Spectrum sensing to detect n-number of primary users using n-number secondary users by applying support vector machine.

Udayamoorthy Venkateshkumar [1*], Srinivasan Ramakrishnan [2]

[1] Department of ECE, Sri Krishna College of Technology, Coimbatore, 641042, India.
[2] Department of IT, Dr.Mahalingam College of Engineering and Technology,Pollachi,642003,India.
[*]uvenkatesh2002@gmail.com, ram_f77@yahoo.com
[*]Orchid-id 0000-0002-4950-3069.

Abstract: In this paper, a cooperative spectrum sensing (CSS) model is proposed to sense n-number of primary users (PUs) using n-number secondary users (SUs) in a sequence by applying support vector machine (SVM) algorithm using three different kernels namely linear, polynomial and radial basis function (RBF) respectively. In this method, fusion centre (FC) instructs all the SUs through control channel, which PU is to be sensed by sending a pre-defined primary user identification code (PU_{id}) and each SU sense the K^{th} PU spectrum information and stored in a database at FC. SU transmits a bit '0' or bit '1' along with PU sensing information to the FC to indicate whether it needs a spectrum band to transmit the data or not. SU add two identification codes along with sensing information to the FC which indicates that from which SU the sensing information received and which PU is sensed by the SU. For simulation 500 data samples are used and the simulation results show an accuracy of 96% and false alarm value of 1.3% in classifying the SU sensing information at FC using RBF kernel. Another method is proposed with multiclass classification by applying SVM algorithm using RBF kernel. The confusion region class is classified with zero false alarm percentage and achieves an accuracy of 99.3% in classifying the SU sensing information at FC.

KEYWORDS: cooperative spectrum sensing, classification, fusion centre, kernel, support vector machine.

1. Introduction

In the late nineteenth century and at the beginning of twenty century major innovations in the field of telecommunication domain made a revolutionary change in this field. Placing satellites in the orbit, wireless cellular phones, major developments in the field of optical domain had taken telecommunication field to the next generation. Major research is focused on effective utilization of bandwidth, and techniques to access the available spectrum by large number of users. Digital modulation techniques like Quadrature Phase Shift Keying (QPSK), Gaussian Minimum Shift Keying (GMSK) and multiplexing techniques like Orthogonal Frequency Division Multiplexing (OFDM) improved the throughput data and accommodation of large number of users in a single channel respectively. Electromagnetic spectrum is a natural resource it cannot be created nor destroyed, so countries having huge population like India have to use the available spectrum resource in an effective way. In the year 2002, Federal Communication Commission (FCC), in the United States prepared a report [1] on utility of the available spectrum and published by Spectrum Policy Task Force, shows majority of spectrum band remains unutilized shown in Figure 1. Innovation of smart phones and Android Operating System (OS) made cellular phones to share information through various apps or applications than using for voice call. Information sharing can be a data, image, online streaming of video or on-demand entertainment channels etc., this made large bandwidth requirement. A report by the Telecom Regulation Authority of India (TRAI) [2], stated that there are more than 500 million mobile phone users in India and in future this number may increase. Technologies available at present may not support to handle this situation so countries like India, decided to stop unutilized services like telegraph from July 15, 2013 to meet out the spectrum demand. Further, researchers have been focusing towards developing techniques to utilize the unused spectrum effectively. In 1999, Mitola discussed the concept of the "pooled radio spectrum"[3] and in 2005, Haykin, [4] coined the phrase "spectrum hole" or white space which can be used by unlicensed users without affecting the licensed user and proposed the challenges that may arise in the realisation of this model. A decade of research, proposed several methods for spectrum sensing have been reviewed in [5,6], includes different types of narrow band sensing, wideband sensing, compressive sensing its performance measurement, applications with advantages and

disadvantages. In addition, the hardware challenges and different standards employed in sensing have also been discussed. In this paper, a cooperative spectrum sensing (CSS) model is proposed, with a fusion centre (FC) to detect the spectrum hole, when n-number of secondary users (SUs) trying to sense a n-number of primary users (PUs). A supervised machine learning (ML) algorithm namely, support vector machine (SVM), is used with 500 data samples taken as reference with "The Mobile Phone Activity Dataset" [7] to detect the spectrum hole. Finally, the SUs sensing information is classified at the FC, based on the kernel applied using SVM algorithm to identify the number of SUs sensed information having vacant spectrum band or it is occupied by PU.

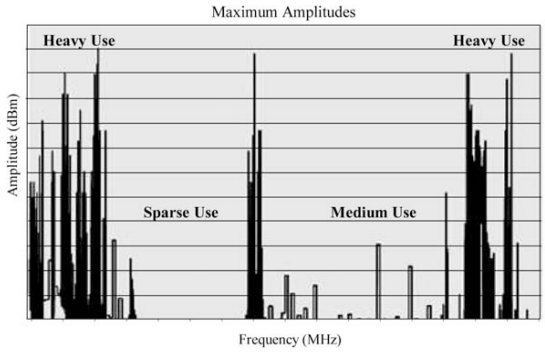

Fig.1 Spectrum utilisation chart report by Federal Communication Commission

The remaining of this paper is organised as follows: Section 2 provides an overview of the existing CSS methods and ML techniques used in spectrum sensing. It also describes the proposed CSS model, which is based on SVM algorithm. Section 3 proposes a multiclass classification using SVM to overcome multi-threshold problem in spectrum sensing. Section 4 compares the performances of the two proposed models discussed in previous chapters and Section 5 presents the simulation results along with the performance metric curves. Finally, Section 6 presents the conclusion of the work.

2. Cooperative spectrum sensing (CSS) model

2.1. Framework of CSS model

A conventional CSS model [8] is shown in Figure 2. The model has one PU transmitter (PU_{TX}), one PU receiver (PU_{RX}), one or more SUs or Cognitive Radio (CR) users used to sense the PU_{TX}, and a FC to receive the sensing information from all SUs. The CR users sense the transmission from the PU_{TX} at every instant of time and pass this information to the FC. The FC receives the sensing information from all the CR users through the control channel and apply mathematical model to determines the presence or absence of PU_{TX}. This information is shared to all SUs and FC will decide which SU is allowed to utilise the available spectrum band. This type of decision is called hard combination. In another method called soft combination where all SUs take decision by itself and convey the result to the FC. In the system model, if the transmitted signal by PU_{TX} is considered as x(t), and the received signal at each SU is considered as y(t) with h(t) as channel gain of the sensing channel and n(t) as zero mean additive white gaussian noise. Then the system identifying the presence of spectrum hole is formulated as binary hypothesis problem as given by equation 1 as

$$H_0 : y[t] = n[t] - - - - - - - - 1$$

$$H_1 : y[t] = h[t] * s[t] + n[t] - - - 2$$

Here, H_0 and H_1 indicates the PU_{TX} spectrum band is vacant and occupied by PU_{RX}, respectively.

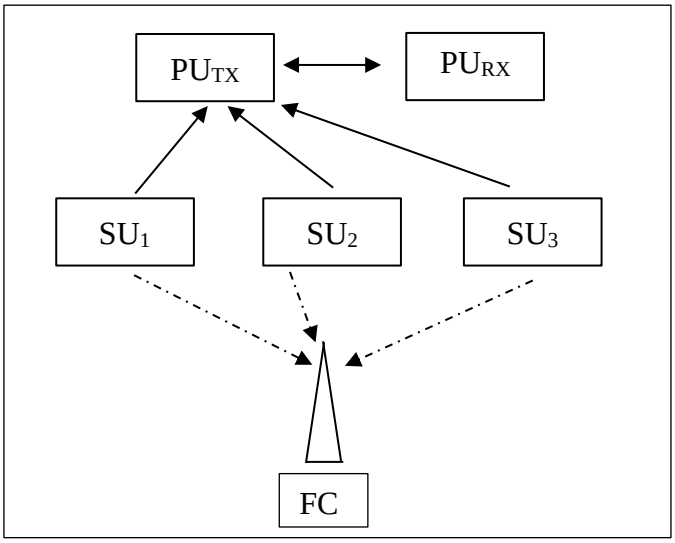

Fig.2 Conventional cooperative sensing model.

