

# Anomaly Detection via Median based Filtering and Online osPCA

Amol M. Pawar, Manisha S. Mahindrakar

Department of Computer Science & Engineering, SGGS IE & T, SRTM University, Nanded, India.  
[pawaramolm@sggs.ac.in](mailto:pawaramolm@sggs.ac.in), [msmahindrakar@sggs.ac.in](mailto:msmahindrakar@sggs.ac.in).

---

**Abstract:** Anomaly detection is a crucial analysis topic in the field of data mining as well as machine learning. Several real world applications like intrusion or master card fraud detection require a decent and effective system to spot deviated data instances. A decent anomaly detection methodology must be able to accurately establish many varieties of anomalies, robust, need comparatively very little resources, and perform detection in period of time. In this paper we proposed the idea of combining the two different algorithms i.e. Median Based Filtering and Online Oversampling PCA for effective detection of anomaly in online updating mode. Median Based filtering uses the inter quartile range which is a measure of statistical dispersion being equal to the difference between the upper and lower quartiles. Whereas oversampling PCA does not need to store the entire covariance matrix or data matrix and thus this approach is a more useful in online or large scale problem. Compared with other anomaly detection algorithm our experimental result verifies the feasibility of our proposed method.

**Keywords:** Anomaly, principal component analysis, median, oversampling, leave one out.

---

## 1. Introduction

Data Mining is the process of automatically discovering useful information in large data repositories and using it for decision making. However, there are lots of problem exist in mining data in large dataset such as data redundancy, the values of attributes is not specific, data is not complete and anomaly (or outlier). Anomaly detection aims to spot a little group of instances that deviate remarkably from the existing data. A renowned definition of “outlier” in [1]: “An observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism,” which gives the general idea of an outlier. Basically, anomaly detection may be found in applications like Office or Homeland Security, MasterCard fraud detection, Intrusion and Business Executive threat detection in Cyber-security, Fault detection, or Malignant diagnosing. However, since exclusively restricted quantity of labeled data are unit accessible for the above mentioned real-world applications, the data mining researchers and the communities of machine learning look after situation of how to estimate anomaly of unseen data or events.

In this paper we proposed the idea of combining the two different algorithms i.e. Median Based filtering which uses the concept of Interquartile Range (IQR) and Online Oversampling PCA for effective detection of online anomaly. This paper also compares other popular outlier detection techniques to determine the feasibility of our proposed method based on Area Under Curve i.e. AUC scores. There are many types of data available for anomaly detection analysis such as binary variables, nominal and ordinal. We will analyze above techniques by using pima and cod-rna data set.

The rest of the paper is organized as follows. Section 2 discusses the work related on anomaly detection techniques. Section 3 provides details of the proposed Anomaly Detection technique. Performance evaluation is done in Section 4. And at last Section 5 contains conclusion and summary of anomaly detection techniques.

## 2. Related Work

In past few years many anomaly detection algorithms has been proposed. These approaches are broadly divided into three categories Statistical approach, Distance based approach and Density based approach. Statistical approaches [1] assume that the data follows some normal or considered distributions, and this kind of approach aims to seek out the outliers that deviate from such distributions. However, most distribution models are assumed univariate, and so the lack of robustness for multidimensional information could be a concern. Moreover, since technique is usually enforced within the original data space directly, their solution models may suffer from the noise within the data.

For Distance based ways [2], the distances between every data point (of interest) and its neighbors are calculated. If the result is more than some preset threshold, then the targeted instance is considered as outlier; whereas it requires knowledge of data distribution. These approaches may encounter issues once the

data distribution is complicated (e.g., multi-clustered structure). In such cases this sort of approach can lead to crucial improper neighbors, and therefore outliers cannot be properly known.

To alleviate the aforementioned drawback, Density-based approach is projected [3]. One among the representatives of this kind of approach is to use a Density-based outlier technique Local Outlier Factor (LOF) to live the outlierness of every data instance. Supported the local density of every data instance, the LOF determines the factor of outlierness, that provides suspicious ranking scores for all samples. The foremost necessary property of LOF is the ability to estimate local structure via Density Estimation. This enables users to spot outliers under a global data structure. However, its price noting that the estimation of local data density for every instance is extremely computationally expensive, particularly once the dimension of the data set is oversize.

