Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

# Support Vector Machines for Class Imbalance Rail Data Classification with Bootstrapping-based Over-Sampling and Under-Sampling

**Ali Zughrat*, M. Mahfouf**, Y.Y.Yang***, S. Thornton******

**Dept of Automatic Control and Systems Eng, University of Sheffield S1 3JD UK*
*(Tel: +44 114 2225611; e-mail: a.zughrat@sheffield.ac.uk).*
***Dept of Automatic Control and Systems Eng, University of Sheffield S1 3JD UK*
*(Tel: +44 114 2225607; e-mail: m.mahfouf@sheffield.ac.uk).*
****Dept of Automatic Control and Systems Eng, University of Sheffield S1 3JD UK*
*(Tel: +44 114 2225611; yongyyang@gmail.com).*
*****Teesside Technology Centre, Tata Steel Europe, TS6 6US UK*
*(e-mail: Steve.Thornton@Tatasteel.com).*

**Abstract:** Support Vector Machines (SVMs) is a popular machine learning technique, which has proven to be very effective in solving many classical problems with balanced data sets in various application areas. However, this technique is also said to perform poorly when it is applied to the problem of learning from heavily imbalanced data sets where the majority classes significantly outnumber the minority classes. In this paper, we tackle the problem of learning from severely imbalanced Rail dataset via a new iterative support vector machine algorithm with bootstrapping-based over-sampling and under-sampling. We combine the good generalization ability of SVMs with the class distribution advantages of resampling techniques. Under-sampling and Over-sampling are commonly used methods for overcoming the class imbalance problem. In this work, we also address the influence of under-sampling and over-sampling techniques on rail data and show that achieving an optimal sampling rate yields a better SVM generalization capability. Experimental results show that the under-sampling outperforms over-sampling. The iterative SVM technique also shows a competitive generalization performance on the under-sampled rail data set, and that under-sampling can decrease the computational complexity of SVM algorithm.

*Keywords:* Support vector machines (SVMs), imbalanced data, under-sampling, oversampling.

## 1. INTRODUCTION

Since their introduction (Vapnik 1995), Support Vector Machine (SVM) has shown a remarkable success in solving many classical problems in various application areas. In classification tasks, SVMs are preferred by researchers to many other classification algorithms as they have a solid mathematical structure, a remarkable generalization performance and the ability to reach optimum classification solutions as the hyper-planes are determined by support vectors (Batuwita & Palade 2010). SVMs are machine learning classification algorithms which consider that the target classes exhibit a similarity in their prior probabilities and misclassification costs. However, in various real-world modelling scenarios, the data available are severely imbalanced. SVM classifiers perform poorly when learning from heavily imbalanced data. Imbalanced data become a real challenge in the knowledge discovery and data mining field. Imbalance data sets, also referred to as class imbalance learning, correspond to domains where there are many more examples of one class than the other class. Imbalance data Classification always causes problems as standard machine learning algorithms tend to be overwhelmed by the majority class and have a poor performance on the minority class. The problem of class imbalance has been addressed by machine learning researchers in two different ways: one is to change the class distribution of the date set at hand via applying various resampling approaches, such as under-sampling, over sampling, or the incorporation of both (Chawla et al. 2002) (Estabrooks et al. 2004). The other way is to internally modify the algorithm's structure by assigning different priorities to training examples and push the classifier to focus on minority class (Akbani et al. 2004).

In this paper, the objective is to elicit a better SVM model for classifying imbalanced rail data set by applying resampling techniques to achieve an optimum sampling rate that yields a better overall performance. The paper is organized as follows: Section 2 introduces an overview of the key production stages (steel making, continuous casting, rolling and finishing) from which the rail data is collected. Section 3 discusses the class imbalance learning solutions that are available for SVM classification. The support vector machines algorithm is derived in Section 4. In Section 5, we present the SVM classification performance and discuss in details a performance comparison and evaluation between the results obtained via bootstrapping-based oversampling and under-sampling. Concluding remarks are given in Section 6.

