STOCK SELECTION USING GENERAL GROWING AND PRUNING RADIAL BASIS FUNCTION (GGAP-RBF) NEURAL NETWORK

NG WEE DING

SCHOOL OF ELECTRICAL & ELECTRONIC ENGINEERING

A DISSERTATION SUBMITTED TO THE NANYANG TECHNOLOGICAL UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE IN COMMUNICATION SOFTWARE AND NETWORKS

2006

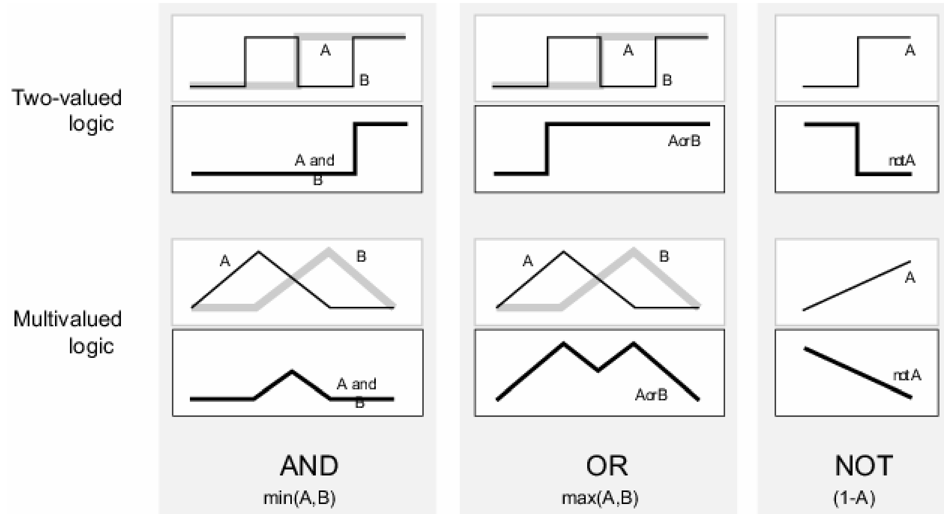

Figure 7: True Table for Fuzzy sets

(directly extracted from MATLAB Help [19]*)*

Given these three functions, we can resolve any construction using fuzzy sets and the fuzzy logical operation *AND*, *OR* and *NOT*. In short, there are three steps in fuzzy system, first is to fuzzify inputs based on the rules, subsequently applying logical operators on the fuzzified inputs, lastly apply implication operator to shape the output fuzzy set.

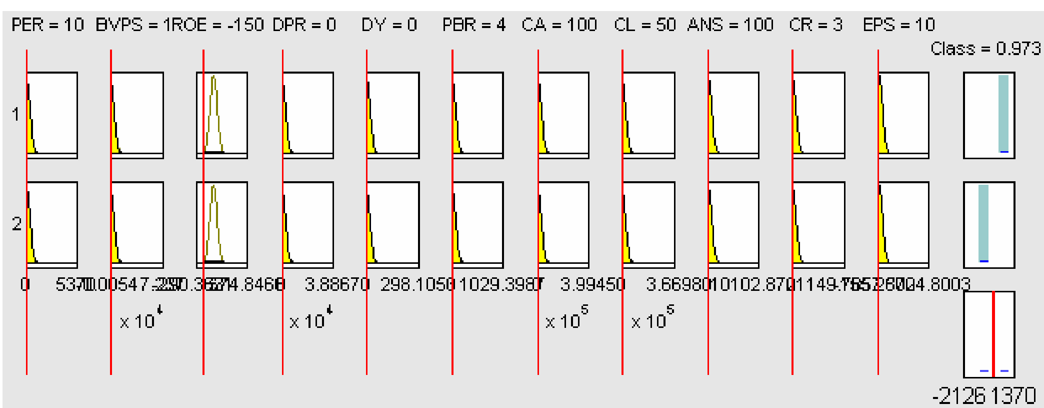

Figure 8: Fuzzy Inference System

Fuzzy inference is the process of formulating the mapping from a given input to an output using fuzzy logic. We present the fuzzy inference diagram of our equities picking problem in Figure 8. There are two rules with 11 attributes for each role. Every attributes will have a member function to fuzzify the input of the attributes. For

4.1.3 Networks Training

The soft computing models under study are Multi-level Perceptrons (MLP) feed-forward neural network, Adaptive Neuro-Fuzzy Inference Systems (ANFIS) and General Growing and Pruning Radial Basis Function (GGAP-RBF).

4.1.3.1 Multi-level Perceptrons (MLP) feed-forward neural network

MLP is configured with the number of hidden neurons being two-times of the input layer, which is twenty-two neurons (eleven attributes times two). The training algorithm is gradient descent with momentum and adaptive learning rate. Both the hidden layer neurons and output layers neurons have tangent sigmoid activation functions, which have the output values between -1 and +1. This MLP configuration can be visualized with Matlab Neural Network Graphical User Interface (GUI), as shown in Figure 10. Please refer to *Chapter 2.1.1.1* for the syntax and semantics of this MLP structure.

