Application of Machine Learning in Fault Detection Using Control Chart Pattern Recognition

Talayeh Razzaghi, Ph.D.

Assistant Professor
Department of Industrial Engineering
New Mexico State University

Outline

Introduction

Control Charts

Control Chart Patterns

Imbalanced Classification

Proposed Methodology: Cost-sensitive learning based algorithm

Computational Results and Discussion

Conclusion

Introduction

Types of Analytical Models:

1. Descriptive Models
   - Shaping the questions and data into a structured problem

Introduction

Types of Analytical Models:

1. Descriptive Models
   - Shaping the questions and data into a structured problem
2. Predictive Models
   - Understanding the data and predicting the future

Introduction

Types of Analytical Models:

1. Descriptive Models
   - Shaping the questions and data into a structured problem
2. Predictive Models
   - Understanding the data and predicting the future
3. Prescriptive Models
   - Seeking optimal decisions to alter the future

Introduction

Types of Analytical Models:

1. Descriptive Models
   - Shaping the questions and data into a structured problem
2. **Predictive Models**
   - Understanding the data and predicting the future
3. Prescriptive Models
   - Seeking optimal decisions to alter the future

Introduction

- Predictive models are of interest to statisticians, computer scientists, and us (industrial engineers)!
- They are referred to with terms such as statistical learning, machine learning, and data mining.
- They have been applied to several applications.
  - Image Recognition
  - Manufacturing
  - Health Informatics
  - Cybersecurity

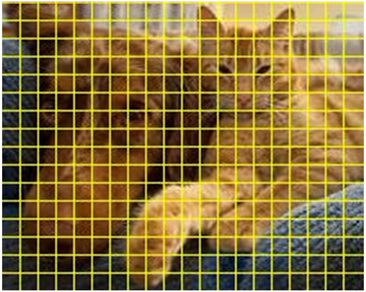
Introduction

- Predictive models are of interest to statisticians, computer scientists, and us (industrial engineers)!
- They are referred to with terms such as statistical learning, machine learning, and data mining.
- They have been applied to several applications.
  - **Image Recognition**

    Why has Image Recognition been at the center of attention for predictive analytics?
  - Manufacturing
  - Health Informatics
  - Cybersecurity

Introduction

Introduction



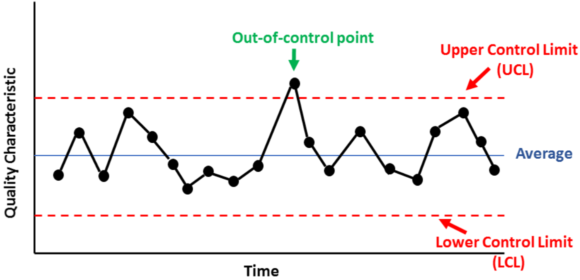
Introduction

The Bad and Good news:

- Not all applications offer a set of clean, perfect, and problem-free data to work.
- It is challenging to recognize and "treat" the issues that appear in real-world datasets.
- Examples of issues: imbalanced-ness, outliers, missing values, and massive size datasets

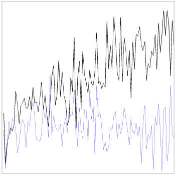
Control Charts

- Control charts are used for monitoring the behavior of a process.
- Control charts, also known as Shewhart charts (Walter A. Shewhart, 1920) or process-behavior charts.
- Control charts are a statistical process control tool used to determine if a manufacturing, chemical or business process is in a state of control.

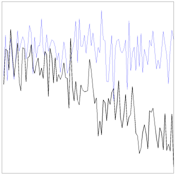


Introduction

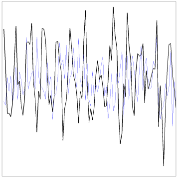
- Control charts are useful to identify not only out-of-control points but also the type of patterns