## 2. OVERVIEW OF RAIL MANUFACTORING ROUTE, KEY PRODUCTION STAGES

The rail manufacturing data utilized in this paper have been gathered from a complex steel manufacturing process which belongs to Tata Steel Europe. The data accumulated from the

rail manufacturing process is the accumulation of more than two years of production period (Yang et al. 2011). The rail production line consists of three key production stages, steel making, continuous casting, rails rolling and finishing as shown in Figure 1.

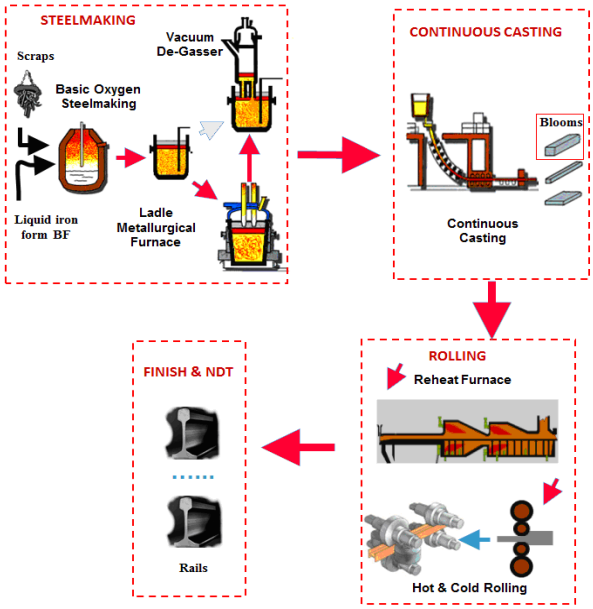


Fig. 1. Overview of the rail Key production stages (with permission from Tata-Steel Europe).

The iron ore as well as additional scraps are continuously charged into the top of huge blast furnace. The objective of the blast furnace is to produce hot metal by physically converting iron oxides into liquid iron. The liquid products (molten iron) are drained from the blast furnace to a basic oxygen steelmaking furnace in which carbon-rich molten iron is refined into steel. Basic oxygen steelmaking process blows oxygen through the molten iron. The key purpose of the process is to reduce the carbon content of the alloys and change it into low-carbon-steel. For further secondary steel making, the liquid steel is passed through a ladle metallurgical furnace in order to adjust steel chemical structures via desulphurization and alloy addition. The de-gasser unit will then improve steel cleanness by removing harmful hydrogen and other gases. In the stage of continuous casting, the rail molten steel is transferred to continuous casting machine by which 8-tonne steel blooms are produced (Yang et al. 2011). These produced blooms are heated and then fed directly to straightening operations and rolling mills at a proper temperature to yield rails up to 120 meters in length. The final stage of rail manufacturing process involves preventative measures against cracks and flaws and evaluating the properties of every rail via an inclusive non-Destructive testing (NDT) to ensure it meets applicable standards and quality control specifications, in addition to dimensional accuracy measurement, before dispatch to clients. A well designed data infrastructure, includes online data servers, is utilized to collect real-time variables, process parameters, quality inspection data and management information from rail production route through extensive instrumentations allocated for online monitoring and process control. The overall data is then saved in a master server where an advanced level of data mining and analysis can be performed. The original rail data collected from the rail production route is very large, with over 200 variables and over 65000 data records cover a production period of two years. Owning the fact that a careful data preparation is an essential part of exploratory data analysis, Data pre-processing framework was carefully designed to tackle the problem of detecting outliers, modifying incorrect data entries, and identifying relevant variables. Moreover, An intensive dimensionality reduction and input selection procedures were carried out on the rail data via correlation analysis and neural network modelling scheme (Yang et al. 2011). Such procedures have many potential advantages as they can reduce utilization and training time, improve overall performance of predictors by defying the curse of dimensionality and enhance generalization by reducing over-fitting. 39 inputs have been selected for this study as the most important input variables for the rail data where the rest of inputs are omitted. All the subsequent analysis will be based only on these inputs. The data has only one output consisting of integer values of (0, 1, 2 and 3) where 0 represents “good” rails, and values (1, 2 and 3) represents defected rails as per flow position ( end, middle, both) respectively. This study focuses on the rejected rails verified via an automatic and manual ultrasonic testing for the presence of internal irregularities such as cracks and flaws, to find root causes as well as identifying bottlenecks in the production route and thus applying appropriate control measures to improve process yields (reduce defects).

