**Midterm Report**

**Financial time series forecasting using support vector machines**

## **Objective**:

The objective of this paper is to analyze the feasibility of Support Vector Machines (SVMs) in predictive analysis of the financial sector. In particular, this paper studies the capacity of SVM to predict the stock price index. Further, this paper also compares SVMs to Back Propagation Neural networks and Case-Based Reasoning in financial forecasting.

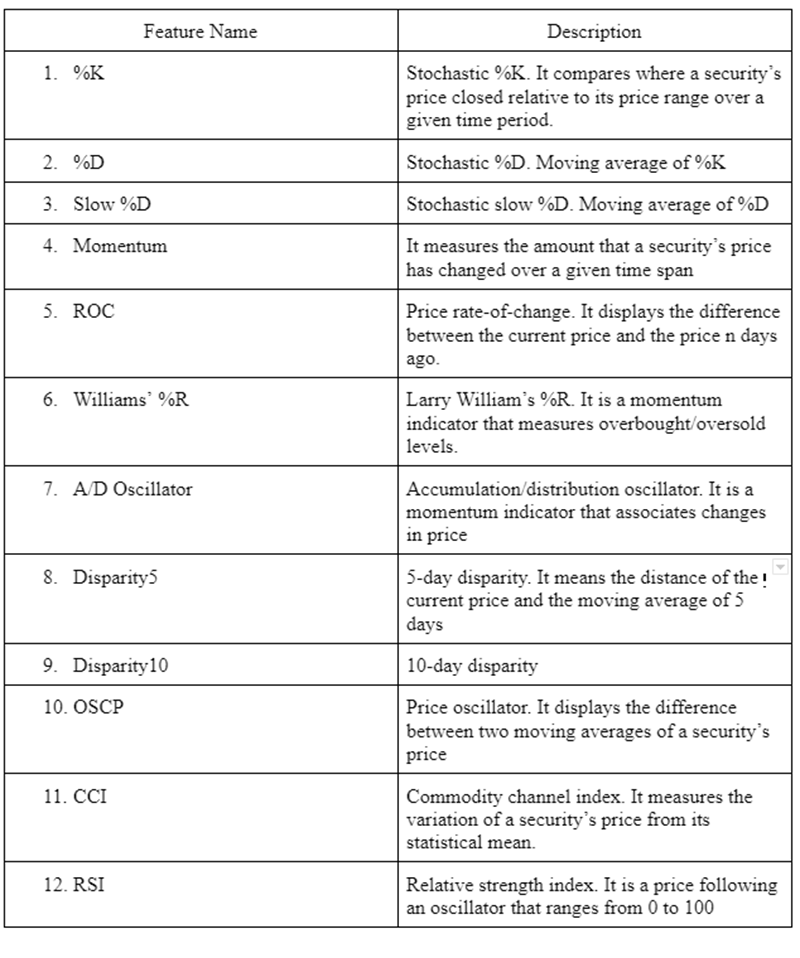
The paper specifies some reasons for analyzing the SVMs. One of the main reasons for considering SVMs is that they come with an embedded risk function derived from the structural risk minimization principle. This principle leads to both an empirical error term as well as a regularized term in the risk function. Regularization helps in better generalization. Moreover, SVM may find a global optimum solution while a neural network may find only a local optimum.

The paper also cites studies that explain the drawbacks of existing methods like Artificial Neural Networks (ANNs). For example, the dimensionality of stock market data is complex. Furthermore, that data is noisy too. ANNs perform poorly on noisy data. Further, backpropagation neural networks face difficulty in selecting too many parameters. Whereas in the case of SVMs, there are no tuning parameters except the upper bound 'C' on the coefficients.

## **Methods**:

The basic idea of SVMs is to create nonlinear class boundaries by mapping input vectors into higher dimensional space. Functions that perform this mapping are called Kernel functions. Kernel functions considered by the paper are Polynomial () and Gaussian Radial Function ().

For each data point, the paper used 12 technical indicators as input variables/features.The output is the change in the stock price index. The 12 technical indicators are presented in Table 1 below.



The output that is predicted by the study is the direction of daily change in the Korean Composite Stock Price Index (KOSPI). The two classes to be separated by the SVM are labeled as 1 and 0. 1 means that the index rose the next day and 0 means it fell the next day.

2928 samples are used in the study. They are the daily data from January 1989 to December 1998. Out of these, around 80% was used for training and the remaining was used as holdout/validation data. To avoid the domination of large valued features, each feature was normalized to the same range [-1.0, 1.0].

The formula used to calculate the prediction performance P:

Ri where Ri is 1 when the prediction is true and 0 when it is false. ‘m’ is the number of test samples.

### SVM:

As the paper considered Polynomial and Gaussian Radial kernel functions, 2 hyper parameters are considered important for the SVM’s performance. 1. Upper bound C 2. Kernel parameter . To perform experiments with different values of hyper parameters, LIBSVM software was used.