Beside this some recently proposed approaches are given below:-

A novel approach named Angle-Based Outlier Detection (ABOD) [5] and some variants assessing the variance in the angles between the difference vectors of a point to the other points. This way, the effects of the “curse of dimensionality” are alleviated compared to purely distance-based approaches. A main advantage of this approach is that ABOD method does not rely on any parameter selection influencing the quality of the achieved ranking. As compare with well-established distance-based method LOF performance of very well especially on high dimensional data. Consequently, a fast ABOD method is planned to get associate degree approximation of the initial ABOD answer. The difference between the ABOD and the fast ABOD approaches is that the latter solely considers the variance of the angles between the target instance and its k nearest neighbors. However, the search of the closest neighbors still prohibits its extension to huge scale issues (batch or on-line modes), since the user can have to be forced to keep all information instances to calculate the desired angle data.

In Online Anomaly Detection using KDE, large backbone networks are regularly affected by a range of anomalies. Online anomaly detection algorithm based on Kernel Density Estimates sequentially and adaptively learns the definition of normality in the given application. It does not assume any prior knowledge regarding the underlying distributions, and then detects anomalies subject to a user-set tolerance level for false alarms. Comparing with the existing methods of Geometric Entropy Minimization, Principal Component Analysis and One-Class Neighbor Machine demonstrates that the KDE method achieves superior performance with lower complexity. Future work will revolve around investigating other means of bootstrapping KEAD to achieve a user-specified error tolerance limit.

### 3. Research Methodology

Figure 1 illustrates the framework of our online anomaly detection. This framework has two phases. Those are:

1. Data Cleaning phase
2. Online Anomaly Detection phase

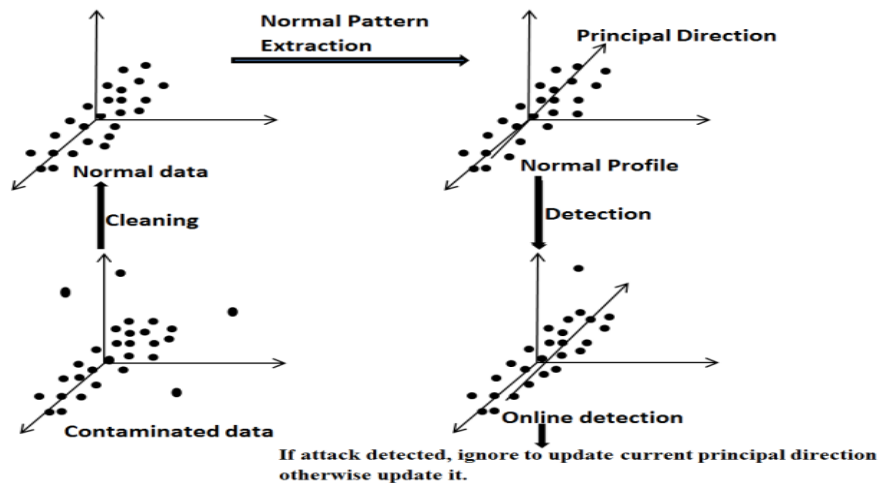


Figure 1. Framework of Online Anomaly Detection.

Here, in this paper we are proposing median based filtering technique to spot distrustful data in the data cleaning phase. In the online anomaly detection phase, our goal is to recognize the new arriving abnormal instance. The quick updating of the principal directions given in this approach can satisfy the online detecting demand. A new arriving instance will be marked if its suspicious score is higher than the mean plus a specified multiple of the standard deviation. This online anomaly detection approach is well suited for online applications as it overcomes the computation issues and utilizes the memory efficiently.

**A. Median based Filtering**

Median based filtering technique is used for removing outliers from a multivariate data set using the median of each column. It removes possible outliers from a data set by specifying a cut off. Cut off is a multiple of the inter quartile range above Q3 and below Q1. Initially it sorts the data from which outliers has to remove. Once the data is sorted interquartile range as well as semi interquartile range is computed. And those columns having data above interquartile ranges are removed. The median based filtering algorithm for data cleaning is described in Algorithm 1. We apply median based filtering algorithm on UCI repository pima and cod-rna database. We check its efficiency by performing the data loss check on dataset by classifying it by applying zeroR classifier in weka. The table 1 shows the result of classification before and after applying the median based filtering. On the basis of factors given in the table we can say that after filtering the dataset it does not affect the structure of the dataset.