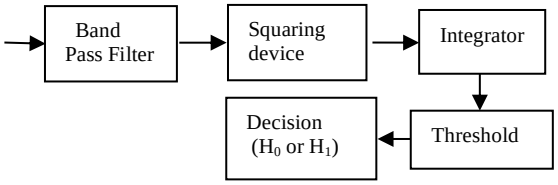

Fig.3 Conventional energy detection technique

The energy detection technique is an efficient method to detect the spectrum hole without any prior knowledge about the PU_{TX}. In

this method PU_{TX} signal power is measured with a threshold value and if the sensed signal power is less than the threshold, then PU_{TX} spectrum band is considered as vacant, and if it is greater than the threshold then it is occupied by PU_{RX}. The block diagram of this method is shown in Figure 3. This method is further improved using double threshold and eigen value based double threshold and Sevcik Fractal Dimension. The dynamic threshold method, energy-efficient spectrum sensing, and a few other advanced methods for CSS have been discussed in [9,10,11,12,13].

Previous work related to CSS using machine learning (ML) [14] where the input signal is classified into predefined classes and H_0 hypothesis on the PU received signal is divided into k discrete regions by using k reliability thresholds. Residual energy of the SU node battery is classified into discrete regions based on the threshold. The result shows increase in throughput when compared to two class hypothesis model. In [15], a blind spectrum sensing scenario is performed using deep learning method. In this method three types of neural networks, namely, a convolutional neural network, a long short-term memory, a fully connected neural network is applied. Sensing was performed in the region of -9 dB to -5 dB, and the false alarm rate was 0.1. In [16], CSS based on soft combination and investigation of the effective channel gain between SU and FC is considered. Optimal weight combining is obtained from linear detection probability and to increase the sensing performance optimal power allocation is obtained by maximising the optimal weight. In [17] k-nearest neighbour algorithm is used for spectrum sensing. The SUs sensed the PUs under varying conditions and send a signal to the FC. FC decides the PU is having a vacant spectrum band or not based on the sensing information received from all SUs and applying a hard combination decision rule. The above-descried methods show how ML is used to sense a spectrum hole in a single PU and maximising the optimal weight increase the sensing performance.

2.2 Proposed cooperative spectrum sensing model using SVM.

The proposed system model is an extension of our previous work [18]. Figure 4 contains n-number of primary user transmitter (PUs), FC and n-number of SUs. All n-number of SUs sense the k^{th} PU for the availability of vacant spectrum band. Assume in our system model, FC is aware about the number of PUs present in the cell structure, and it is allocated with a three-bit identification code to distinguish with each PU. FC share this PU identification code (PU_{id}) to all SUs using control channel and instruct all SUs through control channel, which PU to be sensed for vacant spectrum band. In this scenario, only a few SUs may be looking for vacant spectrum band to transmit the data and few other SUs may not require any channel due to non-availability of data to transmit or few other SUs may be utilizing other PU vacant spectrum band present in the same cell

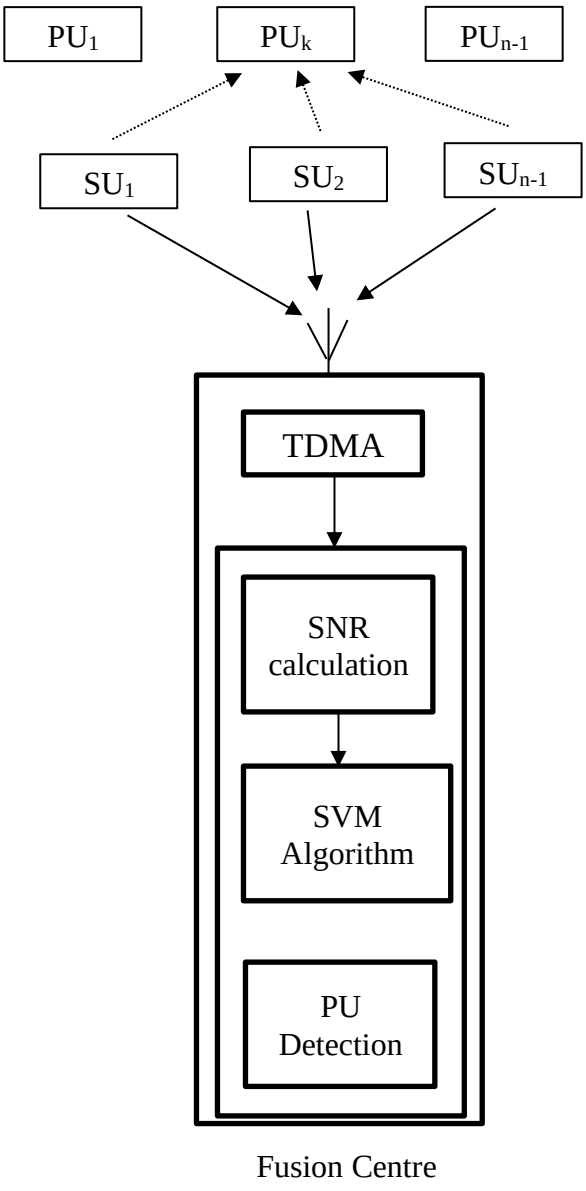

Fig.4 Proposed system model using SVM at fusion centre

region to transmit the data. Based on this scenario, the remaining n-number of SUs sense the k^{th} PU for availability of vacant spectrum band and pass the sensing information to the FC through dedicated

channel. In this model a soft decision fusion model is performed, so all the n-number of SUs will forward the sensing information to the FC. Few SUs out of n-number of SUs, who doesn't need a spectrum band will intimate the FC by transmitting a bit '0' and if needed a bit '1' through control channel. The remaining n-number of SUs sense the k^{th} PU spectrum band to identify whether k^{th} PU is vacant or busy and it will transmit the sensing information to FC along with two identification codes as SU_{id}, and PU_{id}, along with the SUs sensing data in this way the SUs providing wrong information about PU status can be eliminated to improve the system performance. Now FC will apply SVM to all the n-SUs sensing information collected and stored as a database containing SU_{id}, PU_{id}, SNR, distance and Threshold as attributes. Figure 5 shows the block diagram of the proposed model, where the n-number of SUs sensing information is received by the FC at different time interval using Time Division Multiple Access (TDMA), further it is demodulated, converted from frequency domain to time domain to calculate the signal power of k^{th} PU from various SUs transmitted information. Pre-threshold value is added to the database will get updated based on the present SU data, because signal strength values may change depending on the environmental conditions. Further SVM is applied to the database to classify how many SUs sensed information conveying that PU is vacant or busy.