### Backpropagation:

| Parameter | Value |
| --- | --- |
| Number of hidden layers | 3 |
| Sizes of hidden layers | 6,12,24 |
| Stopping criteria | 50,100,200 epochs |
| Learning Rate | 0.1 |
| Momentum | 0.1 |
| Transfer function for hidden nodes | Sigmoid |
| Transfer function for output node | Linear |
| Number of input nodes | 12 |

### Case-Based Reasoning (CBR):

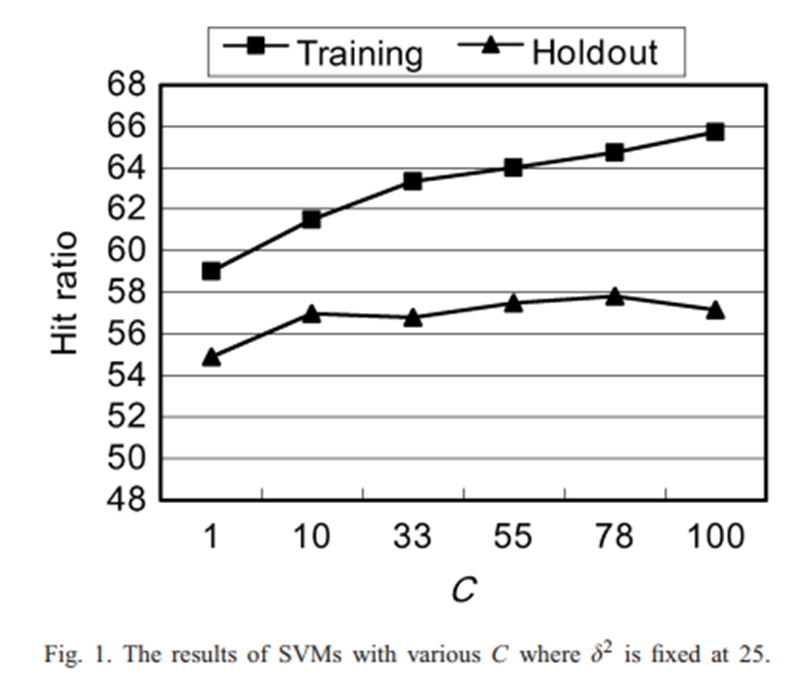
As the data is numerical, the nearest neighbor method which can be easily applied to numerical data is used. The evaluation function using Euclidean distance is as follows:

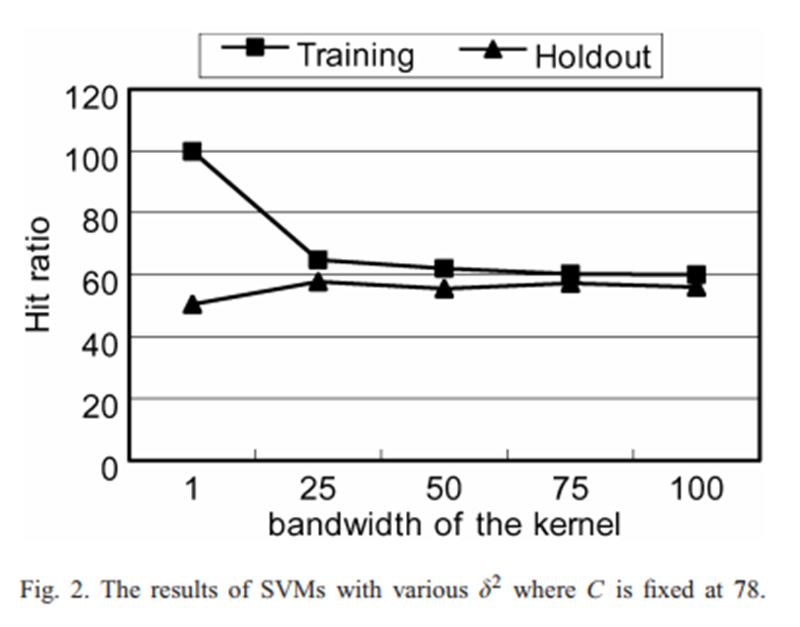
: Distance between

## **Results**:

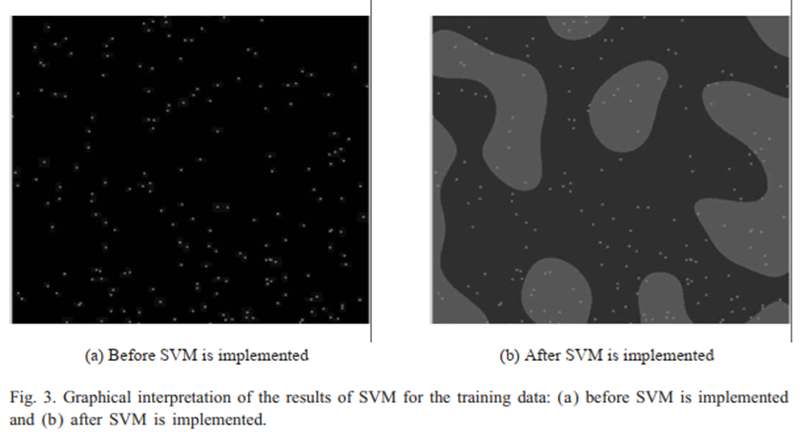
After comparing the performance of polynomial and Gaussian kernel functions, the paper chose Gaussian kernel for the SVMs. Using the studies done earlier by Tay and Cao, the paper chooses the values of between 1 and 100 and the values of C between 10 and 100. Among these values, best prediction performance on holdout/validation data was when is 25 and C is 78. Prediction performance at this best point is 57.8313% on holdout data.

