model to PU status labels (P^i), following the criteria explained in Section II-C. This is because, the validation in our approach is performed using CA, where P^i_{eval} is checked against the value of P^i_{test}. It was observed in Fig. 7(a), that ML algorithms have attained higher value of CA than statistical model. The mean CA for the statistical model is only 0.45, much lower than average value of CA attained by DT, NBC, SVM and SVM +FFA.

5) Blocking Probability: This probability is computed using SVM+FFA, SVM, DT, NBC, HMM and the statistical model [8]. It is further compared with the expected $P(SU_{blocking})$ to compute the difference between evaluated and expected values. It is evident in Fig. 7(b), that SVM+FFA has predicted the $P(SU_{blocking})$ with minimum difference and is very close to the expected one. The expected blocking probability is 0.9191 in Fig. 7(b) while the predicted $P(SU_{blocking})$ using SVM+FFA, SVM, NBC, DT, HMM and statistical model is 0.9264, 0.9322, 0.9638, 0.9577, 1 and 1, respectively. The $P(SU_{blocking})$ for HMM and statistical model is always 1, which implies that both HMM and statistical model have failed to find any block of consecutive free time slot of length out_{su}.

6) Supervised vs Unsupervised Algorithms using different Training/ Testing Data vectors: We have presented the detailed comparison of supervised and unsupervised algorithms using different sizes of training and testing data in Table II. The classification accuracy and computation time for all supervised algorithms increases with an increase in the size of the training data. SVM+FFA has attained the highest CA but with the longest computation time in most cases.

TABLE II

PERFORMANCE COMPARISON OF FIVE ML ALGORITHMS USING DIFFERENT SIZES OF TRAINING/ TESTING DATA.

Performance Comparison			
Training data, Testing data	Technique	Mean CA	Mean Computational Time (s)
15 %, 85 %	Decision Trees	0.7612	0.0132
	Support Vector Machine (SVM)	0.8945	0.0128
	SVM + Fire Fly Algorithm	0.9034	3.0412
	Hidden Markov Model	0.4925	0.0241
	Naive Bayesian	0.8714	0.0084
30 %, 70 %	Decision Trees	0.8028	0.0198
	Support Vector Machine (SVM)	0.9143	0.0153
	SVM + Fire Fly Algorithm	0.9189	3.8947
	Hidden Markov Model	0.4841	0.0191
	Naive Bayesian	0.9064	0.0098

REFERENCES

[1] Y. Chen, H-S. Oh, "A survey of measurement-based spectrum occupancy modelling for cognitive radios", *IEEE Communications Surveys and Tutorials*, vol. PP, no. 99, pp. 1, Oct 2014.

[2] V. Blaschke, H. Jaekel, T. Renk, C. Kloeck, F. K. Jondral, "Occupation measurements supporting dynamic spectrum allocation for cognitive radio design", *Proc. CrownCom'07*, pp. 50-57, Orlando, Florida, Aug. 2007.

[3] S. Kaneko, S. Nomoto, T. Ueda, S. Nomura and K. Takeuchi, "Predicting radio resource availability in cognitive radio - an experimental examination", *Proc. CrownCom'08*, Singapore, May. 2008.

[4] M. Hoyhtya, S. Pollin, A. Mammela, "Classification - based predictive channel selection for cognitive radios", *Proc. ICC'10*, pp. 1 - 6, Cape town, South Africa, May. 2010.

[5] X. Zhou, J. Ma, Y. Li, Y. H. Kwon, A. C. K. Soong, G. Zhao, "Probability-based transmit power control for dynamic spectrum access", *Proc. DySPAN'08*, pp. 1-5, Chicago, USA, Oct. 2008.

[6] X. Zhou, J. Ma, Y. Li, Y. H. Kwon, A. C. K. Soong, "Probability-based optimization of inter-sensing duration and power control in cognitive radio", *IEEE Transactions on Wireless Communications*, vol. 8, pp. 4922 - 4927, Oct. 2009.

[7] Z. Wang, S. Salous, "Spectrum occupancy statistics and time series models for cognitive radio", *Journal of Signal Processing Systems*, vol. 62, Feb. 2011.

[8] Ghosh, S. Pagadarai, D. P. Agrawal, A. M. Wyglinski, "A framework for statistical wireless spectrum occupancy modeling", *IEEE Transactions on Wireless Communications*, vol. 9, No. 1, Jan 2010.