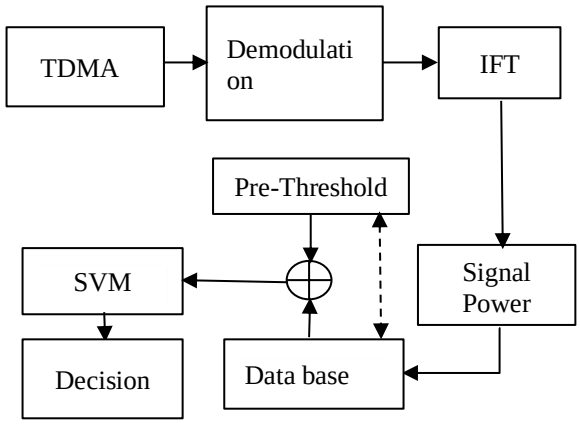

Fig.5 Proposed system block diagram of spectrum sensing using SVM

The n-number of SUs sensing the k^{th} PU at different time intervals, so the entire sensing time (ST) is divided into ST_1, ST_2, ... ST_n with respect to each SUs. Each SU sensing information is transmitted to the FC, so the sensing time is divided into two sub-slots as sensing time (st) and transmission time (tt) as shown in figure 6. Transmission time will include the three-bit PU_{id}, so if PU_{id} is different than the FC instructed through control channel, FC can discard this SU sensing information. Since soft decision is used in our system model, all n-number of SUs transmit the sensed information without making any local decision and FC will take the final decision of k^{th} PU spectrum band is vacant or busy. The sensed information received at the FC from n-number of SUs is

$$y_i[t] = \sum_{i=1}^{n} \sqrt{\rho} h_i[t] \cdot x_i[t] + \varphi_i[t]$$

Where $y_i[t]$= received signal from each SU to FC
$x_i[t]$= transmitted information about k^{th} PU by each SU
$h_i[t]$= channel gain
$\varphi_i[t]$= gaussian noise
ρ= signal to noise ratio

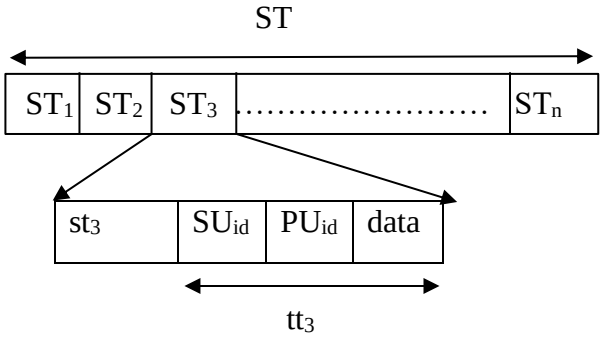

Fig.6 Sensing and transmitted signal duration frame slot

In the proposed system model PU spectrum may be present or absent in the received signal by the k^{th} SU ($\forall\ k = 1,2,3,\ldots..n$). The method is partly based on that described in [8]. The sensed signal by the n-number of SUs is given by

$y_i[t] = \varphi_i[t]\ \forall\ t = 1,2,\ldots.L$

$y_i[t] = \sqrt{\rho} h_i[t] \cdot x_i[t] + \varphi_i[t] \forall\ t = 1,2,\ldots..L$

Where L is the total number of samples received during the sensing time st_1 and ρ is the signal to noise ratio value. The channel is assumed to have flat fading and its gain $h_i[t]$ depends on the distance between PU and each SU and it is given by $h_i[t] \sim CG(0, d_a^{-\alpha})$ where d_a= distance between k^{th} PU and each SU, α= path loss component. PU samples $x_i[t]$ is mutually independent, that is to say, $x_i[k] \neq x_i[l]$ when $k \neq l$ and the noise $\varphi_i[t]$ have zero mean and variance $E[|\varphi_i(t)|^2] = P_n$. The k^{th} PU energy is calculated by FC as $S_e = \sum_{i=1}^{n} |y_i(t)|^2$ and it is stored in the database. The values of S_e and pre-threshold value λ are

considered as input features and it is taken as coordinates of vector **x.** The estimated energy S_e is compared with a pre-threshold value λ, to detect the presence of spectrum hole in k^{th} PU. The direction of the vector **x**=(S_e,λ) that map the values of its coordinates is taken as weight vector **w** and b is considered as bias. In general the plane that separates the two dimensional data is **y** = a**x**+b, and taking $\boldsymbol{x}$ =(x_1,x_2) and $\boldsymbol{w}$ =(a,-1) so the equation becomes

$$\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$$

The plane that separates PU spectrum band is vacant or busy is formed as a hypothesis problem given by

$\boldsymbol{w} \cdot \boldsymbol{x} + b < 0$ as class -1 where PU is busy

$\boldsymbol{w} \cdot \boldsymbol{x} + b \geq 0$ as class +1 where PU is vacant

In order to select the best plane to separate the two classes, an optimization is done by adding a slack variable to the constraints

$y_i(\boldsymbol{w} \cdot \boldsymbol{x_i} + b) \geq 1 - \sigma_i$ where i = 1,2,….n

and L_1 regularisation is applied. A regularization variable R is added for optimization and the equation becomes

$\min_{w,b,\sigma} \frac{1}{2}\|w\|^2 + R\sum_{i=1}^{n}\sigma_i$

subject to $y_i(\boldsymbol{w} \cdot \boldsymbol{x_i} + b) \geq 1 - \sigma_i$, $\sigma_i > 0$,

where i = 1,2,…n.

A Lagrange multiplier β is applied to convert the optimization problem into a dual problem

$\max_{\beta} \sum_{i=1}^{n}\beta_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\beta_i\beta_j\, \boldsymbol{y}_i\boldsymbol{y}_j\boldsymbol{x}_i \cdot \boldsymbol{x}_j$

subject to $0 \leq \beta_i \leq R$, i=1,2,… n, $\sum_{i=1}^{n} a_i\, \boldsymbol{y}_i = 0$

where β_i lies between 0 to R. To solve this optimization problem, consider only the product of two variables $\boldsymbol{x}_i \cdot \boldsymbol{x}_j$in the above equation. The dot product is considered as kernel, which is used to classify the sensed signal data transmitted by each SUs. In the proposed model linear, polynomial and Radial Basis Function (RBF) kernel are applied to the SUs transmitted information to classify the PU spectrum is vacant or busy. Linear kernel is the simplest kernel function and the inner product is given by

$$k(\boldsymbol{x_i} \cdot \boldsymbol{x_j}) = \boldsymbol{x_i^T x_j} + c$$

Polynomial kernel function is given by

$$k(\boldsymbol{x}_i \cdot \boldsymbol{x}_j) = (\alpha \boldsymbol{x}_i \cdot \boldsymbol{x}_j + c)^d$$

RBF kernel function is given by

$$k(\boldsymbol{x}_i \cdot \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x_i} - \boldsymbol{x_j}\|^2}{2\alpha^2}\right)$$

3. Multi-class classification using support vector machine.

The proposed CSS system model is shown in figure 7, where n-number of PU, n-number of SUs and a FC is present. All SUs sense k^{th} PU for a vacant spectrum band and transmit the sensed information to the FC through a dedicated channel. FC receive all sensed information from n-number of SUs at different timeslots using Time Division Multiple Access (TDMA), it is further demodulated and converted to time domain using Inverse Fourier Transform (IFT) to estimate the signal power information transmitted by each SUs.