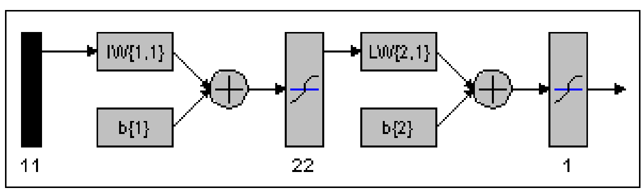

Figure 10: MLP structure

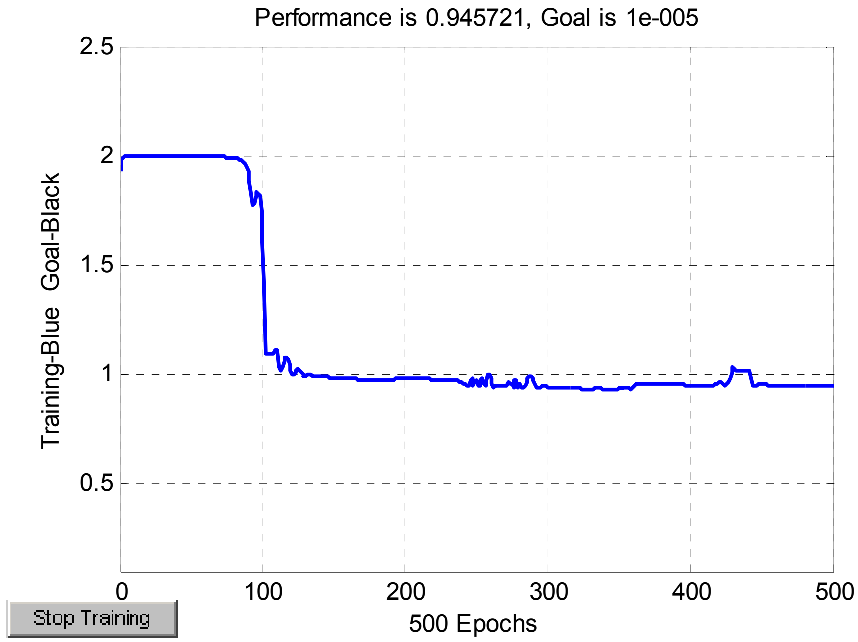

Figure 11: MLP Training

Figure 11 shows the MLP training mean squared error (MSE) against training epochs. The initial MSE is about 2 and it is stabilized at MSE of 0.945721. According to Figure 11, the actual epochs required to reach the possible minimal training error is actually less than 150. The CPU computational time for training with MLP with the above configuration is 188.45 seconds[1]. Therefore, the chosen maximum training epochs for MLP are 500.

[1] Simulation environment: CPU: 1.5GHz; Memory: 768MB; OS: Windows XP.

4.1.3.2 Adaptive Neuro-Fuzzy Inference Systems (ANFIS)

ANFIS is configured using subtractive clustering with a radius of 0.20. And, it is trained for 10 epochs. The trained ANFIS model has two rule nodes, each nodes is represented as a locally-defined linear functions. The CPU computational time for training with ANFIS with the above configuration is 396.85 seconds[2].

We can visualize the ANFIS structure with Matlab Anfis Editor *anfisedit*. We present the ANFIS structure in Figure 12, the developed rules and the membership function after training is shown in Figure 13 and Figure 14 respectively.

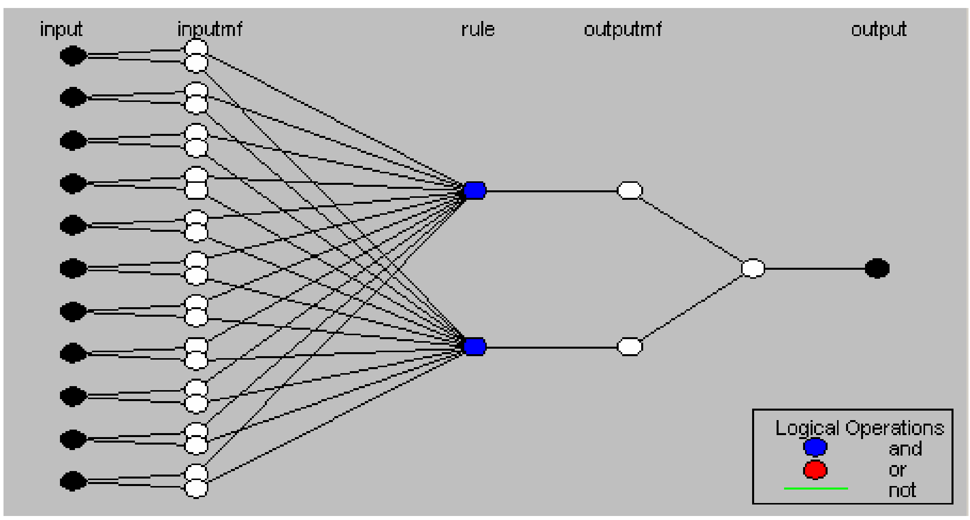

Figure 12: ANFIS structure

Figure 12 presents the ANFIS structure. The inputs comprise eleven attributes that are chosen in *Chapter 3.1*. Please refer to *Chapter 2.1.2* for the syntax and semantics of ANFIS structure.