---

**Algorithm 1: Median based Filtering using median of each column from multivariate data set.**

Input: The Data Matrix  $x$ , OUTLIER\_CUT a cutoff which is a multiple of interquartile range.

Output: Filtered Data.

Calculate the size "num\_column" of the data set " $x$ "

For  $i = 1$  to num\_column do

Calculate the 25th, 50th and 75th percentile, i.e. the median.

Calculate the interquartile and semi interquartile range of each column.

Find the values which are above Q3 and below Q1 to remove the outlier from the original dataset for both "outliersQ1" and "outliersQ3".

End For.

---

Table 1: Efficiency of median based filtering for pima dataset

Factors	Before	After
Dataset (size in rows)	768	535
Correlation Coefficient	0	0
Mean Absolute Error	0.9087	0.8004
Root Mean Squared Error	0.9533	0.8947
Relative Absolute Error	100%	100%
Root Relative Squared Error	100%	100%

**B. Principal Component Analysis**

PCA is a widely used unsupervised dimension reduction method in statistics and data mining because it is a simple and non-parametric method [7, 8]. It can retain those characteristics of the dataset that contribute most to its variance by keeping lower- order principal components. These few components often contain the "most important" aspects of the data. However, PCA does not take the class label into account. To sum up, PCA is an orthogonal basis transformation of the coordinate system in which data are projected into a subspace. The new coordinate values by which we describe the data are called principal components. The new coordinate system is called principal directions in which data would be most separated.

Let  $A \in \mathbb{R}^{n \times p}$ , be the data matrix and each column  $x_i \in \mathbb{R}^p$ , represents an instance. PCA involves the eigenvalue decomposition in the covariance matrix of the data. Its formulation is solving an eigenvalue problem as follows:

$$\Sigma_A U = U \Lambda, \dots \dots \dots (1)$$

Where

$$\Sigma_A = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \dots \dots (2)$$

is the covariance matrix,  $\mu$  is the global mean,  $\Lambda$  is a diagonal matrix whose diagonal entries are eigenvalues, and the resulting  $U$  is a matrix whose column vectors are the eigenvector set. In practice, some eigenvalues have little contribution to variance and can be discarded. It means that we only need to keep few components to represent the data. Besides, PCA explains variance and is sensitive to outliers. A few points distant from the center would have a large influence on variance and its principal directions. In other words, these first few principal directions will be influenced seriously if our data contain some outliers.

Because the covariance matrix is a symmetric positive semi-definite matrix, its eigenvalues are all greater or equal to zero. Essentially, PCA has the following properties if we assume that the eigenvectors are sorted in descending order of the eigenvalue magnitude [9]: (1) the first  $q$  ( $q < p$ ) eigenvectors explain more part of the variance than any other  $q$  orthogonal directions, (2) the square error in representing the data by the first  $q$  principal components is minimal, (3) the principal components are uncorrelated, (4) the first  $q$  principal components have maximal mutual information with respect to the inputs (this holds under Gaussian assumption, and thus depends on the data), and (5) PCA is the method for placing the data in a low dimensional space such that the Euclidean distance between any two points in the low dimensional space is as close as possible to their distance in the original high dimensional space.

### C. Oversampling principal component analysis

We identify outliers in our data and detect the new arriving outliers through the variation of the principal directions. However, the effect “with or without” of a particular data may be diminished when the size of the data is large. On the other hand, the computation in estimating the principal directions will be heavy because we need to recompute the principal directions many times in LOO scenario.