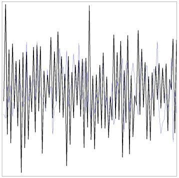
Up trend



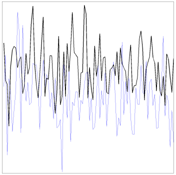
Down trend



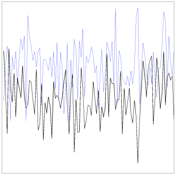
Cyclic



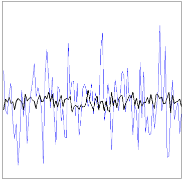
Systematic



Up shift



Down shift

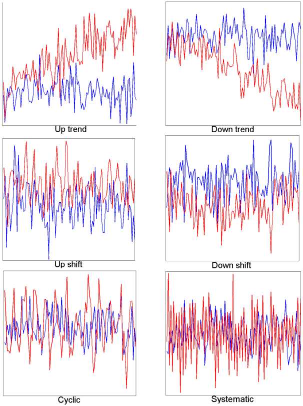


Stratification

Imbalanced Classification

- An application in Quality Control (Control Chart Pattern Recognition) *
  - Trend patterns
    - Stamping tonnage
    - Abnormal signals
  - Shift patterns
    - Variations of machine, material/operator
  - Cyclic Patterns
    - Voltage variability
    - Automotive body assembly
  - Systematic Patterns
    - Automotive body assembly

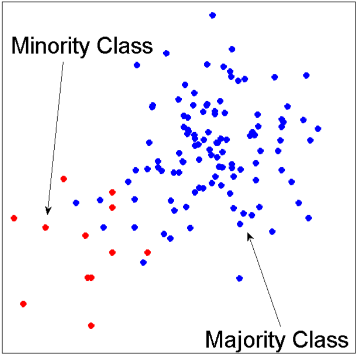
Western Electric Company (1958)



Control Chart Pattern Recognition(CCPR)

- Hachicha, W., & Ghorbel, A. (2012). A survey of control-chart pattern-recognition literature (1991-2010) based on a new conceptual classification scheme. *Computers & Industrial Engineering*, 36(1), 204-222
- However, one important parameter has been ***neglected***!!
  - Abnormal patterns are ***rare*** but important to detect
  - Normal patterns are ***common***
- CCPR belongs to the category of **imbalanced classification**

Imbalanced Data



Applications:

- Breast cancer detection (Verma et al., 2010)
- Credit card fraud detection (Wei et al., 2012)
- Oil spills detection in satellite radar images (Kubat et al., 1998)
- Network intrusion detection (Xu et al., 2011)
- **Control chart pattern recognition** (Xanthopolous & Razzaghi, 2014)

* *T. Razzaghi, P. Xanthopoulos, and A. Otero. Imbalanced Classification: Methods and Applications in Business Analytics. In Encyclopedia of Business Analytics and Optimization. IGI Global, 2014.*

Binary Classification Problem Definition

Preliminaries:

- Data represented by $(x_i, y_i) \in \mathbb{R}^m \times \{-1, 1\}$
  - x_i: actual *data*
  - y_i: corresponding *label* (binary case)

Classification Problem:

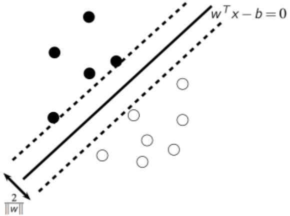
- Find a *classifier* function $f : \mathbb{R}^m \mapsto \{-1, 1\}$
- It can be used to predict the labels y_i^{test} of a group of data samples x_i^{test}
- Classification performance is evaluated through performance measures such as *Accuracy*, *Sensitivity*, *Specificity* and *G-mean*

Support Vector Machines (Vapnik, 2000):

- Classifier is obtained from solution of a *Quadratic Optimization* problem (Computationally tractable)
- Less over fitting in practice (unlike Artificial Neural Networks)
- Nice optimization problem structure

Proposed Methodology

- **Hard Margin Support Vector Machines**



- **Maximize** (*objective*) the separation margin ($2/\|w\|$) subject to **correct classification** (*constraints*)

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2 \tag{1a}$$

$$\text{s.t.} \quad y_i(w^T x_i - b) \geq 1, \qquad i = 1, \ldots, n \tag{1b}$$

Dual Formulation

- An arbitrary data sample x_u is assigned to a class y_u based on the following rule:

$$y_u = sgn(w^T x_u - b) \tag{2}$$

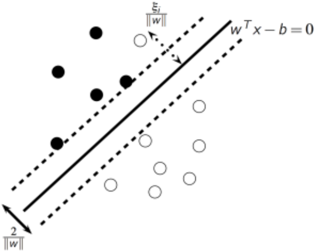
where $sgn(\cdot)$ is the sign function

- The separation hyperplane can be computed as follows:

$$w^* = \sum_{i=1}^{n} y_i \alpha_i^* x_i, \quad b^* = -\frac{\max_{y_i=-1}\langle w^* x_i\rangle + \min_{y_i=1}\langle w^* x_i\rangle}{2} \tag{3}$$

where a_i are the dual variables (or Lagrange multipliers associated with the the i^{th} constraint of the primal)

Inseparable Case: Soft Margin SVM



$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \tag{4a}$$

$$\text{s.t.} \quad y_i(w^T x_i - b) \geq 1 - \xi_i, \qquad i = 1, \ldots, n \tag{4b}$$

- Parameter C controls misclassification penalty

Inseparable Case: Soft Margin SVM

- The dual is calculated by the *Karush-Kuhn-Tucker (KKT)* conditions.