We have two rules for the ANFIS system. The rules govern the fuzzy inference process from inputs to outputs, as explained in Section *2.1.2 Introduction of Fuzzy Logic.*

[2] Simulation environment: CPU: 1.5GHz; Memory: 768MB; OS: Windows XP.

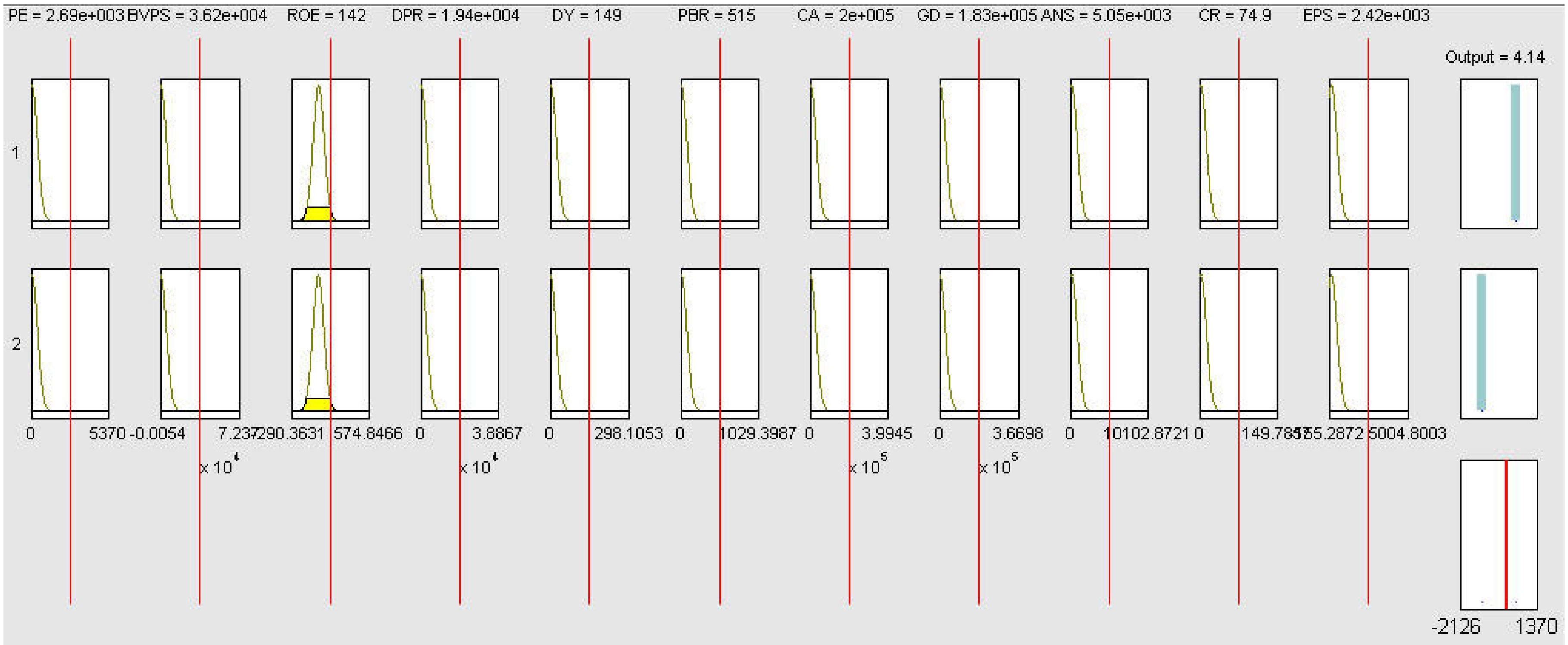

Figure 13: The two rules of ANFIS system, for the fuzzy inference process from inputs to outputs. Each row of plots corresponds to one rule and each column of plots corresponds to either an input variable (yellow) or an output variable (blue).