## 3. CLASS IMBALANCE LEARNING METHODS FOR SVM CLASSIFICATION

An iterative SVM classification strategy has been developed for the rail data with various options and key parameters built in the iterative strategy; only around 25% of the data were successfully classified. In any classification problem, the most important task is to correctly classify the minority class examples. Investigations were carried out on what are the root causes of such skewness of the model’s performance towards the majority class. Such a low success rate of classifying the rejected rails (minority class), was found to be due to the class imbalance phenomena in the training set. Class imbalance significantly hinders the performance of standard classifiers and modelling algorithms. The classification would always be biased in favour of the dominating class (majority), while the data related to the minority class tend to be misclassified. Such concern can be overcome via resampling techniques (Batuwita & Palade 2010); (Akbani et al. 2004); (Estabrooks et al. 2004) to lead to balanced data. To change class distribution of rail data, the following methods are applied for SVM rail data training:

### *3.1 Bootstrapping for Rail Data Resampling*

A data set is imbalanced if the samples corresponding to the majority class outnumber the samples belonging to the minority class. Since standard machine learning techniques and other modelling algorithms yield better classification

performance with balance data sets, Quality classification is not reachable with the current rail data set structure. Therefore, a direct data resampling approach is to be applied to change the class distribution of rail data.

Changing the class distribution can be conducted via different resampling strategies, such as over-sampling, under-sampling or combination of both. However, the oversampling technique has gained extra attention. The advantage of a such technique is that it is external and therefore, easily transportable as well as very simple to implement (Estabrooks et al. 2004). Moreover, over-sampling the minority class data avoids unnecessary information loss (Yang et al. 2011). For the over-sampling to be carried out, the original rail cast data are separated into two sub-sets. One set is for the dominating class and the other is for the minority class. Subsequently, the minority class data are fed into the bootstrapping resampling algorithm. The bootstrapping resampling algorithm yields a multiple randomly resampled subsets that have the same size as the size of the original minority subset (Yang et al. 2011). The resampled subsets are combined with the majority class data to shape the resampled training data that is ready for the subsequent training procedures. The design parameter $R_{mm}$ which is defined as the ratio of the number of samples belonging to the majority class to that belonging to the minority class plays a crucial role in the bootstrapping over-sampling algorithm as it controls imbalance level for the resampled training data set. All of the existed resampling techniques are tailored to resample until the desired ratio between the majority and minority classes is reached. Figure 2 shows the influence of the over-sampling strategy on the rail quality data.

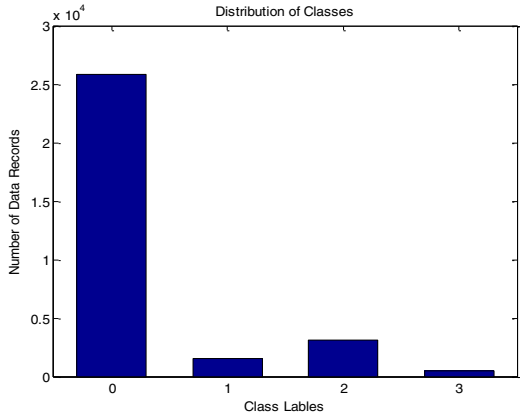


Fig.2. The influence of $R_{mm}$ on the resampled training data.

Oversampling has the property that there is no information from the original training set is lost since we keep all instances from majority and minority classes. However, when it is applied to a large scale data set, technical difficulties arise as the training data size is significantly increased. Therefore, the training time is also increased and a sufficient amount of memory is required to hold the training set. Since the dimensionality of rail data set is very high, the best sampling rate $R_{mm}$ achieved is 5. It is highly important to take into account the resampling time in order to keep time as well as memory complexity under reasonable constraints.

