A Survey and Implementation of Financial Alarm Classification

Jens Wirén and
Farhad Kimanos

Bachelor Thesis at the Department of

Computer Science and Communication,

The Royal Institute of Technology

INTRODUCTION

This thesis is done in co-operation with Scila AB and the goal is to find, implement and evaluate a suitable machine learning algorithm to classify and predict true and false alerts using labeled data. Alerts are triggered in the Scila Surveillance software when certain parameters are exceeded in a trade, such as a to big volume over a to small time-span.

METHODOLOGY

The project began with a study of related work to evaluate and find an suitable machine learning algorithm. Of several examined, *Support Vector Machines* was the method of choice as it is proven to be generally effective.

The SVM classifier is dependent of two parameters but no theoretical foundation for how to chose these exists. Therefore the method of *grid-search* is used and to prevent over-fitting we implement *cross-validation*. Our dataset is by nature very unbalanced and *weights* are used to counter this.

In order to determine which results are good we evaluate several accuracy measurements and conclude that *Balanced Accuracy* is the most useful.

To achieve higher performance we divide the data into subsets based on it's features and characteristics and train a classifier for each one.

CONCLUSIONS

Because of the versatility of the Scila software the choice of parameters are not unique, but rather very much dependent on how different clients use the system.

From our evaluation we conclude that many possibilities exist and it is up to the client to determine what is most useful to them.

SVM is definitely an interesting and useful tool for this purpose. We achieved results good enough to produce prioritized lists and aid a human supervisor, but not to act as an stand-alone classifier.

CONTACT INFORMATION

Jens Wirén
Farhad Kimanos

KTH Royal Institute of Technology

Department of Computer Science and Communications

Lindtedtsvägen 3, 5
100 44 Stockholm

Phone: 08-790 60 00

www.kth.se

Introduction

Machine Learning is a field in computer science that has been growing exponentially during the recent years. In general it can be defined as a branch of artificial intelligence focused on the construction and learning of computational systems.

Apart from massive data mining and pattern recognition where human capacity is simply insufficient machine learning can be used to drastically reduce human workload and automate processes where a more dynamic decision making is required.

Scila AB

This project will be carried out on the request of Scila AB, which is a company within the field of financial surveillance development. Their product is integrated into the customer's system where it looks for suspicious anomalies. When such is detected a human operator is alerted, which initiates an investigation that in turn leads to a classification of the alarm as valid or false. Among the pool of alarms raised during an average business day there are very few that are in fact valid.

Goals

The task appointed to us is to implement a learning model that can learn from the history of decisions made by human operators to try to predict a future decision, there by reducing the workload for the operators.

Related work

The problem is well suited for a number of different machine learning algorithms and the first one is the *Naïve Bayes* classifier which uses prior statistics and Bayes theorem to calculate posterior statistics. The third method investigated is *Artificial immune systems* where you define a number of detectors that searches for patterns and if a detector successfully classifies an input it's weight is incremented and opposite and vice versa. Both these methods tends to becoming biased when the distribution between classes is uneven.

The second method is *Neural Networks* which mimic neurons firing in a network. It can solve complex problems but at the price of computational cost and that it acts like a "black box".

Support Vector Machines

The final method is called *Support Vector Machines* and this method attempts to separate classes with a hyperplane. Every input you want to classify consists of a number of values. These can be seen as base vectors in an n-dimensional room. Consider a 2-dimensional vector space and assume that the classes are linear. Then a line exits that can separate the two classes and only the points which lie on this line is relevant. These are known as *Support Vectors.* We only need to compare a new input with these points. The line separating the two classes can be written as:

$y = \bar{w} \cdot \bar{x} + b$ and defining the margin as: $M = \frac{1}{2\sqrt{\bar{w} \cdot \bar{w}}}$

We can formulate this as an optimizing problem where we want to maximize the margin:

$$minimize \quad \bar{w} \cdot \bar{w} + C \cdot \sum_{i=1}^{l} \varepsilon_i^2$$

$$subject\ to \quad y(\bar{w} \cdot \bar{x} + b) \geqslant 1 - \varepsilon_i, i = 1, \ldots, l$$

where the constraint ensures that an input is in the correct class. Here ε is a *slack variable* which allows some error. This is necessary to create a useful classifier. Otherwise it will be perfect on the dataset it is trained on but generalize poorly.

The genius of SVM is the fact that it is always a convex optimization problem. This means that the maximum margin can now be found using Lagrange multipliers and by solving the dual representation of our problem, because the duality gap is always zero. In this dual representation the input data will only exist in an inner product with itself. This can be exploited to switch space by changing the inner product to one that can solve a higher dimensional problem, such as the Radial Base Function or Gaussian kernel.

Implementation

It was early concluded in the project that to improve the accuracy we had to train separate SVMs specialized at specific subsets of the data. These subsets consists of specific alarm types that were selected primarily based on number of occurrence and manageability of features. Furthermore to find the most suited parameters a grid-search was employed, where the confusion matrix of each point was analyzed using some accuracy measurement.

Each grid-search point was cross-validated to improve the generalizability of the trained model.

In our case false alarms makes up about 22 times more data points than valid alarms. Due to this unbalancedness one has to increase the misclassification costs of the optimization problem for the minority class. Otherwise the solution will strongly favor the correct classification of the more numerous false alarms and almost completely neglect the less frequent valid alarms.