[9] C. Rudin, K. L. Wagstaff, ”Machine learning for science and society”, *Springer Journal on Machine Learning*, vol. 95, no. 1, pp. 1-9, Nov 2013.

[10] K. W. Choi, E. Hossain, D. I. Kin, ”Cooperative spectrum sensing under a random geometric primary user network model”, *IEEE Transaction on Wireless Communications*, vol. 10, no. 6, June 2011.

[11] K. M. Thilina, K. W. Choi, N. Saquib, and E. Hossain, ”Machine learning techniques for cooperative spectrum sensing in cognitive radio networks”, *IEEE Journal on Selected Areas in Communications*, vol. 31 , no. 11, pp. 2209 -2221, Nov 2013.

[12] D. Willkomm, S. Machiraju, J. Bolot, A. Wolisz, ”Primary users in cellular networks: A large-scale measurement study”, *Proc. DySPAN' 08*, pp. 1-11, 2008.

[13] V. K. Tumuluru, P. Wang, D. Niyato, ”Channel status prediction for cognitive radio networks”, *Wiley Wireless Communications and Mobile Computing*, vol. 12, no. 10, pp. 862-874, July 2012.

[14] S. Pagadarai and A. M. Wyglinski, ”A linear mixed-effects model of wireless spectrum occupancy”, *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, no. 203178, Aug. 2010.

[15] Z. Xuping, P. Jianguo, ”Energy-detection based spectrum sensing for cognitive radio”, *Proc. CCWMSN' 07*, pp. 944 -947, Dec. 2007

[16] A. J. Petain, ”Maximizing the utility of radio spectrum: broadband spectrum measurements and occupancy model for use by cognitive radio”, *Ph.D. dissertation*, Georgia Institute of Technology, Atlanta, GA, USA, 2005.

[17] M. Zhang, B. Li, S. Jiang, ”Call blocking probabaility and packet delay in cognitive radio networks” , *Proc. WiCOM' 12*, Sept 2012.

[18] A. Y. N, M. I. Jordan, ”On discriminative vs generative classifiers: A comparison of logistic regression and Nave Bayes”, *Advances in Neural Information Processing Systems*, pp. 841-848, 2002.

[19] L. Rokach, O. Maimon, ”Decision Trees”, *Data Mining and Knowledge Discovery Handbook*, Springer Publisher, 2nd ed, 1285, 2010.

[20] P. Flach, ”Machine Learning: The Art and Science of Algorithms that make sense of data”, *Cambridge university press*, Sept 2012.

[21] P. Laskov, C. Gehl, S. Kruger, K.R. Muller, ”Incremental support vector learning: analysis, implementation and applicatins”, vol. 7, pp. 1909-1936, 2006.

[22] A. b. Hur, J. Weston, ”A user's guide to support vector machines”, *Data Mining Techniques for the Life Sciences Methods in Molecular Biology*, vol. 609, pp. 223-239, 2010.

[23] X. Yang, ”Firefly algorithms for multimodal optimization”, *LNCS 5792*, vol. 5792, pp. 169-178, 2009.

[24] S. Chatterjee, A. S. Hadi, ”Simple linear regression”, *Regression Analysis by Example*, Fourth Edition, 2006 John Wiley and Sons.

[25] G. A. Fink, ”Markov Models for pattern Recognition: From theory to Application”, *Springer Advances in Computer Vision and Pattern Recognition*, 2nd edition, 2014.