*3.2 Under-sampling*

Under-sampling is a popular resampling strategy that seeks to change the class distribution of the training data. It is considered as an independent pre-processing stage that can straightforwardly re-balance the date before training the classifiers. Therefore, it can be employed with any classification algorithm. In Random under-sampling, the majority class examples in the training data set are randomly eliminated until a desired ratio between the majority and minority class $R_{mm}$ is achieved. Consequently, the overall number of training examples is significantly reduced. Regardless of its simplicity, under-sampling has empirically shown to be very effective sampling approach. Since the rail data are highly dimensional data, there is a significant saving in classification time as well as memory. Theoretically, the main drawback of random under-sampling is that it discards data that may contain useful information for building an accurate model (Chawla et al. 2002). The significant $R_{mm}$ achieved via under-sampling rail quality data is a value of 1. The effect of under-sampling on the overall size of training data is illustrated in Figure 3.

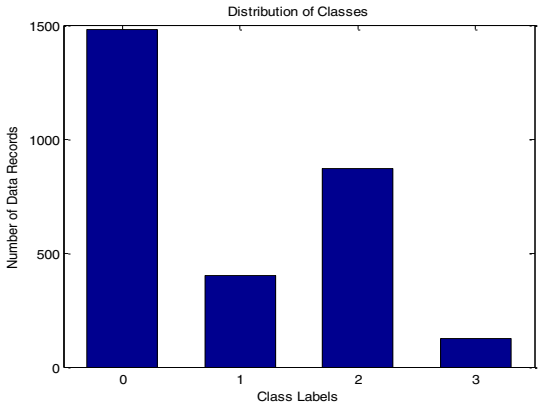


Fig.3. $R_{mm}$ effect to the size of training data.

## 4. SVM LEARNING THEORY

The theory of Support Vector Machines is a new classification technique particularly suitable for binary classification, although it can also be extended to other applications. The basic idea of SVMs is to find an optimum hyper-plane which separates the high-dimensional data into its two classes. Since the given data may often not be linearly separable, the notion of a "kernel feature space" is introduced which casts the data into a higher dimensional space where the data is separable. In this section we briefly review the idea of SVM in classification problems.

Let S be the data set of labeled training points $(y_1, x_1), \ldots, (y_l, x_l)$ Where $x_i \in R^n$ represents n-dimensional data points, $y_i$ represents the classes of which these data points are belonging to, $y_i = \{-1, 1\}$ and $i = 1, \ldots, n$. In order to find the hyper-plane that better separates the classes, the data records are mapped into a higher dimensional space via a mapping function $\varphi$, then the separating hyper-plane defined by the weight vector $(w)$ and bias $(b)$ can be represented as follows (Batuwita & Palade 2010):

$$w \cdot \varphi(x) + b = 0 \tag{4.1}$$

However, in many real-world applications, the data points are not totally linearly separable. Therefore, the optimization constraints can be generalized by introducing a slack variable $\varepsilon_i \geq 0$ where the soft-margin optimization problem is expressed as follows (Batuwita & Palade 2010):

$$Min \ \left(\frac{1}{2} w \cdot w + C \ \sum_{i=1}^{n} \varepsilon_i\right)$$

$$s.t. \ y_i \ (w \cdot \varphi(x_i) + b) \geq 1 - \ \varepsilon_i \,,$$

$$\varepsilon_i \geq 0, \ \ i = 1, \ldots \ , n \tag{4.2}$$

The variables $\varepsilon_i \geq 0$ hold for the misclassified points, the summation term $\sum_{i=1}^{n} \varepsilon_i$ is the measurement of the amount of misclassification, and the parameter C is the regularization parameter. The aforementioned optimization problem is a quadratic problem (QP) that can be solved by constructing nonnegative Lagrangian multipliers $\alpha_i$:

$$\text{Max } W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \varphi(x_i) \cdot \varphi(x_j)$$

$$\text{s.t. } \sum_{i=1}^{n} y_i \alpha_i = 0, \qquad 0 \leq \alpha_i \leq C, \qquad i = 1, \ldots \ , n \tag{4.3}$$