After 10 epochs of training, the final ANFIS which has the minimum checking error will be chosen. The new membership functions for the eleven attributes are shown as below: -

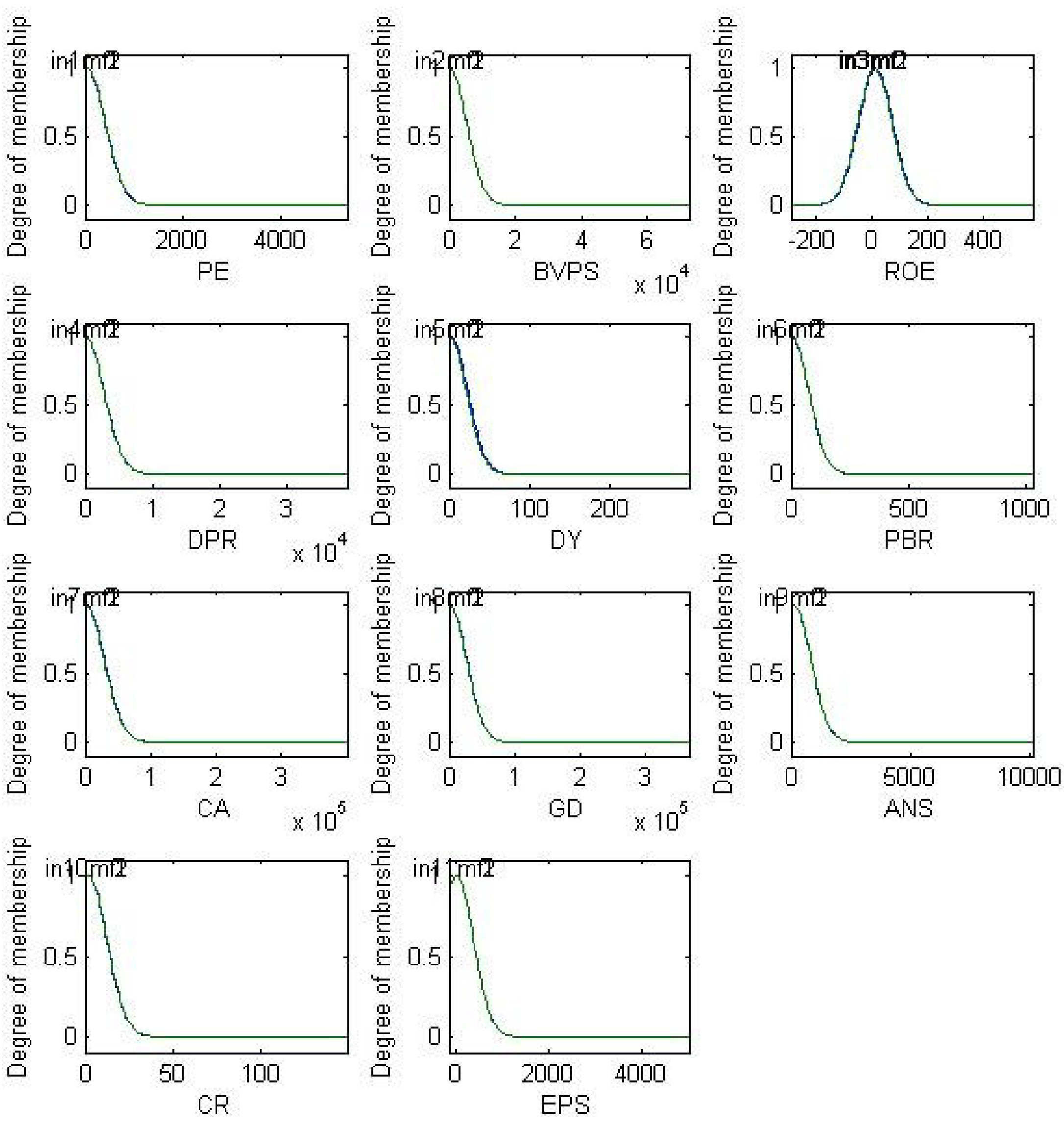

Figure 14: Membership functions for ANFIS attributes after training.

(Horizontal-axis shows the degree of membership)

The root-mean squared error for the training data has been shown as below: -

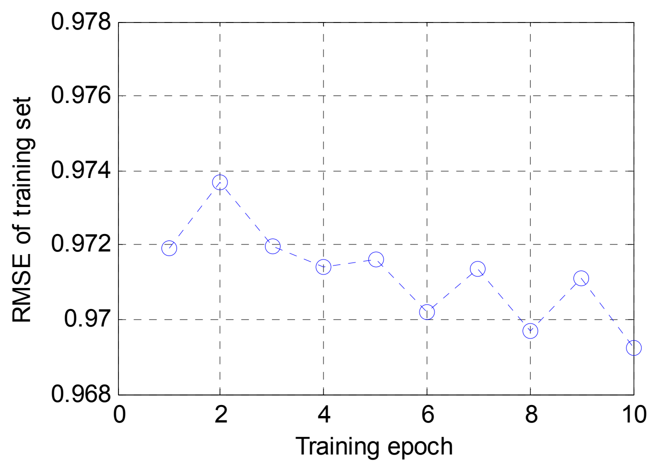

Figure 15: RMSE of ANFIS training

The oscillations after epoch 4 are due to the heuristic rules to increase and reduce step size in ANFIS training algorithm [17].