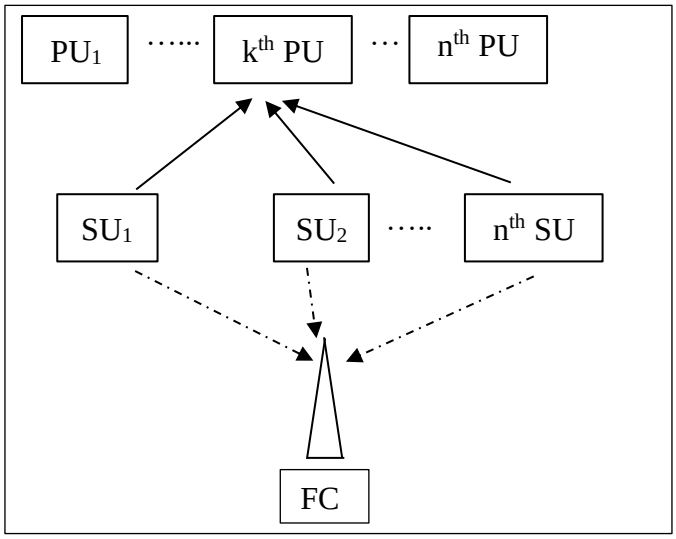

Fig.7 Proposed system model using multi-class SVM

All SUs sensed information is stored in a database as shown in figure 8. The multiclass pre-threshold value is updated based on the present dataset values this would increase the detection accuracy. Different SUs is located at different locations in a sensing zone, the distance between each SU with k^{th} PU may vary. Those SUs which are located at farthest distance from k^{th} PU may have weak signal strength and the sensed information may be false due to fading. FC have to decide whether the k^{th} PU, have a vacant spectrum band, from the n-number SUs sensed information. FC apply SVM to all the n-number of SUs sensed dataset to classify the SUs sensed information into three different class whether the PU have vacant spectrum band, number of SUs conveying PU is busy and few PUs SNR range value falls very close to the threshold region so it's difficult to decide due to channel fading and the number of SU may misguide the FC due to non-availability of data to transmit. Multiclass SVM use one-against-all

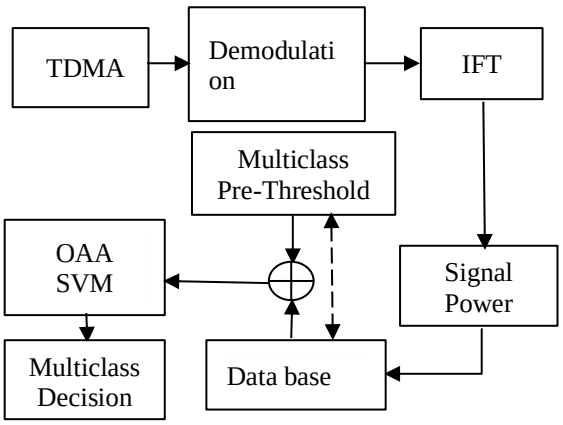

Fig.8 Proposed system block diagram using multi-class SVM

(OAA) to construct 'k' binary SVM classifiers, where 'k' is the number of classes with 'k' decision functions $w_1^T(t)\varphi(x_i) + b_1, \ldots\ldots, w_k^T(t)\varphi(x_i) + b_k$. The final output is given to the class that has highest output value of

$x \equiv argmax_{i=1,\ldots.k}(w_i^T(t)\varphi(x) + b_i$

In this model three types of classes is formed from n-number of SUs sensing information.

1. PU_busy
2. Confusion_region
3. PU_vacant

4. Performance analysis of proposed methods

4.1 Spectrum sensing using SVM.

ML is applied to predict the future output from the collected dataset, which will make statistical decision making and used to solve data mining problems. In this paper a supervised ML algorithm called SVM is applied, where the dataset is divided into training and test set to predict the future values of the system as shown in figure 9. A mobile data set [19] is used as a reference to analyse our system performance model. The transmitted signal from the k^{th} PU is sensed by the n-number of SUs as discussed in chapter 2.2. and the signal power is calculated at the FC and stored in a database. The attributes considered to classify the SUs are signal to noise ratio (SNR), PU_{id}, SU_{id} the pre-threshold value is updated from the current database and applied to SVM algorithm. SVM algorithm split the dataset samples into training set and test set and classify the sensed information of training and test samples into following four different categories as

a. True Positive (TP)
b. False Positive (FP)
c. False Negative (FN)
d. True negative (TN)

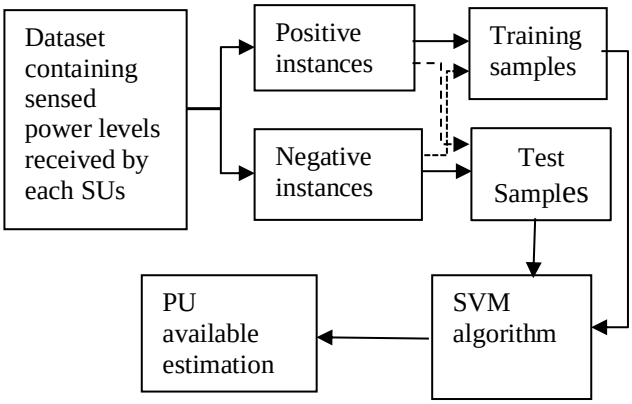

Fig.9 Supervised ML flow diagram

Table 1. Confusion matrices for 125 test samples

Algorithm: SVM Confusion Matrix		
125 Test Samples		
		True /Actual
Predicted		Positive	Negative
	Positive	44	7
	Negative	1	73

These four different categories form a confusion matrix as given in Table 1. In the proposed method FC have to decide whether PU spectrum band is occupied or vacant. So in the confusion matrix TP is considered as PU busy and TN region is considered as PU vacant. FP region indicates that those PU who is actually busy is wrongly predicted as vacant and FN region indicates PU actually vacant is predicted as busy. In this method FC has to identify number of SU sensed signal conveys PU is busy, so TP and FP region is given importance in the confusion matrix, since if a PU is vacant is wrongly predicted as busy it will not affect the system performance but if a PU is actually occupied but wrongly predicted as vacant by FC and if the SU starts transmitting the information then the PU system performance will be affected. Table 2 gives the confusion matrix formed using linear kernel and Table 3 gives the false alarm rate and percentage of detection using linear kernel. Similarly, Table 4 and Table 6 gives the confusion matrix formed using polynomial kernel and RBF kernel followed by Table 5 and Table 7 gives the false alarm rate and percentage of detection using polynomial kernel and RBF kernel respectively.

Table 2. Confusion matrix for SVM algorithm using linear kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	2	-	-	1
2	20	15	5	0	1	-	-	4
3	30	22	8	0	2	-	-	6
4	40	30	10	0	6	-	-	2
5	50	37	13	0	6	-	-	7
6	60	45	15	0	7	-	-	8
7	70	52	18	0	8	-	-	10
8	80	60	20	0	13	-	-	7
9	90	67	23	0	13	-	-	10
10	100	75	25	0	14	-	-	11
11	110	82	28	0	15	-	-	13
12	120	90	30	0	14	-	-	16
13	150	112	38	2	16	2	-	20
14	200	150	50	4	18	3	1	28
15	250	187	63	4	24	3	1	35
16	300	225	75	3	31	1	2	41
17	350	262	88	7	27	6	1	54
18	400	300	100	10	33	8	2	57
19	450	337	113	7	42	5	2	64
20	500	375	125	8	44	7	1	73

Table 3. False alarm rate and detection accuracy for different number of test samples using linear kernel

S.No	Number of PUs as test samples	Percentage of false alarm	Percentage of detection	
		PUs busy	PUs available	PUs busy
1	3	nil	66.6	33.3
2	5	nil	20	80
3	8	nil	25	75
4	10	nil	60	40
5	13	nil	46.1	53.3
6	15	nil	46.6	53.3
7	18	nil	44.4	55.5
8	20	nil	65	35
9	23	nil	56.6	43.3
10	25	nil	56	44
11	28	nil	53.5	46.4
12	30	nil	46.6	53.3
13	38	5.2	42.1	52.6
14	50	6	36	56
15	63	4.7	38	55.5
16	75	1.3	41.3	54.6
17	88	6.8	30.6	61.3
18	100	8	33	57
19	113	4.4	37.1	56.6
20	125	5.6	35.2	58.4