In order to overcome the first problem, we employ “oversampling” scheme to amplify the outlierness on each data point. For identifying an outlier via LOO strategy, we duplicate the target instance of removing it. That is, we duplicate the target instance many times (e.g. 10% of the whole data) and observe how much variation the principal directions vary. With this oversampling, the principal direction and mean of the data will only be affected slightly if the target instance is a normal data point (see Fig. 2(a)). On the contrary, the variation will be enlarged if we duplicate an outlier (see Fig. 2(b)). On the other hand, we also can apply oversampling scheme in the LOO procedure with incremental case. The main idea is to enlarge the difference of the effect between a normal data point and an outlier. For computation issue, we need to recompute the principal direction many times in the LOO scenario. In order to avoid this heavy loading, we also proposed two strategies to accelerate the procedure in estimating principal direction. The first one is the fast updating for the covariance matrix. Another one is the solving the eigenvalue problem via the power method [10, 11]. As Eq. (1) shows the formulation of PCA is solving eigenvalue decomposition on the covariance matrix of the data. However, it is unnecessary to completely re-compute the covariance matrix in the LOO procedure. The difference of covariance matrix can be easily adjusted while we only duplicate one instance. Hence, we consider a light updating of covariance matrix for fast computation [10].

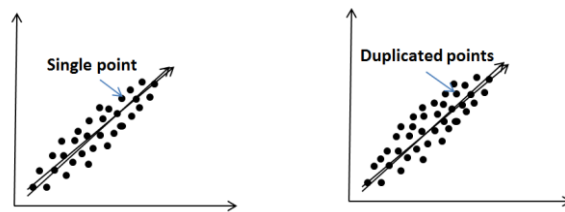


Figure 2(a): Over-sampling a normal data point

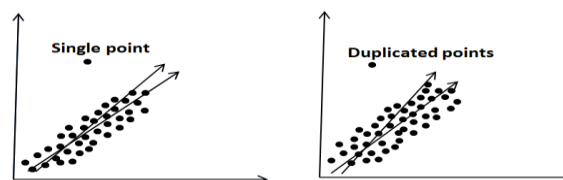


Figure 2(b): Over-sampling an outlier

Let  $Q = \frac{AA^T}{n}$  be the pre computed scaled outer product matrix. We use the following updating for the adjusted mean vector  $\tilde{\mu}$  and covariance matrix  $\tilde{\Sigma}$  :

$$\tilde{\mu} = \frac{\mu + r \cdot x_t}{1+r} \dots\dots(3)$$

And

$$\tilde{\Sigma} = \frac{1}{1+r}Q + \frac{r}{1+r}X_tX_t^T - \tilde{\mu}\tilde{\mu}^T, \dots\dots(4)$$

Where,  $A \in \mathbb{R}^{n \times p}$  is the data matrix,  $x_t$  is the target instance and  $r$  is the parameter of the proportion of the whole data in duplicate  $x_t$ . From Eq. (3), it shows that we can keep the matrix  $Q$  in advance and need not to recompute it completely in LOO procedure. If we do not use the oversampling technique, that is to say, we only remove one instance in LOO procedure. Then we use the following updating for the adjusted mean vector  $\tilde{\mu}$  and covariance matrix  $\tilde{\Sigma}$  :

$$\tilde{\mu} = \frac{\mu \cdot n - x_t}{n-1} \dots\dots(5)$$

And

$$\tilde{\Sigma} = \frac{n}{n-1} \left( Q - \frac{x_t x_t^T}{n} \right) - \tilde{\mu} \tilde{\mu}^T, \dots\dots(6)$$

In extracting the first principal direction, we also apply the power method for fast computation. Power method [10, 11] is an eigenvalue algorithm for computing the greatest eigenvalue and the corresponding eigenvector. Given a matrix  $M \in \mathbb{R}^{p \times p}$ , this method starts with an initial normalized vector  $\mu_0$ , which could be an approximation to the dominant eigenvector or a nonzero random vector, and then iterative computes the  $\mu_{k+1}$  as follows:

$$\mu_{k+1} = \frac{M\mu_k}{\|M\mu_k\|} \dots\dots(7)$$

The sequence  $\{\mu_k\}$  converges on the assumption that  $M$  has  $p$  linear independent eigen vectors and that the corresponding eigenvalue satisfy  $|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_p|$ . Some books like [10] describes that  $M$  must have  $p$  distinct eigenvalues. In fact, this assumption is too strict. If  $M$  has  $p$  distinct eigenvalues, then  $M$  has  $p$  linear independent eigenvectors. However, if  $M$  has  $p$  linear independent eigenvectors, it doesn't mean that  $M$  has distinct  $p$  eigenvalues. The starting vector  $\mu_0$  can be expressed as the linear combination [10]:

$$\mu_0 = b_1 v_1 + b_2 v_2 + \dots + b_p v_p \dots\dots(8)$$

And

$$\mu_1 = \frac{\lambda_1}{c_1} \left( b_1 v_1 + b_2 \left( \frac{\lambda_2}{\lambda_1} \right) v_2 + \dots + b_p \left( \frac{\lambda_p}{\lambda_1} \right) v_p \right) \dots\dots(9)$$