The training algorithms among these three models, namely MLP, ANFIS and GGAP-RBF, are different in nature. The training error representations between Figure 11 and Figure 15 have different meaning that can not be directly comparable with each others. As an example, the training of ANFIS is affected by the increment and decrement of step size in Figure 15; and the training of MLP is affected by the momentum setting. Figure 11 and Figure 15 provides general idea on how training error curves are. Furthermore, the performance of MLP, ANFIS, and GGAP-RBF are compared from various perspectives in the subsequent sections. Therefore, the training curves comparison for these three models is not a necessity.

4.1.3.3 General Growing and Pruning Radial Basis Function (GGAP-RBF)

GGAP-RBF which has been proposed by [11] in 2005, will be used to study our problem. We applied the provided MATLAB source codes by [11] for training. The CPU computational time for training with GGAP-RBF is 360.7 minutes (21,642 seconds)[3]. The time complexity for GGAP-RBF is obviously too high as compared to MLP and ANFIS, which spent 188.45 seconds and 396.85[4] seconds respectively.

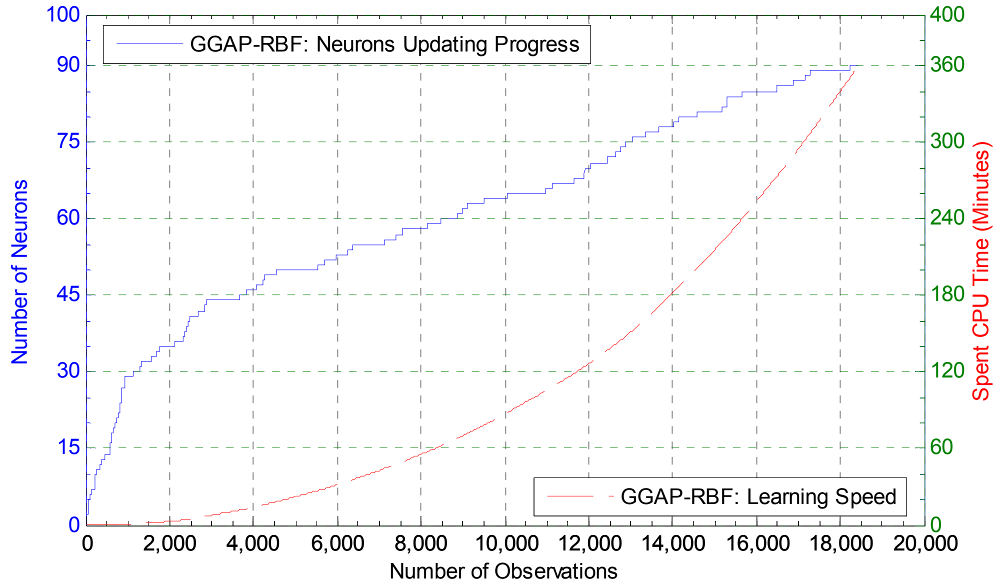

Figure 16: **Learning speed and neuron updating progress for GGAP-RBF**

To understand this, we plot the learning speed and neuron updating progress against the number of inputs as shown as the figure above. For this problem, the [11] proposed GGAP-RBF [11] obtains a total of 90 neurons after the six-hour training. The time complexity is exponential. The more the neurons are added, the slower the algorithm work. As you can see, the first-half of the training set, which comprises about 9,224 input rows, occupied the first hour of the total training CPU time. On the other hand, the second-half of the training set occupied about five times more than first-half of the training data on the spent CPU time on training. This shows that GGAP-RBF does not scale well with the numbers of inputs although there are usually large numbers of instances for financial problems.

The above is summarized as below: -

Soft-computing models	Computational Time (for training)	Descriptions
MLP	188.45 seconds	Training algorithm: Gradient descent with momentum;

[3] Simulation environment: CPU: 1.5GHz; Memory: 768MB; OS: Windows XP.
[4] Simulation environment: CPU: 1.5GHz; Memory: 768MB; OS: Windows XP.

Recall is the process of putting input data into a trained network and receiving the output; subsequently compare the output with the desire output.

We plotted the 18,448 training input instances with regards of its predicted output for the three soft computing models under studied, as shown as Figure 17, Figure 18 and Figure 19. Let us look at each of them in details.