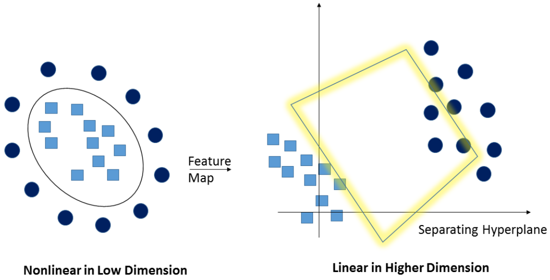
$$\max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \tag{5a}$$

$$\text{s.t.} \quad \sum_{j=1}^{n} \alpha_i y_i = 0 \tag{5b}$$

$$0 \leq \alpha_i \leq C \qquad i = 1, \ldots, n \tag{5c}$$

Extension to Nonlinear Classification (Kernels)

- Often the data sets are not linearly separable and the soft margin SVM, while feasible, yields poor performance (Cristianini and Shawe-Taylor, 2000)



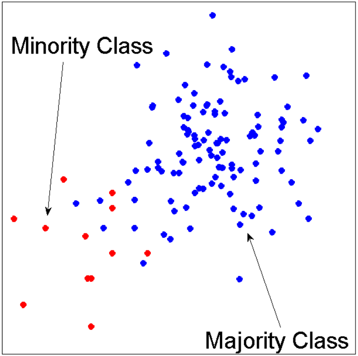
Extension to Non Linear Classification (Kernels)

- Embed data from **input space** to a higher dimension **feature space**
- This is done through an embedding function $\phi(x)$
- We denote $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$
- Popular kernel functions include:

Name	**Function**
Polynomial*	$\left(a x_i^T x_j + c\right)^d$
RBF	$\exp\left(-\gamma \|x_i - x_j\|^2\right)$
Cauchy	$\left(1 + \frac{1}{\alpha}\|x_i - x_j\|^2\right)^{-1}$
Inverse multi quadratic	$\left(\|x_i - x_j\|^2 + \alpha^2\right)^{-1/2}$

* For $a = 1, c = 0$ and $d = 1$ it is a *linear* kernel

Imbalanced Classification



Methods:

- Resampling (Chawla et al., 2002)
- Ensemble Learning (Boosting, bagging, etc.) (Freund and Schapire., 1997)
- **Cost-sensitive Learning** (Veropoulos et al., 1999)

Cost-Sensitive SVM

- Penalize misclassification of each class with different coefficient (Veropoulos, 1999)

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C^+ \sum_{\{i|y_i=+1\}}^{n^+} \xi_i + C^- \sum_{\{i|y_i=-1\}}^{n^-} \xi_i \tag{6a}$$

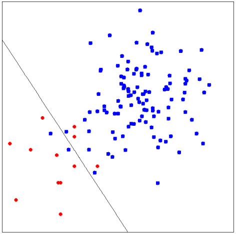
$$\text{s.t.} \quad y_i(w^T\phi(x_i) - b) \geq 1 - \xi_i, \qquad i = 1, \ldots, n \tag{6b}$$

$$\xi_i \geq 0, \qquad i = 1, \ldots, n \tag{6c}$$

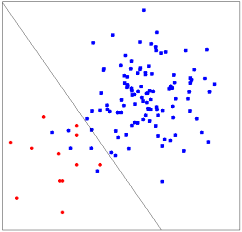
- The weights are usually chosen to be inversely proportional to the size of each class (n^+ and n^-):

$$C^+ = \frac{C}{n^+}, \quad C^- = \frac{C}{n^-} \tag{7}$$

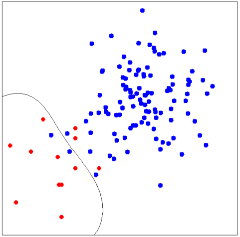
Proposed Methodology



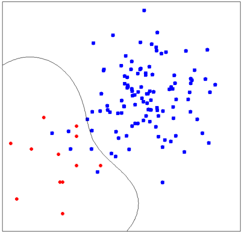
(a) Linear SVM



(b) Linear WSVM



(c) SVM with RBF kernel



(d) WSVM with RBF kernel

Performance Measures

- **Accuracy**: the percent of the correctly classified examples over the total number of examples
- **Sensitivity**
- **Specificity**

$$Sensitivity = \frac{TP}{TP + FN}, \quad Specificity = \frac{TN}{TN + FP} \tag{8}$$