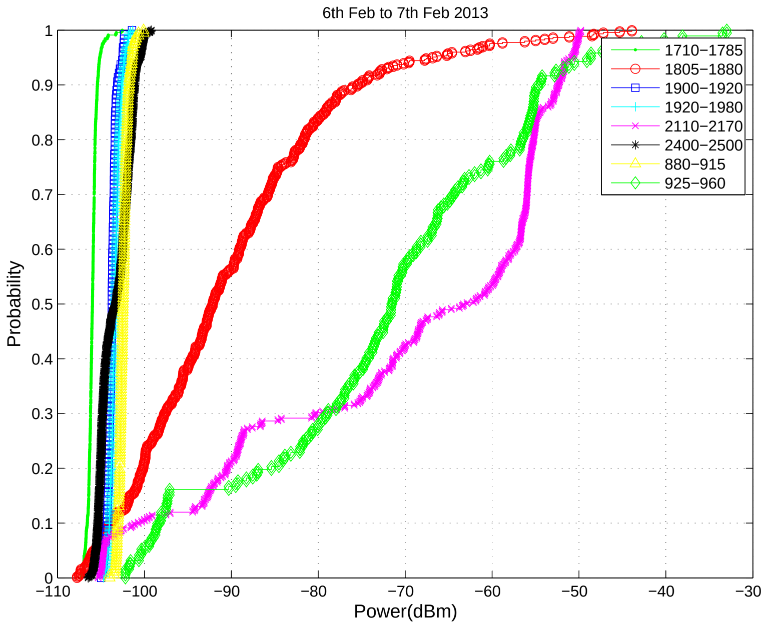
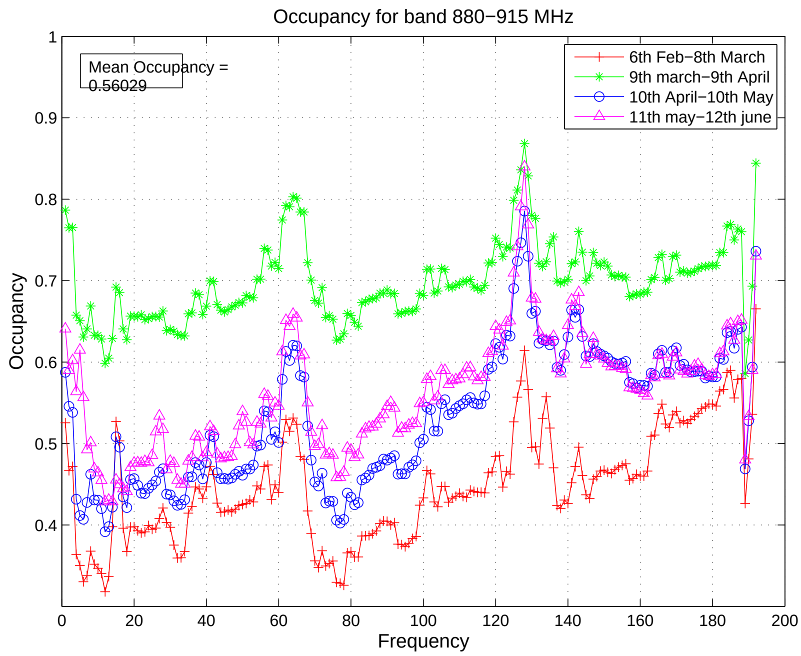
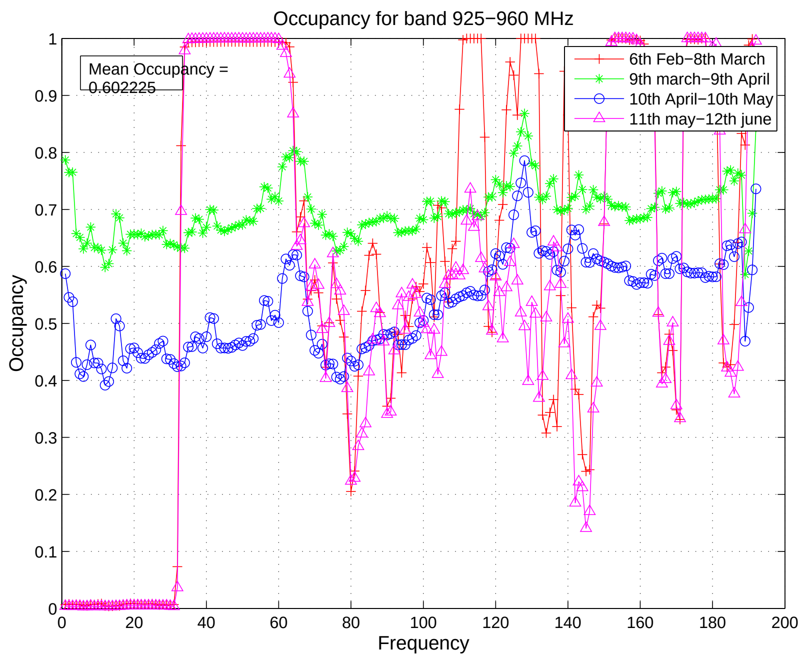


Fig. 2. The CDFs for the eight bands between 880-2500 MHz.

[26] D. Garrette, J. Baldridge, "Type-supervised hidden Markov models for part-of-speech tagging with incomplete tag dictionaries", *EMNLP - CoNLL'12*, pp. 821-831, Stroudsburg, PA, USA, 2012.



(a)



(b)

Fig. 3. Occupancy VS spectrum frequency for (a) Band 880-915 MHz (b) 925-960 MHz.