After  $k$  iteration we arrive at

$$\mu_k = \frac{\lambda_1^k}{c_1 c_2 \dots c_k} \left( b_1 v_1 + b_2 \left( \frac{\lambda_2}{\lambda_1} \right)^{k-1} v_2 + \dots + b_p \left( \frac{\lambda_p}{\lambda_1} \right)^{k-1} v_p \right) \dots\dots(10)$$

Based on the fact that [11]

$$\lim_{k \rightarrow \infty} \frac{b_1 \lambda_1^k}{c_1 c_2 \dots c_k} = 1 \dots\dots(11)$$

Therefore, the sequence of vector  $\{\mu_k\}$  converges to the dominant eigenvector:

$$\lim_{k \rightarrow \infty} \mu_k = v_1 \dots\dots(12)$$

In our scenario, the covariance matrix  $\Sigma_A \in \mathbb{R}^{p \times p}$  is symmetric, there is an orthogonal matrix  $V$  that diagonalizes  $\Sigma_A$ , that is  $V^T \Sigma_A V = \Lambda$  where  $\Lambda$  is diagonal [12]. Hence  $\Sigma_A$  has  $p$  linear independent eigenvector. In addition  $\Sigma_A$  is a positive semi-definite matrix, its eigenvalues are all greater than or equal to zero. Generally speaking, there always exists a direction on which the original data projected would have the largest variance. Therefore, we can use the power method in extracting the first principal direction. Moreover, from Eq. (6), we can see that power method does not compute matrix decomposition but uses the matrix multiplication. Based on this property, the power method can converge rapidly and make our LOO procedure faster. On the other hand if we want to find the remaining eigenvectors, we could use deflation process [10, 12]. Note that we only use the first principal direction in our experiment so we only apply the power method in estimating the first principal direction. Algorithm for online anomaly detection is described in Algorithm 2.

**Algorithm 2: Over-sampling Principal Component Analysis for On-line Anomaly Detection**

- Input: Data Matrix
- Output: Score of Outlierness.
- Compute the Principal Direction for data matrix.
- For  $i=1$  to  $n$
- Compute Principal Direction for targeted instance.
- Compute Score = absolute value of cosine similarity.
- End For.

#### 4. Performance Evaluation

In our experiments, we implement all methods in MATLAB [13]. Before each experiment, we do normalization in order to let each attribute contribute equally. We let each attribute value subtract the mean of that feature and then divide it by the standard deviation of that feature. Generally speaking, the result of classification can be represented in a confusion matrix as shown in table 2. The main diagonal entries record accurate classification result, on the other hand, the skew diagonal entries record erroneous classification result. If the skew diagonal entries of the confusion matrix are all zero, the result of classification is perfect. Three performance measures, precision, recall (detection rate, true positive rate) and false positive rate, are also of interest. They are defined individually by  $TP/(TP + FP)$ ,  $TP/(TP + FN)$ ,  $FP/(TP + FP)$ , and . Notice that false positive rate equals one minus precision. The Receiver Operating Characteristics curve (ROC) was first used in signal detection theory to represent the tradeoff between the detection rates and false positive rates.

Table 2: The Confusion Matrix for Evaluating the Result of Classification

	<b>Predicted Outlier</b>	<b>Predicted Normal</b>
<b>Actual Outlier</b>	True Positive (TP)	False Negative (FN)
<b>Actual Normal</b>	False Positive (FP)	True Negative (TN)

The ideal ROC has 100% detection rate and 0% false positive rate. The area under the ROC, or simply AUC, has been shown as a measurement for the quality of ranking. We use the AUC to evaluate the suspicious outlier ranking. The closer the AUC value is to 1, the closer the ROC is to the ideal ROC. The AUC value quantitatively evaluates the performance. The range of AUC value is between 0 and 1. The following is a terse formula to compute AUC value,

$$AUC = \frac{\sum_{i=1}^m \sum_{j=1}^n I(f(x_i) > f((x_j))}{mn}$$

Where I(true) equals 1 and I(false) equals 0. The m stands for number of positive instances and n represents number of negative instances.