Table 4. Confusion matrix for SVM algorithm using polynomial kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	2	-	-	1
2	20	15	5	0	1	-	2	2
3	30	22	8	0	2	-	2	4
4	40	30	10	0	2	1	3	4
5	50	37	13	0	6	-	7	0
6	60	45	15	0	7	-	4	4
7	70	52	18	0	8	-	7	3
8	80	60	20	0	13	-	2	5
9	90	67	23	0	13	-	2	8
10	100	75	25	0	14	-	5	6
11	110	82	28	0	15	-	4	9
12	120	90	30	0	14	-	4	12
13	150	112	38	9	9	9	-	20
14	200	150	50	6	15	6	-	29
15	250	187	63	13	14	13	2	34
16	300	225	75	12	20	12	2	41
17	350	262	88	17	16	17	2	53
18	400	300	100	10	31	10	1	58
19	450	337	113	15	32	15	-	66
20	500	375	125	39	12	39	11	63

Table 5. False alarm rate and detection accuracy for different number of test samples using polynomial kernel

S.No	Number of PUs as test samples	Percentage of false alarms	Percentage of detection	
		PUs busy	PUs available	PUs busy
1	3	nil	66.6	33.3
2	5	nil	20	40
3	8	nil	25	50
4	10	10	20	40
5	13	nil	46.1	nil
6	15	nil	46.6	26.6
7	18	nil	44.4	16.6
8	20	nil	65	25
9	23	nil	56.5	34.7
10	25	nil	56	24
11	28	nil	53.5	32.1
12	30	nil	35	40
13	38	23.6	23.6	52.6
14	50	12	30	58
15	63	20.6	22.2	53.9
16	75	16	26.6	54.6
17	88	2.2	18.1	60.2
18	100	1	31	58
19	113	13.2	28.3	58.4
20	125	31.2	9.6	50.4

Table 6. Confusion matrix for SVM algorithm using RBF kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	2	-	-	1
2	20	15	5	0	1	-	1	3
3	30	22	8	0	2	-	-	6
4	40	30	10	0	6	-	-	4
5	50	37	13	0	5	1	-	7
6	60	45	15	0	6	1	-	8
7	70	52	18	0	8	-	-	10
8	80	60	20	0	13	-	-	7
9	90	67	23	0	13	-	-	10
10	100	75	25	0	14	-	-	11
11	110	82	28	0	15	-	-	13
12	120	90	30	0	14	-	-	16
13	150	112	38	2	16	2	-	20
14	200	150	50	3	18	3	-	29
15	250	187	63	3	24	3	2	34
16	300	225	75	1	31	1	2	41
17	350	262	88	7	26	7	2	53
18	400	300	100	7	34	7	3	56
19	450	337	113	7	40	7	2	64
20	500	375	125	7	44	7	1	73

Table 7. False alarm rate and detection accuracy for different number of test samples using RBF kernel

S.No	Number of PUs as test samples	Percentage of false alarms	Percentage of detection	
		PUs busy	PUs available	PUs busy
1	3	nil	66.6	33.3
2	5	nil	20	60
3	8	nil	25	75
4	10	nil	60	40
5	13	7.6	38.4	53.8
6	15	6.6	40	53.3
7	18	nil	44.4	55.5
8	20	nil	65	35
9	23	nil	56.5	43.4
10	25	nil	56	44
11	28	nil	53.5	46.4
12	30	nil	46.6	53.3
13	38	5.2	42.1	52.6
14	50	6	36	58
15	63	4.7	38	53.9
16	75	1.3	41.3	54.6
17	88	7.9	29.5	60.2
18	100	7	34	56
19	113	6.1	35.3	56.6
20	125	5.6	35.2	58.4

4.2 Performance analysis of multiclass classification using SVM.

The system model shown in chapter 3, form a multiclass classification problem applying SVM algorithm using RBF kernel to identify the number of SUs sensed information conveying into three classes as PU busy, confusion region and PU vacant. Here a new class called confusion region is introduced where the PU SNR value is very close to the threshold value or the SNR of PU is equal to threshold value it is classified into confusion region, in this way the PU is actually busy is wrongly predicted as vacant which increases the percentage of false alarm can be eliminated. Table

8 gives the confusion matrix of (3,3) for a three-class classification problem, in which if TP is considered as PU busy, then matrix position (1,1) is taken as TP value, TN values are in the matrix position (2,2), (2,3), (3,2) and (3,3), FN region indicating PU actually busy is wrongly predicted as vacant is given by matrix position (2,1) and (3,1), and FP region indicating PU actually vacant is wrongly predicted as busy is given by matrix positions (1,2) and (1,3) respectively. In this classification method as it is mentioned in chapter 4.1, FC ability to classify PU busy as TP region and FN region where it is actually busy, but wrongly predicted as vacant which will affect the system performance with increasing the false alarm percentage. Therefore, TP and TN are given higher priority than FP and FN region.

Table 8. Confusion matrix for PU busy class using multiclass SVM algorithm using RBF kernel

Algorithm: SVM				
500 data samples and 150 Test Samples				
		True/Actual		
		PU busy	Confusion region	PU vacant
Predicted	PU busy	55	1	0
	Confusion region	0	51	0
	PU vacant	1	0	42

Table 9. Confusion matrix for PU busy class using multiclass SVM algorithm with RBF kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	1	-	-	2
2	20	14	6	0	3	-	-	3
3	30	21	9	0	3	-	-	6
4	40	28	12	0	3	-	-	9
5	50	35	15	0	5	-	-	10
6	60	42	18	0	7	-	-	11
7	70	49	21	1	10	1	-	10
8	80	56	24	1	7	1	-	16
9	90	63	27	0	7	-	-	20
10	100	70	30	1	11	-	1	18
11	150	105	45	1	16	-	1	28
12	200	140	60	1	26	1	-	33
13	300	210	90	4	33	3	1	53
14	400	280	120	4	43	3	1	73
15	500	350	150	2	55	1	1	93

Table 10. False alarm rate and detection accuracy for PU busy class with different number of test samples using RBF kernel.

S.No	Number of PUs as test samples	Percentage of false alarms	Percentage of detection	
		SUs busy	SUs available	Other classes
1	3	nil	33.3	66.6
2	6	nil	50	50
3	9	nil	33.3	66.6
4	12	nil	25	75
5	15	nil	33.3	66.6
6	18	nil	38.8	61.1
7	21	4.7	47.6	47.6
8	24	4.1	29.1	66.6
9	27	nil	25.9	74
10	30	nil	36.6	60
11	45	nil	35.5	62.2
12	60	1.6	43.3	55
13	90	3.3	36.6	58.8
14	120	2.5	35.8	60.8
15	150	0.6	36.6	62

Table 9, Table 11 and Table 13 shows the confusion matrix formed using multiclass SVM algorithm with RBF kernel for three different multiclass regions namely PU busy class, confusion region, PU vacant class. Table 10, Table 12, Table 14 gives the percentage of false alarm and percentage of detection for the multiclass regions. From Table 13 the objective to classify the confusion region class without any false alarm is achieved since the value obtained is nil and the other two class regions may have few percentages of false alarm but the purpose of multiclass classification is to avoid the PU whose SNR ranges fall near to the threshold value and considering

those PUs to detect the vacant spectrum band may increase the false alarm rate. So, confusion region is classified without any false alarm indicates that system performance is improved by applying multiclass classification.