- **G-mean**

$$G - Mean = \sqrt{Sensitivity * Specificity} \tag{9}$$

Table: Confusion Matrix

	Positive class	Negative class
Positive class	TP	FP
Negative class	FN	TN

Other Performance Measures for CCPR

- **Average Target Pattern Run Length (ATPRL)** (Hwarng & Hubele, 1991): the the average number of samples needed for discovering an abnormal pattern.
- **Average Run Length Index (ARLIDX)** (Hwarng & Hubele, 1991): which equals to the fraction of ATPRL divided by the discovery rate of abnormal patterns.

- The ARL-based measures are important especially for applications where **the production of each sample is cost and labor intensive.**
- Ultimately one wants to detect an anomaly with the **lower** ATPRL possible.

Experimental Setup

- SVM and WSVM models were solved using LIBSVM-3.12 and LIBSVM-weights-3.12.
- Data processing and further scripting were done in MATLAB.
- Experiments were conducted for highly imbalanced problems where 97.5% of the data belong to the normal class and only 2.5% belong to the abnormal.
- For each classification problem, we generate a total of 1000 data points and for cross validation purposes, 90% of the data was used for training and the rest 10% was used for testing.
- All data are normalized prior to classification, so that they have zero mean and unitary standard deviation.
- Radial basis function (RBF) kernel was used.

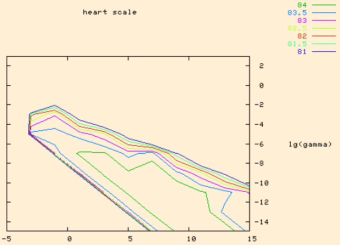
LIBSVM – A Library for Support Vector Machines

https://www.csie.ntu.edu.tw/~cjlin/libsvm/

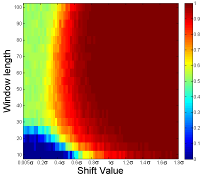
LIBSVM -- A Library for Support Vector Machines

Chih-Chung Chang and Chih-Jen Lin

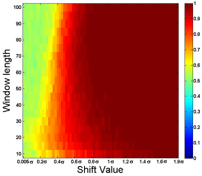
NEW Version 3.23 released on July 15, 2018. It conducts some minor fixes.
NEW LIBSVM tools provides **many extensions** of LIBSVM. Please check it if you need some functions not supported in LIBSVM.
NEW We now have a nice page LIBSVM data sets providing problems in LIBSVM format.
NEW A practical guide to SVM classification is available now! (mainly written for beginners)
We now have an easy script (easy.py) for users who know NOTHING about SVM. It makes everything automatic--from data scaling to parameter selection.
The parameter selection tool grid.py generates the following contour of cross-validation accuracy. To use this tool, you also need to install python and gnuplot.



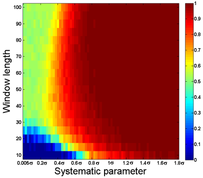
Computational Results



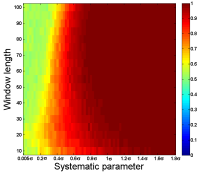
(c) Up shift/ Down shift (SVM)



(d) Up shift/ Down shift (WSVM)



(e) Systematic (SVM)



(f) Systematic (WSVM)

- SVM results in **poor classification performance** for inseparable and partially separable cases
- Our proposed WSVM is **effective** for CCPR in **a highly imbalanced environment**!

SVMs: more than 2 classes?

- The SVM as defined works for $K = 2$ classes. What do we do if we have $K > 2$ classes?
  - **One versus All (OVA)**: Fit K different 2-class SVM classifiers $\hat{f}_k(x)$, $k = 1, \ldots, K$; each class versus the rest. Classify x^* to the class for which $\hat{f}_k(x^*)$ is largest.
  - **One versus One (OVO)**: Fit all $\binom{k}{2}$ pairwise classifiers $\hat{f}_{kl}(x)$. Classify x^* to the class that wins the most pairwise competitions.
- Which to choose? If K is not too large, use OVO.