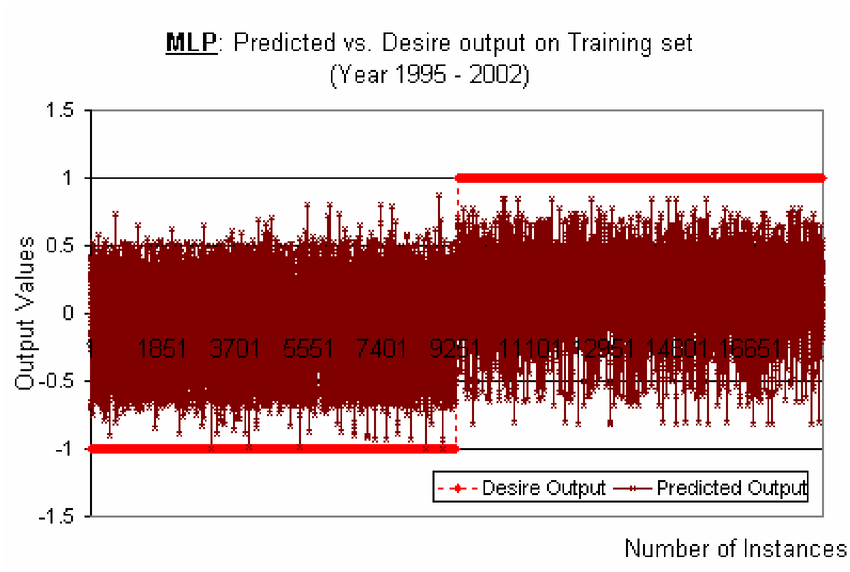

Figure 17: MLP: Training results (training set for the year 1995-2002)

Refer to Figure 17, the desire outputs of total 18,448 training input instances have been sorted in order to form the step function-liked pattern. The desire outputs value is either "Class 1" or "Class 2", which is numerically represented as -1 and +1 respectively. And, there are half of the "Class 1" data (as shown as +1) and half of the "Class 2" data (as shown as -1). That means the data is sorted in such a way that the "Class 1" instances are grouped to form step-up (+1) pattern and "Class 2" instances are grouped to form step-down (-1) pattern. We re-apply the training input instances to the trained networks, and obtain the predicted outputs. The predicted outputs' values are fuzzy and the range of the values are dependent of the soft-computing models, for example the output values of our MLP is in the range between -1 and +1, as the transfer function being used is tangent sigmoid, which has minimum output value of -1 and maximum of +1.

We expect that the predicted output values should tend to have higher values for instances which have desire outputs of +1 (right-hand-side of the step-function-liked the plot), and vice-versa. The training algorithms can be concluded to be not useful if there is no such relationship.

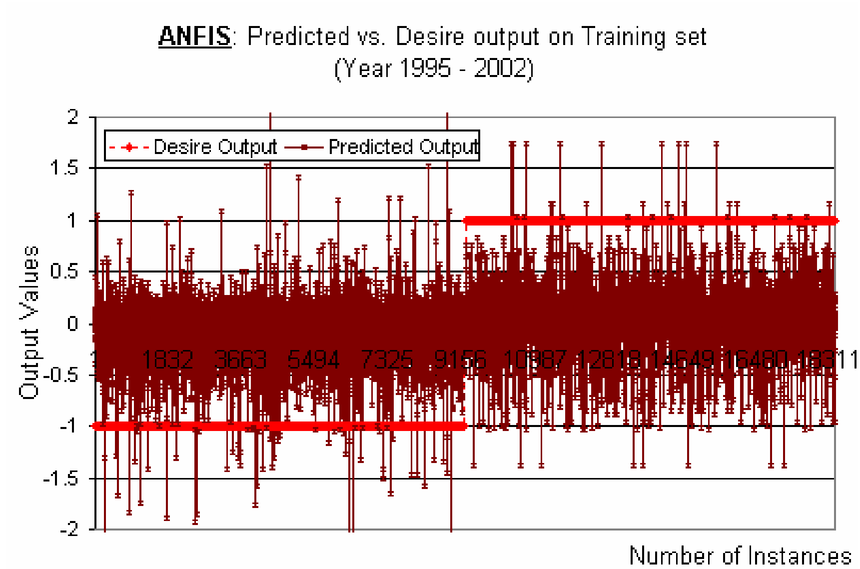

Figure 18: ANFIS: Training results (training set for the year 1995-2002)

The outputs of ANFIS are by linear function and the output values may be out of the range of -1 and +1, as shown in Figure 18. There is observable slightly upward trend for the predicted output values from the range between -0.6 and +0.25 to the range between -0.5 and +0.5. If we only consider the predicted output values which are above +0.5 (use your hand to cover at the +0.5 y-axis horizontally), the majority of the points are on the right-side of the step function, which reflects that majority of the predicted output having values equal or greater than +0.5, are having desired output of "Class 1"(+1). This shows that ANFIS has meaningful output for applying its trained network on training set, similarly to MLP.