SVM employs a kernel function K that implements the dot product between the functions $\varphi(x_i)$, in such a case, the dual optimization problem can be transformed from an input space to a higher dimensional space. Accordingly, the nonlinear separating hyper-plane can be achieved as the solution of (Batuwita & Palade 2010):

$$\text{Max } W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$\text{s.t. } \sum_{i=1}^{n} y_i \alpha_i = 0, \ 0 \leq \alpha_i \leq C, \qquad i = 1, \ldots \ , n \tag{4.4}$$

After solving the quadratic problem and finding the optimal value of $\alpha_i$, the data points that have nonzero $\alpha_i$ values and fall in the margin are called support vectors.

## 5. SVM CLASSIFICATION AND PERFORMANCE COMPARISON

As stated earlier, the support vector machine algorithm is sensitive to the class imbalance learning (Batuwita & Palade 2010); (Lin & Wang 2002). In data classification, the choice of a Kernel function is challenging and becomes a central problem (Micchelli & Pontil 2005); (Prajapati & Patle 2010). Mapping the non-linear input space to a higher feature space (linear) via a kernel function depends significantly on the nature of the data. Therefore, The radial basis function (RBF) as a kernel is employed due to its clear implementation and the potential effectiveness on overall performance (Sahoo et al. 2013); (Prajapati & Patle 2010). Applying the aforementioned framework on the imbalanced rail data set has led to a poor generalization as well as a large number of support vectors. Support vector machine parameters, regularization parameter (C) and the width of (RBF), are optimized based on the grid search approach. The model's performance is skewed towards the majority class where the minority class is poorly classified at less than 25%. Sensitivity, specificity and accuracy performances are employed in our experiment as performance metrics throughout the confusion matrix. The confusion matrix for two class problem is illustrated in Table 1.

**Table 1. Two-class Confusion Matrix**

| | Predicted positive | Predicted Negative |
|---|---|---|
| Real Positive | TP(True Positive) | FN(False Negative) |
| Real Negative | FP(False Positive) | TN(True Negative) |

The performance measures of SVM classifier are assessed as follows:

$$\text{Specificity} = \frac{TN}{(TN+FP)}, \tag{4.5}$$

$$\text{Sensitivity} = \frac{TP}{(TP+FN)}, \tag{4.6}$$

And the total accuracy is expressed as:

$$\text{Accuracy} = \frac{(TN+TP)}{(TP+TN+FP+FN)} \tag{4.7}$$

Specificity is the ability of the algorithm to accurately classify the majority class whereas Sensitivity is the ability of the algorithm to accurately classify the minority class. Accuracy refers to the overall percentage that both classes are correctly classified. In this paper, bootstrapping based over-sampling and under-sampling schemes with different sampling rates are tailored to overcome the class imbalance phenomena. Consequently, it is not only the model's performance that has been significantly improved but the number of the support vectors has also been reduced. Table 2 illustrates a performance comparison of SVM algorithm with under-sampling and bootstrapping-based oversampling.

**Table 2. Performance comparison of SVM algorithm**

| | Number of Support vectors | Sensitivity % of testing set |
|---|---|---|
| Under-sampling | 2171 | 65.3 % |
| Oversampling | 23452 | 47.1% |

With under-sampling, the SVM algorithm has shown a good generalization with significant classification performance increment of 65.3%. Moreover, under-sampling technique succeeded in drastically reducing the number of support vectors of SVM classifier to 2171. The advantages of fewer support vectors will mostly mean short computational time and small memory requirements (Zheng et al. 2013). Theoretically, under-sampling has mostly the best trade-off between algorithm generalization capability and the number of support vectors. The maximum number of iterations is controlled via the number of parameters utilized in the grid search scheme. The results agree with the hypothesis that

under-sampling the majority class reduces the total number of training examples, speeding up the training time and accordingly ensure promising classification performance. Figure 4 illustrates the classification performance of the iterative SVM algorithm on the under-sampled rail data set.