Results

The alarm type "Ramping 0,3%" was one of the four alarm types that was selected for the analysis, due to number of occurrence and convenient of trigger parameters (features).

The performance of the classification was also tested for different positive class weights apart from the most obvious one, which is the occurrence ratio between the two classes. It was shown that the optimal weight lies close to the theoretical for the selected dataset.

We did in general reach an accuracy of 60-75% of correctly classified data points depending on the dataset. The best points corresponded predominantly to peak BAC.

Conclusions

We found that most available accuracies favored the correct classification of the majority class, if it was overwhelmingly more numerous. Positive and negative recall for instance takes little effect of misclassifications of the opposing class. However while the three less refined measurements used in this project (precision, positive and negative recall) have great limitations, they can be used to draw qualitative conclusions about the three remaining more complex measurements, namely F-score, MCC and BAC.

As mentioned, BAC was concluded to be the superior accuracy measure of the three in our case. Since it maximizes both negative and positive recall for the same parameter set.

Regarding the extreme outliers we conducted several test both with and without them. No significant effect was observed leading us to conclude that the outliers were to few to significantly disturb the data.

One of the main drawbacks of SVM is the huge computational time. We executed the calculations of one of the mainframe computers of Scila with 64 high-end cores. Despite full access to all cores the computational time of one 40-by-40 gridsearchs for about 10 000 datapoints could span up to several hours. Thus some gridsearch points were dropped to get results within reasonable time.

In conclusion SVM can definitely be useful to ease the burden of human operators. However, to make this practically viable the process of training and updating classifiers must be more or less autonomous. Furthermore it have to be updated along with new data over time. Exactly how this could be implemented was not investigated in this project and are left for future development.

Even thou quite impressing results were achieved with SVM, it would be of great interest to look for alternative techniques, such as neural network or time-series evaluation.

Dual Form and Kernel Functions

Using Lagrange multipliers we can formulate the dual form of the optimization problem:

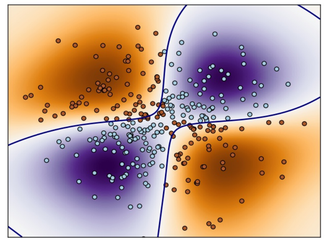
$$maximize \quad L(\bar{w}) = max\Sigma_{i=1}^{l}\alpha_i - \frac{1}{2}\Sigma_{i=1}^{l}\Sigma_{j=1}^{l}\alpha_i\alpha_j t_i t_j \bar{x}_i \cdot \bar{x}_j$$

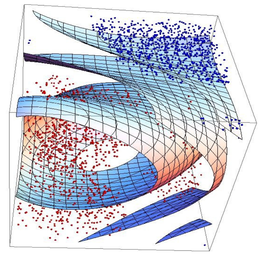
$$subject\ to \quad 0 \leq \alpha \leq C \qquad and \quad \Sigma_{i=1}^{l}\alpha_i \bar{x}_k = 0$$

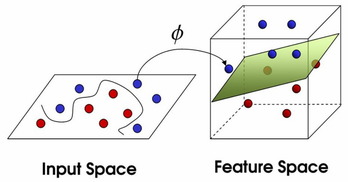
where the input vectors only appear in an inner product. This means we can change the representation of the data and allow for a more complex separation using a kernel. The most common one is the RBF-kernel defined as:

$$\bar{K}(\bar{x}_i, \bar{x}_j) = exp(-\gamma(\bar{x}_i - \bar{x}_j)^2)$$

and examples of hyperplanes are shown in the pictures.





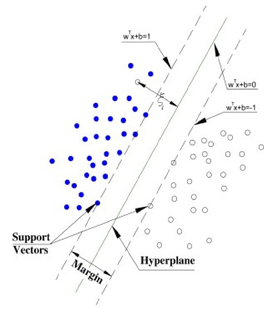


Separating Data by Changing Feature Space

Many datasets are not linear and cannot simply be separated by a linear classifier. This problem can always be circumvented by mapping the data onto a higher dimensional space. If this space is chosen appropriately the data can always be made separable.

A Soft-margin Classifier in a Linear Feature Space

This picture shows a linear classifier where to classes are separated by a two-dimensional hyperplane, i.e. a line. The margin is defined as the maximal separation distance between the classes. Data points on the margin are called Support Vectors and contain all the information needed to classify a new input. Just above the centre in the picture an input has been wrongly classified and the slack variable ξ is a measure of this. This allows for an SVM with good accuracy that can generalize well.



Grid-search and Accuracy Measurements

To find the C/γ-parameters best suited for a specific dataset a grid-search is employed. The performance of each point is evaluated using accuracy measurements based on the confusion matrix of the cross-validation. The figure shows heatmaps over the six different accuracy measurements used in this project for a grid-search of 40-by-40, thus 1600 different points. The three top heatmaps displays precision, positive and negative recall, that are in essence measurements of different aspects of the classification accuracy. While the three bottom F-score, MCC and BAC shows more of an overall accuracy of the model.

In our case the balanced accuracy (BAC) was by far the best measurement, due to its ability to take the recall of both classes into consideration, thus suffer less from unbalancedness between the two classes.

F-score and MCC were able to pinpoint regions with high precision, which renders classifiers with low accuracy for the minority class. In our case this is less satisfactory.