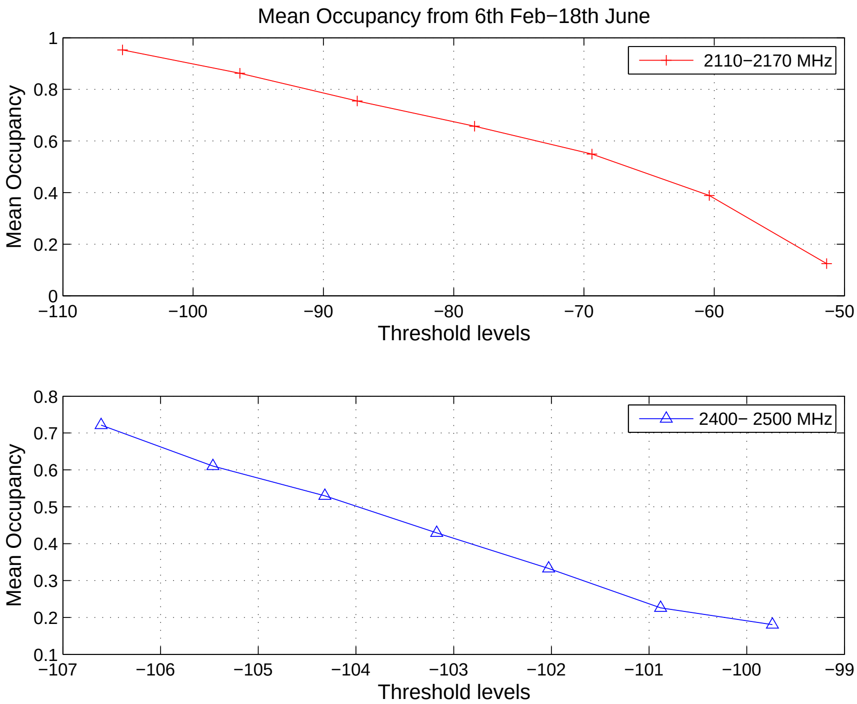


Fig. 4. Effect of different threshold levels on mean occupancy for a (a) Aperiodic band (2110-2170 MHz) (b) Periodic band (2400-2500 MHz).