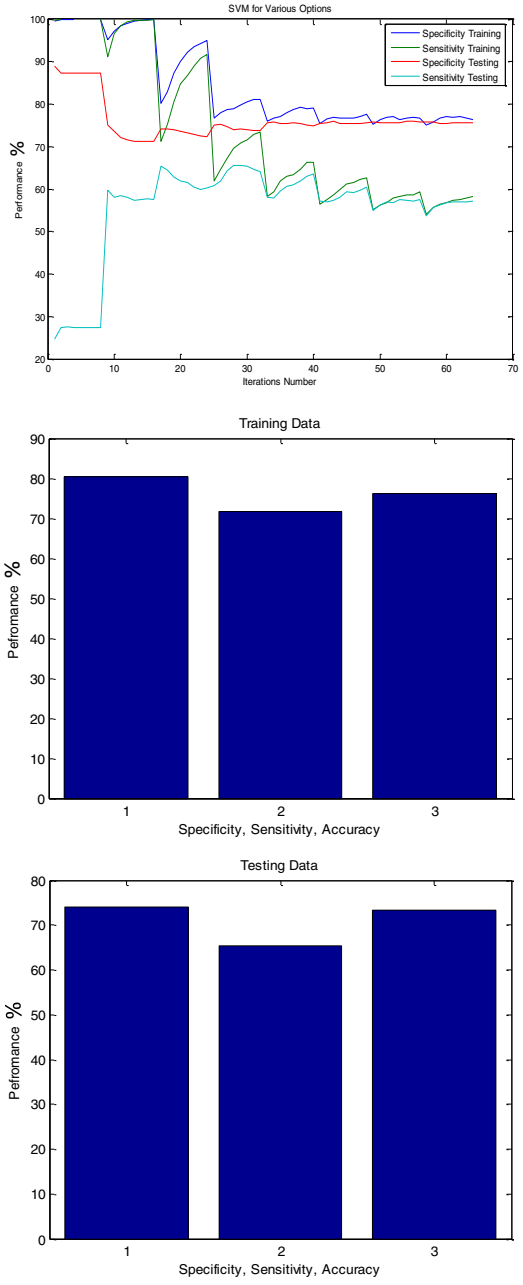


Fig.4. SVM classification for Under-sampled rail data.

The employment of the Bootstrapping-based over-sampling technique causes performance degradation to 47.1% and thus weak generalization capability, because time complexity grows dramatically as the size of the data increase. The SVM model built from an over-sampled data yields a large number of support vectors as shown in table 2. Our experimental results show that Bootstrapping-based over-sampling increases the computational cost associated with SVM training algorithm. It is worth mentioning that Large Data Modelling is Hungry for Resources and no convergence occurs when using 4GB memory due to the computationally expensive optimization phase. As a result, the computer memory has been extended to 16 GB. Figure 5 shows SVM performance with bootstrapping-based oversampling technique. Although it is a solid mathematical structure, it is worth mentioning that the SVM technique has drawbacks as it can be computationally expensive when dealing with large scale data and it tends to produce a large number of support vectors.