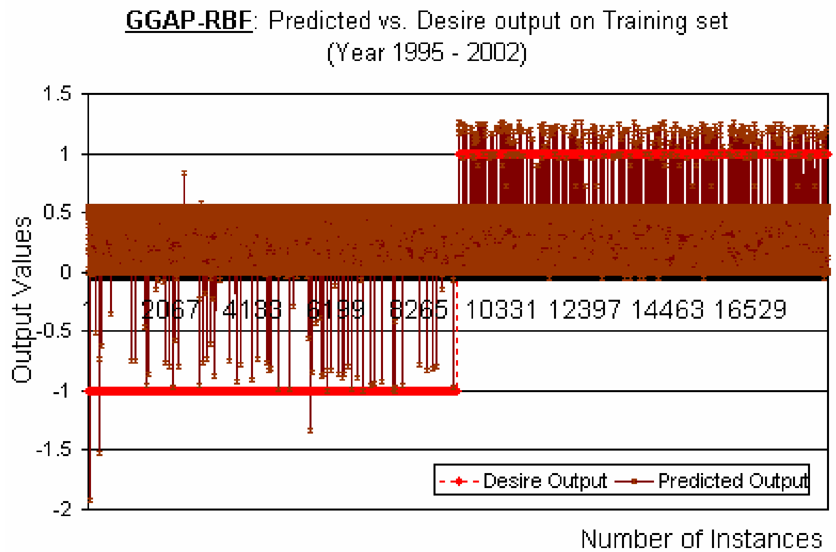

Figure 19: GGAP-RBF: Training results (training set for the year 1995-2002)

GGAP-RBF performs differently to MLP and ANFIS. Let's look at the left-side of the step function, which contains all the "Class 2" instances. Though majority of the instances having predicted output values in the range of 0 and +0.5, but there is almost zero instances having predicted output values greater than +0.5. Similar pattern can be found for all "Class 1" instances as well (right-side of the step function). What does this imply? Let us define the rules as such: First rule, for all predicted output values equal or greater than +0.6, mark them as "Class 1". Second rule, for all predicted output values equal or lower than -0.1, mark them as "Class 2". And, simple ignore those instances with predicted output values in between +0.6 and -0.1. As you can see from Figure 19, majority of the instances have predicted output values in the range of +0.6 and -0.1, and thus will be ignored. However, the identified "Class 1" and "Class 2" instances, based on the defined two rules, are having almost 100% of accuracy. This shows that GGAP-RBF has the most desirable outcomes for recall rate.

Refer to Figure 17, Figure 18 and Figure 19, we see that there is such positive relationship between the predicted outputs and desire outputs for all three models. We see that there is a shift up-trend for the predicted outputs from "Class 2" to "Class 1" for all three models. This shows that the chances to accurately classify the data are more than 50%. (The probability to correctly flip a coin is 50-50, hence we can only conclude the existence forecasting ability only if the chances for success is more than 50%).

4.2.1.2 Test Results

Good recall rate is meaningless as the models are trained and tested with the same set of instances which results in high level of bias. What we are interested in is not recall rate, but test rate. Test rate applies out-of-sample data as the inputs to the trained network and compare the output to their known desire outputs. Out-of-sample data refers to the samples which are not used in training the models.

The test set of Setting 1 comprises last two years of data (total of 2,422 input rows). Here, the test set will be applied to form the inputs of the trained models. The same data arrangement as the previous section is applied, as shown as the Figure 20, Figure 21 and Figure 22.

The test set is an imbalanced data. The "Class 1" (+1) is minority data which is only at about 10% of the total test set. Thus, the step-liked function has a late step-up.

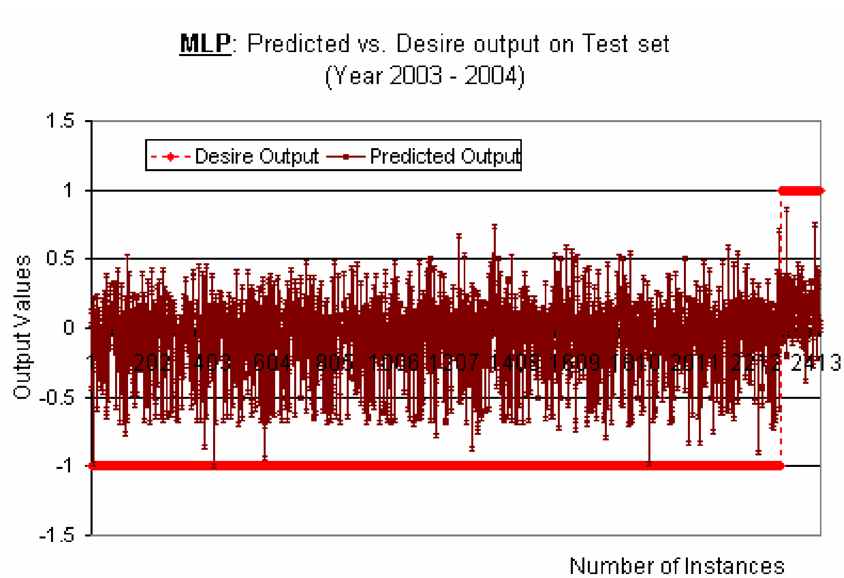

Figure 20: MLP: Test results (test set for the year 2003-2004)

Refer to the Figure 20, we see that for MLP model, there is slightly positive correlation between the predicted output values with the desire output values. Majority of the "Class 1" instances have predicted output values of equal or greater than zero. By observation, similar pattern can be identified in ANFIS model as well, as shown as the next figure. In the "Class 1" instances (which is on the right-hand side of the step-function that contains group of +1 desire output values), there is a chunk of predicted output, which have values fall into the range of 0 and +0.5. This chunk of data is slightly having higher values than the predicted output of those "Class 2" instances, and shows that there is positive correlation between predicted outputs and desire outputs.