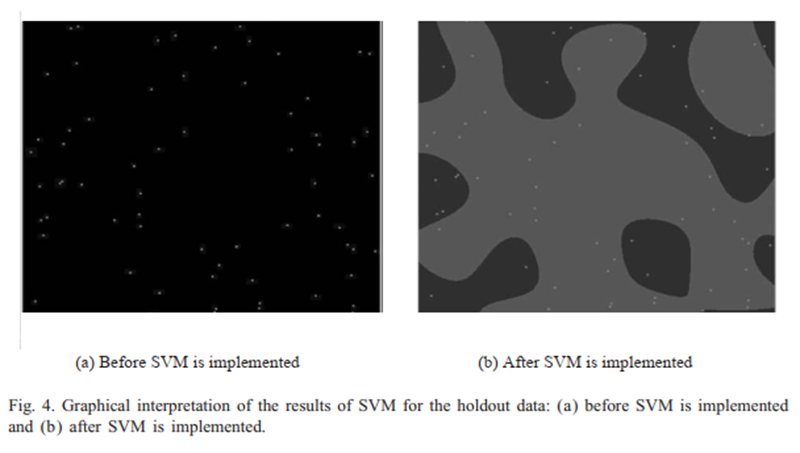
By fixing one of and C at their best performance values and varying the other, the paper verified that the results support the conclusions of Tay and Cao partly. For example, though the performance on training data increases with C, performance on holding data peaks at C= 78. This can be explained in the terms of bias-variance trade-off. C is an upper limit on the complexity of the parameters/weights. So, a higher C will allow more complex weights/ models, which will lead to overfitting. Similarly, the paper confirmed the the Tay and Cao’s conclusion that small value of would over-fit the training data and large value underfits. Figure 1 and 2 display this phenomenon.



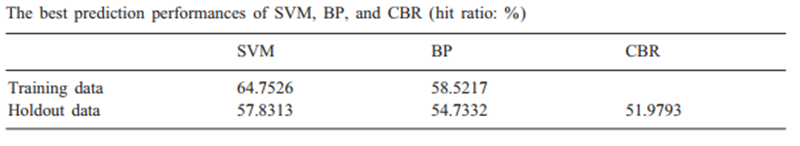


The following figures, Figure 3 and Figure 4 depict the data points before and after the separation into two classes by SVM. We can see how SVM can classify non linearly separable data into two separate classes.

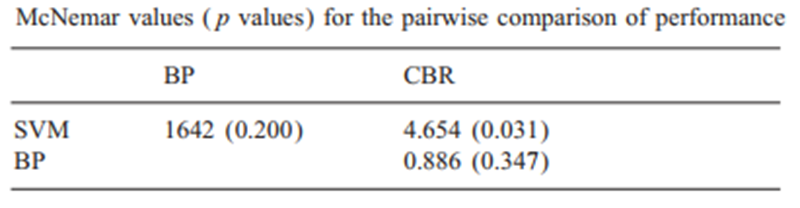




This study then compares the best SVM model with the best backpropagation (BP) and case based reasoning (CBR) performances. The below table shows the comparison. Clearly, SVM outperformed backpropagation and case based reasoning. This, SVM is shown to be viable tool in forecasting financial time series.



Finally, as shown below, McNemar tests show that SVM beat CBR at 5% significance level. However, SVM does not significantly do better than BP.



## **Conclusion**:

Various experiments conducted using SVMs showed that prediction of SVM is heavily influenced by the values of upper bound C and kernel parameter . Results also show that SVM performed better than BPN and CBR. This may be due to the structural risk minimization inherent in SVM design leading to better generalization. Finally, the paper concludes that SVM is a strong alternative for the forecasting of financial imer series.

## **What new can be proposed:**

This paper uses standard SVM. However, the cost function of standard SVM is computationally expensive and not viable for large data sets. So, more efficient methods can be used for training like Least Squares SVMs (LS-SVMs), SVMs using Sequential Minimal Optimization (SMO) etc. Similarly Generalized Eigenvalue Proximal Support Vector Machine (GEPSVM) and Twin Support Vector Machine (TWSVM) can be used which outperform standard SVMs. As discussed earlier, financial time series prediction faces noisy data and non-stationary information. Literature provides Twin Support Vector Regression (TSVR) to handle such issues better than standard SVM.