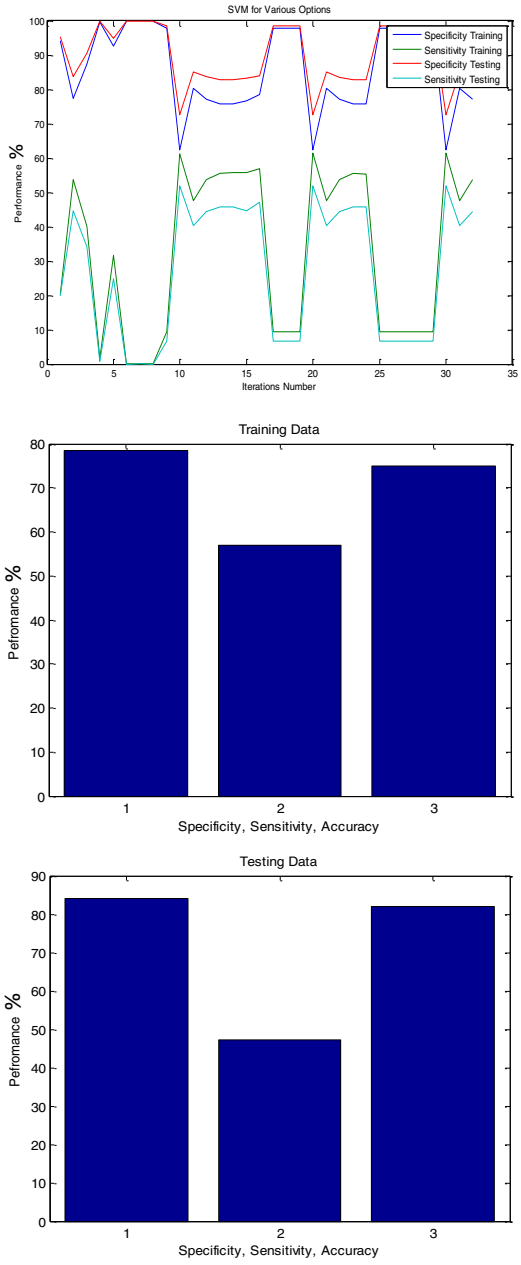


Fig.5. SVM performance with Bootstrapping-based Oversampling Scheme.

## 6. CONCLUSIONS

An approach to rail data classification via iterative SVMs with bootstrapping-based oversampling and under-sampling has been described. The results show that SVM is a promising algorithm for the resampled rail data classification

problem. Resampling techniques adopted in our experiment play crucial role for effective data classification, however, under-sampling can suppress the number of support vectors and result in a SVM with a significant performance gain. It also shows a remarkable reduction of the complexity of memory and training time. Class imbalance is not the only problem which tends to govern the performance of the learning algorithms, but there are other elements which potentially hinder the classification performance such as the overall size of the data set. Future research will investigate a cost sensitive learning and apply distinct costs to change class distribution of training data set. A second aspect worthy of further investigation is the inclusion of clustering prior to classification to reduce the number of support vectors.

## ACKNOWLEDGEMENT

The authors wish to thank Tata Steel Europe for the permission to utilize Rail Production Process data in this research.

## REFERENCES

Akbani, R., Kwek, S. & Japkowicz, N., 2004. Applying Support Vector Machines to Imbalanced Datasets. *In: proceedings of European Conference on Machine Learning: ECML*, pp.39–50.

Batuwita, R. & Palade, V., 2010. FSVM-CIL: Fuzzy Support Vector Machines for Class Imbalance Learning. *IEEE Transactions on Fuzzy Systems*, 18(3), pp.558–571.

Chawla, N. V et al., 2002. SMOTE : Synthetic Minority Over-sampling TEchnique. *Artificial Intelligence Research*, 16, pp.341–378.

Estabrooks, A., Jo, T. & Japkowicz, N., 2004. A Multiple Resampling Method for Learning from Imbalanced Data Sets. *Computational Intelligence*, 20(1), pp.18–36.

Lin, C.-F. & Wang, S.-D., 2002. Fuzzy support vector machines. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 13(2), pp.464–71.

Micchelli, C.A. & Pontil, M., 2005. Learning the Kernel Function via Regularization. *Journal of machine learning research*, 6, pp.1099–1125.

Prajapati, G.L. & Patle, A., 2010. On Performing Classification Using SVM with Radial Basis and Polynomial Kernel Functions. *3rd International Conference on Emerging Trends in Engineering and Technology,ICETET*, pp.512–515.

Sahoo, P. et al., 2013. On the study of GRBF and polynomial kernel based support vector machine in web logs. *Emerging Trends and Applications in Computer Science (ICETACS),1st International Conference on.IEEE*, pp.1–5.

Vapnik, V.N., 1995. *The Nature of Statistical Learning Theory*, Berlin, Germany: Springer.

Yang, Y.Y. et al., 2011. Adaptive neural-fuzzy inference system for classification of rail quality data with bootstrapping-based over-sampling. *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, pp.2205–2212.

Zheng, J. et al., 2013. An online incremental learning support vector machine for large-scale data. *Neural Computing and Applications*, 22(5), pp.1023–1035.