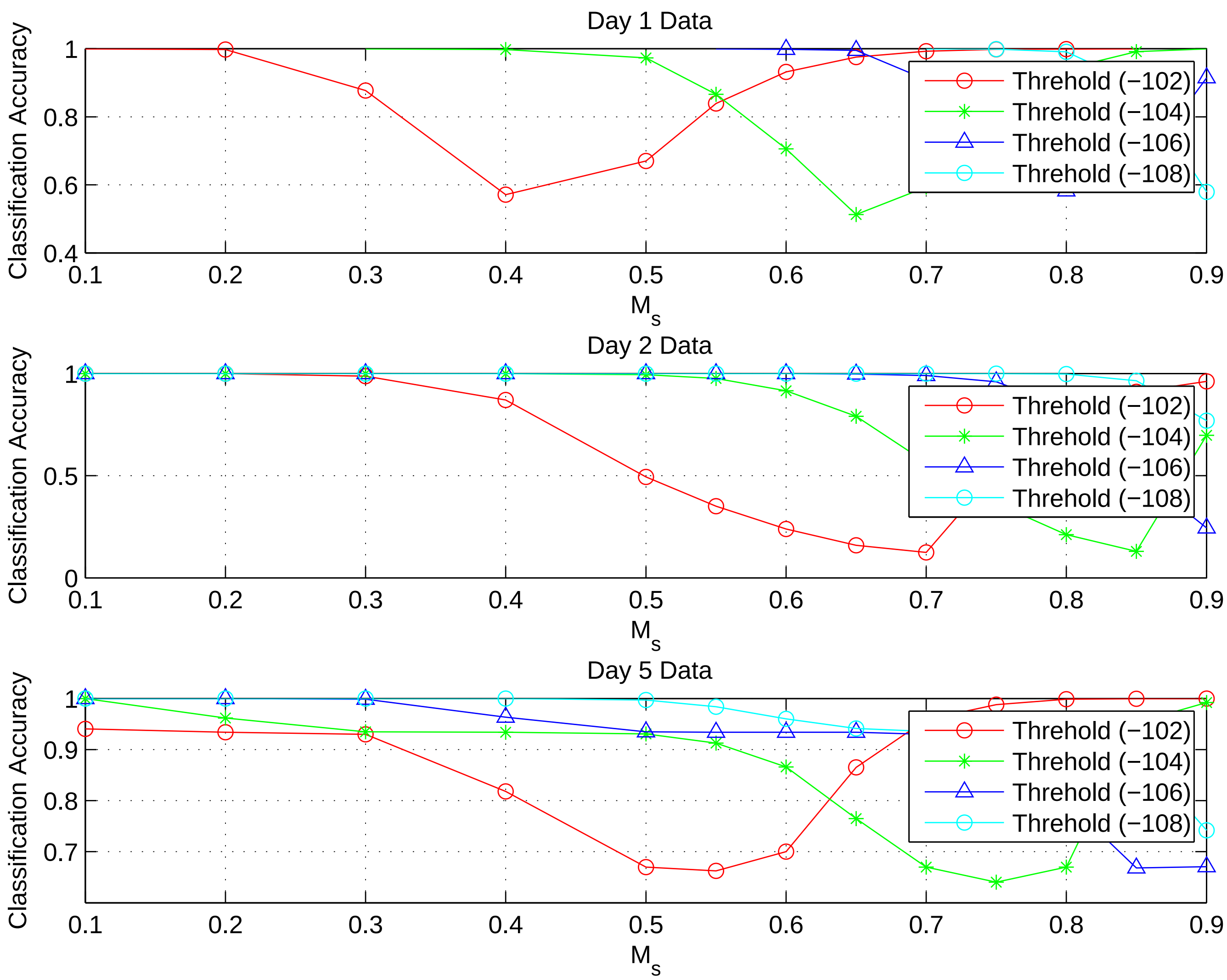
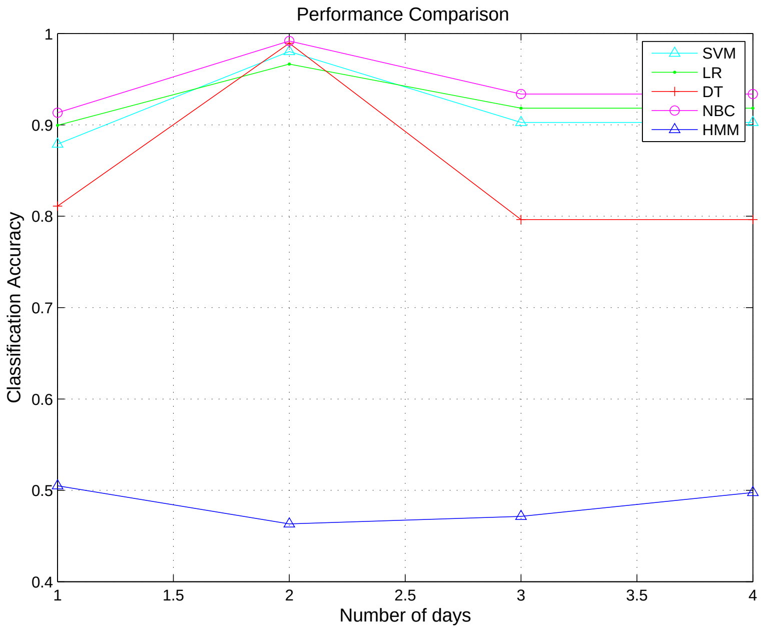
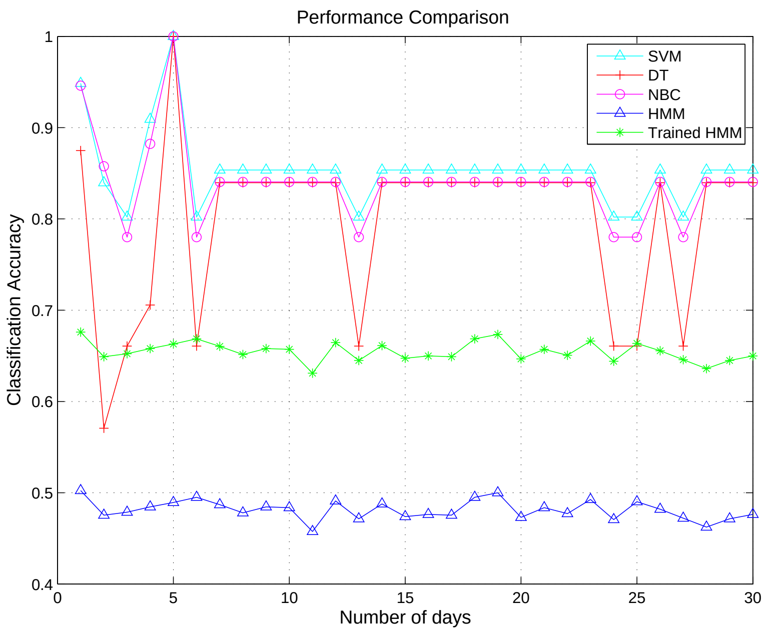


Fig. 5. Selection of optimal threshold (γ) and optimal splitting range ($[U_{oc}, L_{oc}]$) for determining the classification criteria of three days data.