We evaluate our outlier detection method on Pima and Cod-rna dataset which can be obtained from UCI repository of machine learning data archive [14]. In this experiment we compare our method (median based filtering + online osPCA) with oversampling PCA (osPCA), online oversampling PCA (online osPCA), Fast ABOD and LOF. We use the area under the ROC curve (AUC) [15] to evaluate the suspicious outlier ranking. We present the average AUC score in table 3. From this table, we observe that our proposed online anomaly detection technique consistently achieved better or comparable results, while our method is the most computationally efficient one among the method considered.

Table 3: AUC Scores of Fast ABOD, LOF, Oversampling PCA (osPCA) with Power Method, osPCA with Online Updating Algorithm and Our Technique of Combining MBF+osPCA.

Dataset \ Algorithm	Pima (768*8)	Cod-rna (59535*8)
Fast ABOD	0.38907	0.64121
LOF	0.6493	0.47176
osPCA	0.66684	0.48186
Online osPCA	0.67624	0.48185
MBF + osPCA	0.69717	0.57437

#### Conclusion

This paper explored the idea of combining the two different algorithms for online anomaly detection which is totally different from other available techniques. From the experimental results, we demonstrate that the combining the algorithms can improve the effectiveness of outlier detection algorithm. In addition, an effective computation for computing the covariance matrix and the speedup of the power method for estimating principal direction in LOO are also proposed for reducing the computational loading and satisfying the online detection of anomaly. Furthermore, our method does not need to keep the entire covariance matrix or data matrix during the on line detection phase. Therefore, compared with the other anomaly detection method, our approach is able to achieve the satisfactory results while significantly reduce computational cost and memory requirement. Thus our technique is useful for large scale data problem.

## References

- [1] D.M. Hawkins, Identification of Outliers. Chapman and Hall, 1980.
- [2] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-Based Detection and Prediction of Outliers," *IEEE Trans. Knowledge and Data Eng.*, vol. 18, no. 2, pp. 145-160, 2006.
- [3] W. Jin, A.K.H. Tung, J. Han, and W. Wang "Ranking Outliers Using Symmetric Neighborhood Relationship", *Proc. Pacific-Asia Conf. Knowledge Discovery and Data Mining*, 2006.
- [4] C. C. Aggarwal and S. P. Yu, "An effective and efficient algorithm for high-dimensional outlier detection", *The VLDB Journal*, vol. 14, pp. 211-221, 2005.
- [5] H. P. Kriegel, M. Schubert, and A. Zimek, "Angle-Based Outlier Detection in High-Dimensional Data," *Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and data Mining*, 2008.
- [6] Ahmed, T. Online Anomaly Detection using KDE.2009. Proc. IEEE Conf. Global Telecomm. 2009.
- [7] Yuh-Jye Lee, Yi-Ren Yeh, and Yu-Chiang Frank Wang," Anomaly Detection via Online Oversampling Principal Component Analysis" *IEEE Transactions on Knowledge and Data Engineering*, Vol. 25, No. 7, July 2013.
- [8] Ian Jolliffe, "Principal Component Analysis," *Springer-Verlag*, New York, 1986.
- [9] B. Scholkopf, A. J. Smola, and K.-R. Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Computation*, pages 1299-1319, 1998.
- [10] G. H. Golub and C. F. Van Loan, "Matrix Computations," Johns Hopkins University Press, Baltimore, MD, USA, 1983.
- [11] L. N. Trefethen and D. Bau, "Numerical Linear Algebra," Society for Industrial and Applied Mathematics, Philadelphia, 1997.
- [12] S. J. Leon, "Linear Algebra with Applications," Prentice Hall, 2006.
- [13] E. Alpaydin, "Introduction to Machine Learning," The MIT Press, 2004.
- [14] MATLAB, "User's Guide" The Math Works, Inc., Natick, MA 01760, 1994-2001.<http://www.mathworks.com>.
- [15] "UCI Repository of Machine Learning Databases", <http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html>
- [16] P. Bradley, "The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms," *Pattern Recognition*, pages 1145-1159, 1997.