Multi-class classification

- The weighting Strategy for **multi-class WSVM** for CCPR

$$C_i = \frac{C}{n_i} \qquad i = 1, 2, \ldots, m \tag{10}$$

Table: Classification results for multi-class SVM and WSVM for CCPR with window length=10 and highly imbalanced data. Rows are related to predicted class labels and the columns are related to real labels.

		N	Dt	Ut	S	Ds	Us	C	Str
SVM	N	**1.00**	0.00	0.00	0.05	1.00	1.00	1.00	1.00
	Dt	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
	Ut	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
	S	0.00	0.00	0.00	0.95	0.00	0.00	0.00	0.00
	Ds	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Us	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WSVM	N	0.65	0.00	0.00	0.00	0.15	0.19	0.13	0.31
	Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
	Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
	S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
	Ds	0.06	0.00	0.00	0.00	**0.75**	0.00	0.00	0.08
	Us	0.10	0.00	0.00	0.00	0.00	**0.77**	0.00	0.00
	C	0.07	0.00	0.00	0.00	0.00	0.04	**0.70**	0.00
	Str	0.11	0.00	0.00	0.00	0.10	0.00	0.17	**0.61**

Multi-class classification

Table: Classification results for multi-class SVM and WSVM for CCPR with window length=50 and highly imbalanced data. Rows are related to predicted class labels and the columns are related to real labels.

		N	Dt	Ut	S	Ds	Us	C	Str
SVM	N	**1.00**	0.00	0.00	0.00	**0.40**	**1.00**	**0.47**	**1.00**
	Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
	Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
	S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
	Ds	0.00	0.00	0.00	0.00	0.60	0.00	0.00	0.00
	Us	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00	0.53	0.00
	Str	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
WSVM	N	0.98	0.00	0.00	0.00	0.20	0.37	0.27	0.37
	Dt	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00	0.00
	Ut	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00	0.00
	S	0.00	0.00	0.00	**1.00**	0.00	0.00	0.00	0.00
	Ds	0.00	0.50	0.00	0.00	**0.80**	0.00	0.00	0.00
	Us	0.00	0.00	0.00	0.00	0.00	**0.63**	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00	**0.73**	0.00
	Str	0.01	0.00	0.00	0.00	0.00	0.00	0.00	**0.63**

Wafer dataset (Adopted from UCR Time Series Classification Archive)

- Electronics manufacturing usually involves a large number of steps (> 250) which can induce defects to the final product.
- Quality control is performed by recording the different frequencies that are emitted by the plasma during the process.
- The data set composed of 1000 training samples (of length152 each) and 6174 testing samples of the same length (Olszewski, 2001; Keogh et al., 2011). The training samples are imbalanced (903 are majority and 97 minority).

Table: Performance for the wafer manufacturing industry dataset

		Sensitivity	Specificity	Gmean	Accuracy
Training	SVM	**0.9996**	0.9160	0.9156	**0.9913**
	WSVM	0.9967	**0.9350**	**0.9319**	0.9905
Testing	SVM	**0.9971**	0.9654	0.9811	**0.9937**
	WSVM	0.9895	**0.9895**	**0.9895**	0.9895

Results (cont'd)

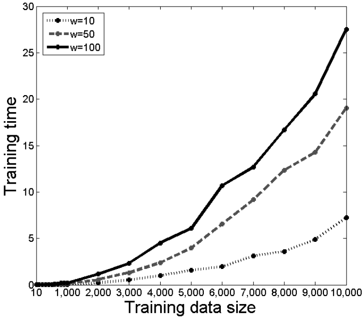


Figure: WSVM training and testing time vs. training size for cyclic pattern

Results (cont'd)

- For all patterns and most problem instances, **WSVM** has **lower ARLIDX**
- **Lower ARLIDX** are obtained compared to the ARLIDXin