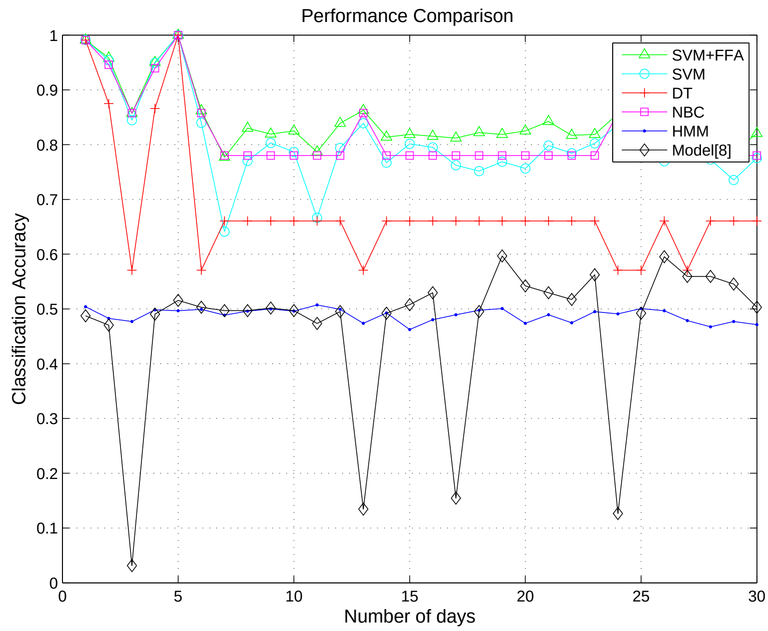


(a)

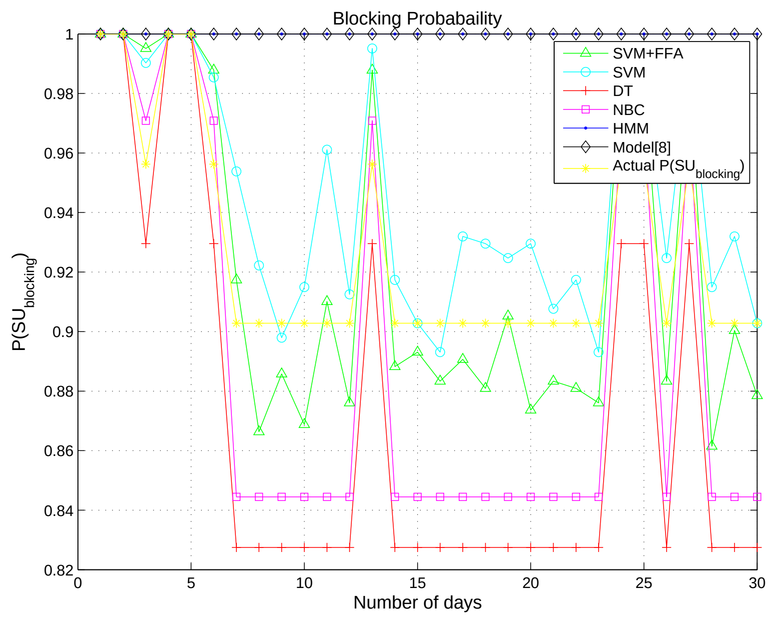


(b)

Fig. 6. Performance Comparison of (a) SVM, DT, NBC, LR and HMM with $k = 55$. (b) SVM, DT, NBC, HMM and trained HMM with $k = 192$.



(a)



(b)

Fig. 7. (a) Performance comparison of ML algorithms: SVM, DT, NBC, HMM, 'SVM+FFA' and statistical model [8] using $k = 192$ for 30 days. (b) Comparison of 'Actual $P(SU_{blocking})$' with the $P(SU_{blocking})$ evaluated using SVM, DT, NBC, HMM, 'SVM+FFA' and statistical model [8] using $k = 192$ for 30 days.