Table 11. Confusion matrix for confusion region class using multiclass SVM algorithm with RBF kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	1	-	-	2
2	20	14	6	0	3	-	-	3
3	30	21	9	0	1	-	-	8
4	40	28	12	0	2	-	-	10
5	50	35	15	0	5	-	-	10
6	60	42	18	0	3	-	-	15
7	70	49	21	1	3	-	1	17
8	80	56	24	1	8	-	1	15
9	90	63	27	0	8	-	-	19
10	100	70	30	0	8	-	-	22
11	150	105	45	0	13	-	-	32
12	200	140	60	0	21	-	-	39
13	300	210	90	0	33	-	-	57
14	400	280	120	2	35	-	2	83
15	500	350	150	1	51	-	1	98

Table 12. False alarm rate and detection accuracy for PU confusion region with different number of test samples using RBF kernel

S.No	Number of PUs as test samples	Percentage of false alarms	Percentage of detection	
		Confusion_region	Confusion_region	Other classes
1	3	nil	33.3	66.6
2	6	nil	50	50
3	9	nil	11.1	88.8
4	12	nil	16.6	83.3
5	15	nil	33.3	66.6
6	18	nil	16.6	83.3
7	21	nil	14.2	80.9
8	24	nil	33.3	62.5
9	27	nil	29.6	70.3
10	30	nil	26.6	73.3
11	45	nil	28.8	71.1
12	60	nil	35	65
13	90	nil	36.6	63.3
14	120	nil	29.1	69.1
15	150	nil	34	65.3

Table 13. Confusion matrix for PU vacant class using multiclass SVM algorithm with RBF kernel for different number of data samples.

S. No	Number of SUs	Number of training and test samples		Incorrectly classified test samples	TP	FP	FN	TN
1	10	7	3	0	1	-	-	2
2	20	14	6	0	-	-	-	6
3	30	21	9	0	5	-	-	4
4	40	28	12	0	7	-	-	5
5	50	35	15	0	5	-	-	10
6	60	42	18	0	8	-	-	10
7	70	49	21	0	7	-	-	14
8	80	56	24	0	8	-	-	16
9	90	63	27	0	12	-	-	15
10	100	70	30	1	10	1	-	19
11	150	105	45	1	15	1	-	29
12	200	140	60	1	12	-	1	47
13	300	210	90	4	20	1	3	66
14	400	280	120	1	38	1	1	80
15	500	350	150	1	42	1	-	107

Table 14. False alarm rate and detection accuracy for PU vacant class with different number of test samples using RBF kernel

S.No	Number of PUs as test samples	Percentage of false alarms	Percentage of detection	
		SU vacant	SUs vacant	Other classes
1	3	nil	33.3	66.6
2	6	nil	nil	100
3	9	nil	55.5	44.4
4	12	nil	58.3	41.6
5	15	nil	33.3	66.6
6	18	nil	44.4	55.5

7	21	nil	33.3	66.6
8	24	nil	33.3	66.6
9	27	nil	44.4	55.5
10	30	3.3	33.3	63.3
11	45	2.2	33.3	65.4
12	60	nil	20	78.3
13	90	1.1	22.2	73.3
14	120	0.8	31.6	66.6
15	150	0.6	28	71.3

5. Simulation results

In this section, the proposed method of CSS when n-number of SUs sense the k^{th} PU for vacant channel and transmit the information to FC where SVM algorithm is used to classify the SUs sensed data to identify the k^{th} PU is vacant or busy is simulated using python programming and the graphs of the performance metrics mentioned in [20], is discussed here.

5.1 Results using SVM algorithm

The performance of SVM algorithm, using three different kernels namely linear, polynomial and RBF to classify the n-SUs sensed information at FC is processed using training and test data samples ranging from 10 to 500. Test data simulation results alone discussed in this section. Data samples of 200,400 and 500 with corresponding test data samples of 50, 100 and 125 for all the three different kernels applied using python programming to classify the SUs information is shown from figure 10,11,12,13,14, 15,16,17 and 18.