Parameter	Uptrend		Upshift		Systematic		Cyclic		Stratification	
	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM	SVM	WSVM
0	155.54	155.19	155.65	155.06	155.21	155.19	155.92	155.83	155.28	155.22
0.005	16.33	**14.60**	138.41	**111.11**	110.83	**90.91**	78.51	**66.85**	7.90	**7.36**
0.03	13.04	**9.83**	129.87	**96.67**	106.90	**70.87**	**56.12**	56.64	8.00	**7.49**
0.055	11.34	**6.64**	**81.49**	81.75	94.95	**67.39**	75.76	**67.50**	8.03	**7.46**
0.08	9.46	**6.61**	103.12	**66.67**	80.73	**53.62**	**51.37**	63.33	8.05	**7.47**
0.105	8.37	**7.28**	106.51	**59.09**	80.36	**44.00**	67.11	**62.50**	7.96	**7.47**
0.13	8.09	**7.84**	90.50	**51.73**	62.92	**53.36**	67.40	**54.04**	7.99	**7.53**
0.155	7.50	**6.96**	70.50	**53.83**	74.21	**26.98**	69.64	**66.67**	8.05	**7.43**
0.18	7.10	**6.96**	73.83	**33.55**	59.54	**25.72**	67.94	**42.83**	8.10	**7.37**
0.205	6.81	**6.24**	62.91	**24.07**	67.33	**22.34**	53.69	**44.60**	7.91	**7.50**
0.23	7.47	**7.12**	76.66	**20.64**	77.87	**10.29**	69.44	**38.27**	8.20	**7.44**
0.255	7.01	**6.89**	37.44	**19.59**	62.76	**10.34**	59.36	**26.54**	8.11	**7.57**
0.28	7.66	**6.97**	40.74	**6.99**	79.42	**9.38**	69.61	**21.10**	8.08	**7.54**
0.305	7.01	**6.75**	49.65	**7.62**	52.66	**4.89**	75.76	**12.86**	8.16	**7.46**
0.33	7.27	**6.94**	60.99	**6.82**	36.13	**5.16**	59.33	**11.31**	8.24	**7.68**
0.355	6.49	**6.27**	37.28	**8.05**	49.41	**6.12**	54.31	**7.20**	8.47	**7.56**
0.38	7.50	**7.37**	25.59	**6.52**	48.56	**5.39**	56.96	**5.67**	8.50	**7.77**
0.405	**6.67**	6.69	24.99	**6.57**	27.87	**4.64**	66.00	**5.69**	9.06	**7.85**
0.43	6.75	**6.69**	18.74	**6.04**	22.04	**4.96**	46.91	**6.11**	9.56	**8.60**
0.455	6.76	6.76	19.08	**6.13**	23.24	**5.39**	45.03	**7.40**	10.76	**9.52**
0.48	6.50	**6.48**	15.13	**5.59**	25.15	**5.11**	42.57	**6.45**	37.64	**24.38**
0.505	**6.50**	6.52	14.86	**5.86**	16.26	**5.20**	38.86	**7.29**	**16.39**	16.44
0.53	6.85	**6.84**	13.81	**6.34**	13.82	**5.94**	45.88	**7.08**	40.81	**34.74**
0.555	6.69	6.69	13.33	**6.40**	15.65	**5.84**	44.54	**6.39**	47.17	**32.24**
0.58	6.51	6.51	12.59	**5.44**	22.55	**4.75**	40.28	**5.57**	72.46	**32.87**
0.605	**6.61**	6.65	16.58	**6.20**	20.11	**6.02**	37.30	**6.43**		
0.63	6.44	**6.43**	11.40	**5.85**	13.22	**5.61**	39.77	**6.83**		
0.655	**6.45**	6.50	13.81	**5.75**	12.36	**5.16**	31.47	**6.24**		
0.68	**6.21**	6.23	10.61	**5.71**	10.90	**5.48**	21.48	**6.56**		
0.705	**6.29**	6.30	12.46	**6.32**	16.04	**5.14**	18.27	**5.51**		
0.73	6.38	**6.34**	8.33	**5.89**	10.78	**5.97**	25.24	**6.21**		
0.755	6.26	**6.23**	10.09	**5.91**	10.19	**5.95**	17.92	**5.59**		
0.78	**6.18**	6.21	8.07	**5.93**	11.58	**5.71**	15.82	**7.25**		

Conclusion

- The proposed WSVM is **more effective** for imbalanced learning in CCPR problem.
- Current study results are **encouraging enough** in terms of average run length, computational time, and G-mean.
- **WSVM multi-class classification** helps to detect the abnormal points based on their types which **outperforms** SVM multi-class classification under **a highly imbalanced environment**.
- **Accuracy** might not be a proper performance indicator for imbalanced classification problems.
- SVMs do not directly provide **probability estimates**, these are calculated using an expensive five-fold cross-validation (Plat, 1999).
- For **nonlinear boundaries**, kernel SVMs are popular. Can use kernels with LR and LDA as well, but computations are more expensive.

Thank you!