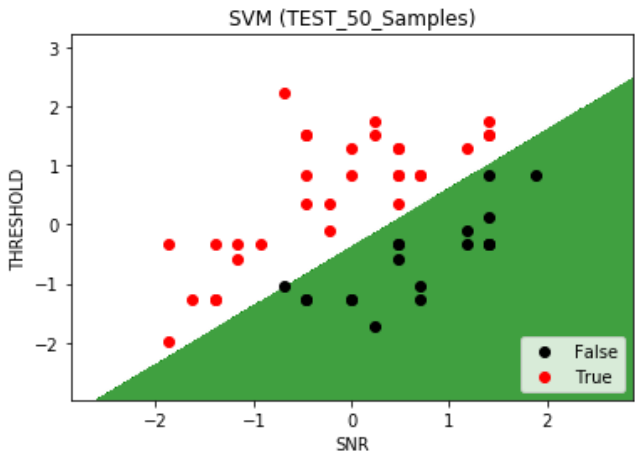

Fig.10 SVM classification for 50 SUs as test samples using linear kernel

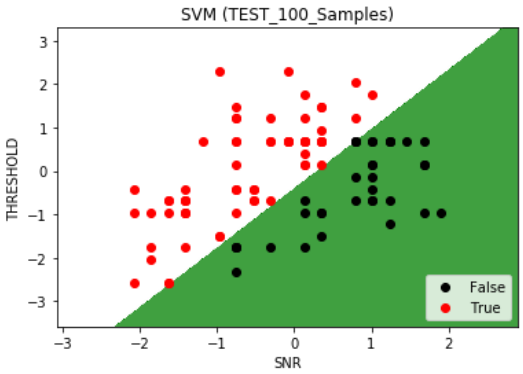

Fig.11 SVM classification for 100 SUs as test samples using linear kernel

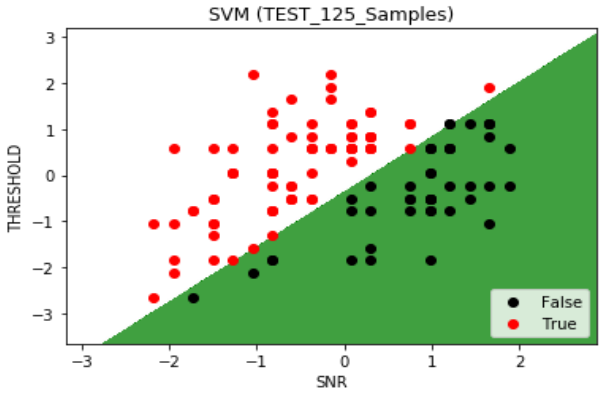

Fig. 12 SVM classification for 125 SUs as test samples using linear kernel

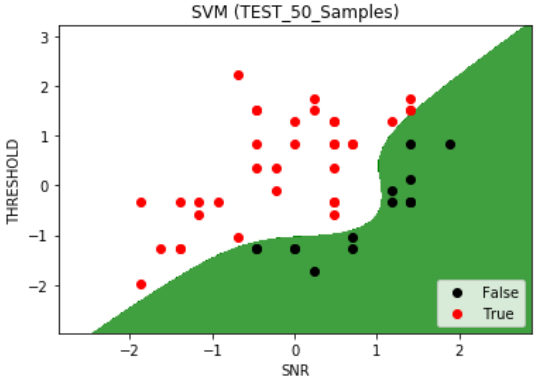

Fig.13 SVM classification for 50 SUs as test samples using polynomial kernel

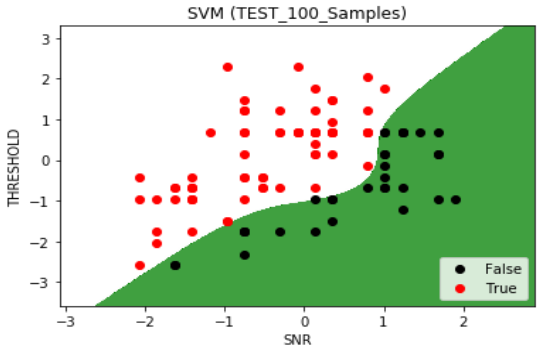

Fig.14 SVM classification for 50 SUs as test samples using polynomial kernel

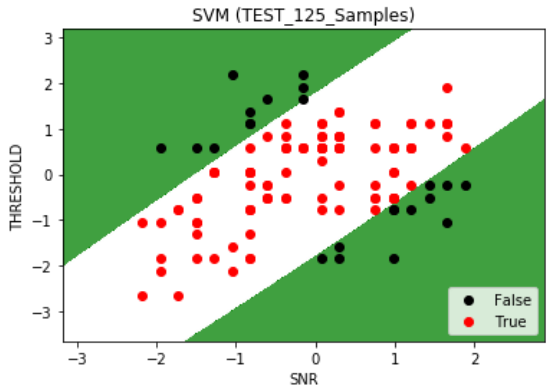

Fig.15 SVM classification for 125 SUs as test samples using polynomial kernel

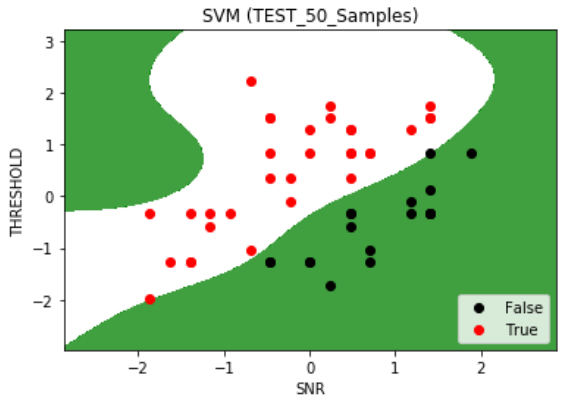

Fig.16 SVM classification for 50 SUs as test samples using RBF kernel

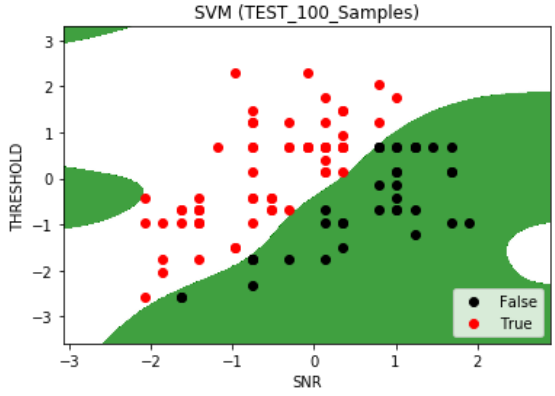

Fig.17 SVM classification for 100 SUs as test samples using RBF kernel

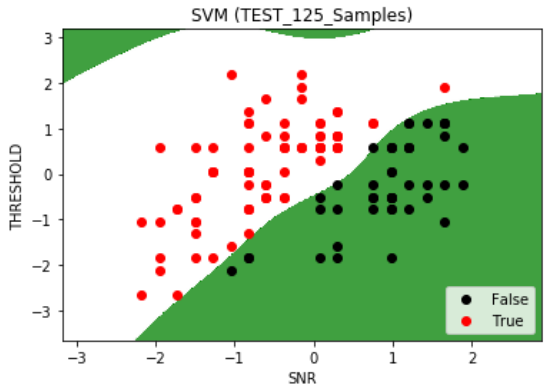

Fig.18 SVM classification for 125 SUs as test samples using RBF kernel

Table 15. Accuracy of SVM algorithm using different kernels to classify the n-number of SU test samples

S.No	Number of SUs test samples	Linear kernel	Polynomial kernel	RBF kernel
1	3	100	100	100
2	5	100	60	80
3	8	100	75	100
4	10	100	60	100
5	13	100	46.1	92.3
6	15	100	73.3	93.3
7	18	100	61.1	100
8	20	100	90	100
9	23	100	91.3	100
10	25	100	80	100
11	28	100	85.7	100
12	30	100	86.6	100
13	38	94.7	76.3	94.7
14	50	92	88	94
15	63	93.6	76.1	92
16	75	96	81.3	96
17	88	92	78.4	89.7
18	100	90	89	90
19	113	93.8	86.7	92
20	125	93.6	60	93.6

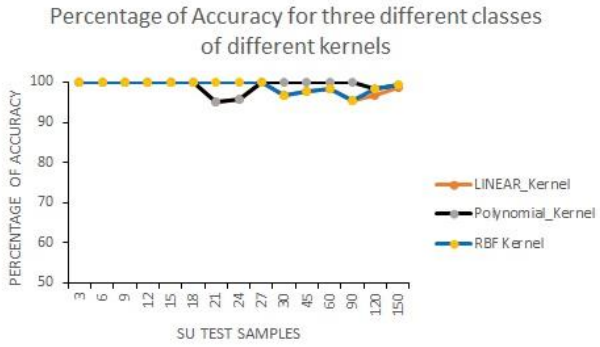

Fig.19 Percentage of accuracy of three different kernels to classify n-number of SUs by the fusion centre

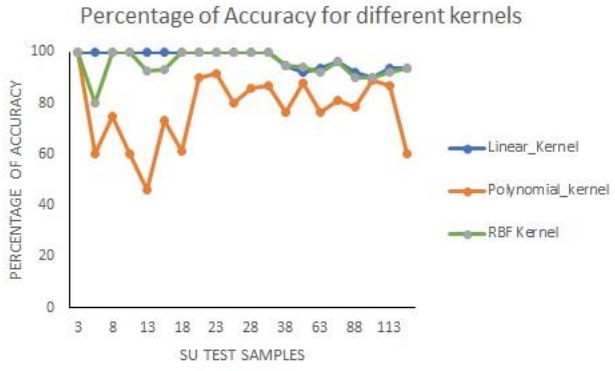

Fig.20 Percentage of false alarm of three different kernels to classify n-number of SUs sensing information by the fusion centre

Table 15 shows the accuracy of predicting the PU busy class using three different kernels as mentioned in chapter 2. From the three different

kernels applied, even though linear kernel shows better performance but most of the time the system will not behave linearly, so RBF kernel shows better performance when compared with polynomial kernel with maximum accuracy value of 96% for 300 data samples with corresponding 75 test data samples. Figure 19 shows percentage of accuracy in classifying PU busy class by three different kernels. Graph shows few test samples are classified with 100% accuracy for both linear and RBF kernel and when test samples are increased to 125 samples, accuracy value is reduced from 96% to 93.6% using RBF kernel. Figure 20 shows percentage of false alarm using three different kernels, few test samples are classified without any misclassification using RBF kernel and for 75 test samples, a false alarm value of 1.3% is obtained and increased to 5.6% for 125 test samples.

5.2 Results using multiclass SVM algorithm

The proposed method in chapter 3 using multiclass classification is implemented using python programming and the accuracy of predicting PU busy class, confusion region and PU vacant class is shown in Table 16. Result shows using RBF kernel a maximum accuracy of 98.6% is achieved to classify PU busy class whereas confusion region and PU vacant class is classified with a maximum accuracy of 99.3% for both having 500 data samples with corresponding test data of 150 samples respectively. PU confusion region has a minimum accuracy of 95.2% for 70 data samples with corresponding 21 test data whereas the PU vacant class has a minimum accuracy of 96.6% for 100 data samples with corresponding test samples of 30.

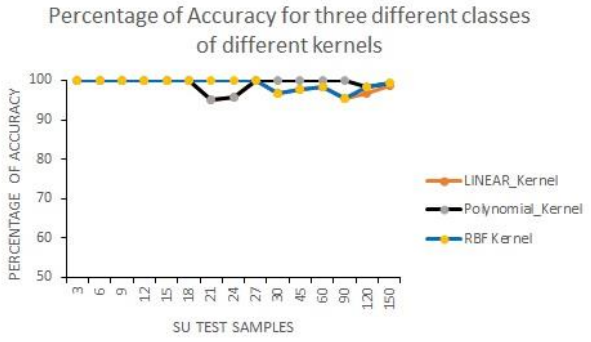

Fig.21 Percentage of accuracy for three different classes of different kernels to classify the SUs by the fusion centre.

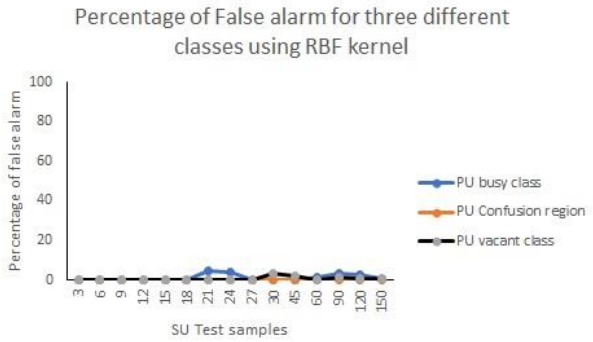

Fig.22 Percentage of false alarm for three different classes to classify the SUs by using RBF kernel

Table 16. Accuracy of SVM algorithm for different classes of test samples

S.No	Number of SUs test samples	PU busy Accuracy	Confusion region Accuracy	PU vacant Accuracy
1	3	100	100	100
2	6	100	100	100
3	9	100	100	100
4	12	100	100	100
5	15	100	100	100
6	18	100	100	100
7	21	95.2	95.2	100
8	24	95.8	95.8	100
9	27	100	100	100
10	30	96.6	100	96.6
11	45	97.7	100	97.7
12	60	98.3	100	98.3
13	90	95.5	100	95.5
14	120	96.6	98.3	98.3
15	150	98.6	99.3	99.3

6. Conclusion

The proposed CSS method to sense n-number of PUs by n-number of SUs in a sequence by applying SVM algorithm using linear, polynomial and RBF kernel. Simulation results show that RBF kernel has achieved 93.6% accuracy for a maximum of 500 data samples with corresponding 125 test data samples. Even though few data samples have 100% accuracy but a maximum of 96% accuracy is achieved for 300 data samples with corresponding 75 test data samples. By increasing the number of data samples beyond 300 accuracy in classifying the SU sensing information falls to 93.6%. Another proposed method with multiclass classification by applying SVM algorithm using RBF kernel has achieved zero false alarm percentage for the confusion region class. This indicates that by using multiclass classification, those PUs whose SNR values very close to the threshold value is classified 100% which will increase the system performance. The

percentage of accuracy for PU busy class is 98.6%, confusion region class is 99.3% and PU vacant class is 99.3%. This work can be extended to scheduling the identified spectrum hole from different PUs to different SUs in a CSS system model.

References

[1]. Marcus M, Burtle J, Franca B, Lahjouji A, McNeil N. (2002, November). *Report of the unlicensed devices and experimental licenses working group.* Federal Communications Commission Spectrum Policy Task Force. Retrieved February 1, 2021 from https://transition.fcc.gov/ sptf/files /E&UWGFinalReport.pdf

[2]. TRAI, (2018, November) '*Telecommunication in select countries polices-statistics'*, Retrieved February 1, 2021 from www.trai.gov.in/release-publication/reports/ study-paper

[3]. Mitola, J. (1999). Cognitive radio for flexible mobile multimedia communications. *IEEE International Workshop on Mobile Multimedia Communications*, 3-10.

[4]. Haykin, S. (2005). Cognitive radio: brain-empowered wireless communications. *IEEE journal on selected areas in communications*, *23*(2), 201-220.

[5]. Arjoune, Y., & Kaabouch, N. (2019). A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions. *Sensors*, 19(1), 126. https://doi.org/ 10.3390/s 1901 0126

[6]. Lu, L., Zhou, X., Onunkwo, U., & Li, G. Y. (2012). Ten years of research in spectrum sensing and sharing in cognitive radio. *EURASIP journal on wireless communications and networking*, 1-16. https://doi.org/ 10.1186/ 1687-1499-2012-28.

[7]. Barlacchi, G., De Nadai, M., Larcher, R., Casella, A., Chitic, C., Torrisi, G., & Lepri, B. (2015). A multi-source dataset of urban life in the city of Milan and the Province of Trentino. *Scientific data*, 2(1), 1-15.

[8]. Chatterjee, S., Banerjee, A., Acharya, T., & Maity, S. P. (2014, August). Fuzzy c-means clustering in energy detection for cooperative spectrum sensing in cognitive radio system. *International workshop on multiple access communications, Springer*, 84-95.

[9]. López-Benítez, M., & Casadevall, F. (2012). Improved energy detection spectrum sensing for cognitive radio. *IET communications*, 6(8), 785-796.

[10]. Bhowmick, A., Chandra, A., Roy, S. D., & Kundu, S. (2015). Double threshold-based cooperative spectrum sensing for a cognitive radio network with improved energy detectors. *IET Communications*, 9(18), 2216-2226.

[11]. Charan, C., & Pandey, R. (2016). Eigenvalue based double threshold spectrum sensing under noise uncertainty for cognitive radio. *Optik*, 127(15), 5968-5975.

[12]. Chen, C., Chen, Y., Qian, J., & Xu, J. (2018, November). Triple-threshold cooperative spectrum sensing algorithm based on energy detection. IEEE 5th International Conference on Systems and Informatics, 791-795.

[13]. Diao, X., Dong, Q., Yang, Z., & Li, Y. (2017). Double-threshold cooperative spectrum sensing algorithm based on Sevcik fractal dimension. *Algorithms*, 10(3), 96.

[14]. Jan, S. U., Vu, V. H., & Koo, I. (2018). Throughput maximization using an SVM for multi-class hypothesis-based spectrum sensing in cognitive radio. *Applied Sciences*, 8(3), 421.

[15]. Yang K, Huang Z, Wang X, Li X. (2019, January). A blind spectrum sensing method based on deep learning. *Sensors*,19(10), 2270.

[16]. Ostovar, A., & Chang, Z. (2017). Optimisation of cooperative spectrum sensing via optimal power allocation in cognitive radio networks. *IET Communications*, 11(13), 2116-2124.

[17]. Shah, H. A., & Koo, I. (2018). Reliable machine learning based spectrum sensing in cognitive radio networks. *Wireless Communications and Mobile Computing*.

[18]. Venkateshkumar, U., & Ramakrishnan, S. (2020). Detection of spectrum hole from n-number of primary users using machine learning algorithms. *The Journal of Engineering*, 2020(5), 175-188.

[19]. Telecommunications - SMS, Call, Internet – MI. (2015, May). Retrieved February 1, 2021 from https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi :10.7910/DVN/EGZHFV

[20]. Boutaba, R., Salahuddin, M. A., Limam, N., Ayoubi, S., Shahriar, N., Estrada-Solano, F., & Caicedo, O. M. (2018). A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. *Journal of Internet Services and Applications*, 9(1), 1-99.

Declarations

Funding: No funding was received to assist with the preparation of this manuscript.

Conflicts of Interest / Competing Interests: The authors have no conflicts of interest to declare that are relevant to the content of this article.

All authors certify that they have no involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

The authors have no financial or proprietary interests in any material discussed in this article.