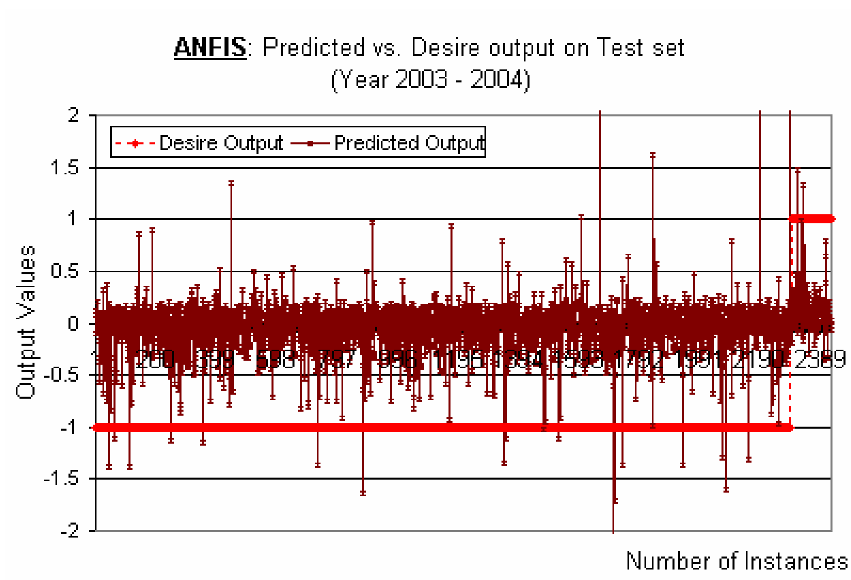

Figure 21: ANFIS: Test results (test set for the year 2003-2004)

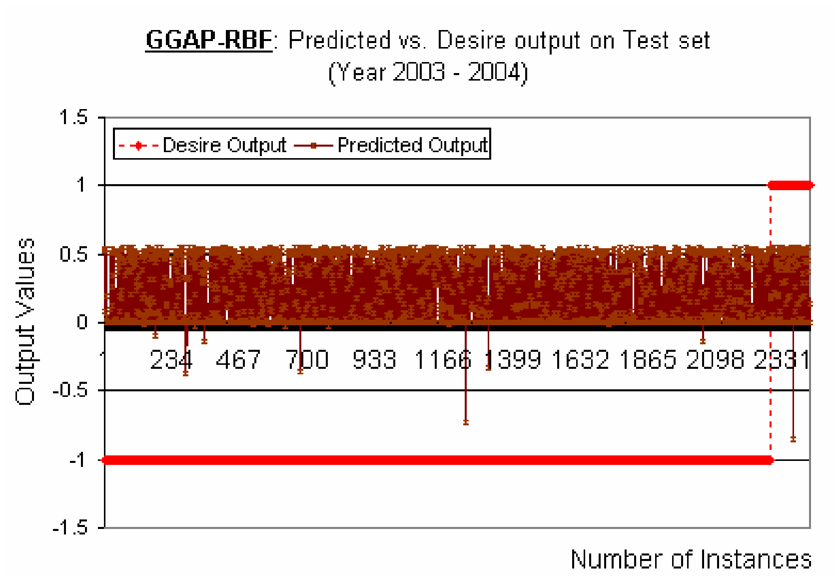

Figure 22: GGAP-RBF: Test results (test set for the year 2003-2004)

The test results for GGAP-RBF is different than MLP and ANFIS, as shown in the Figure 22. Similar to its training results, majority of the predicted output values fall into the range of 0 and +0.5. There are only a few instances which have values larger or equal to +0.6 and smaller or equal to -0.1. There is no distinct cut off values based on the predicted output to classify the training set. This shows that the GGAP-RBF has no ability on prediction in training set! It is no better than tossing a coin.

After studying the results on training set (recall) and test set (generalize), we found out that MLP and ANFIS performs similar in terms of accuracy of predictions while GGAP-RBF performs excellently for recalling its trained models with training set but extremely poor for generalization. The reasons may be due to one or more of the mentioned challenges of applying soft computing on financial problems: noisy and non-deterministic nature of the data.

If we simply apply 0 as cut-off-point for the output values to classify the equities into “Class 1” or “Class 2” for MLP and ANFIS, and we also simply apply 0.25 as cut-off-point for GGAP-RBF, then we can show the summary of the accuracies results as below: -

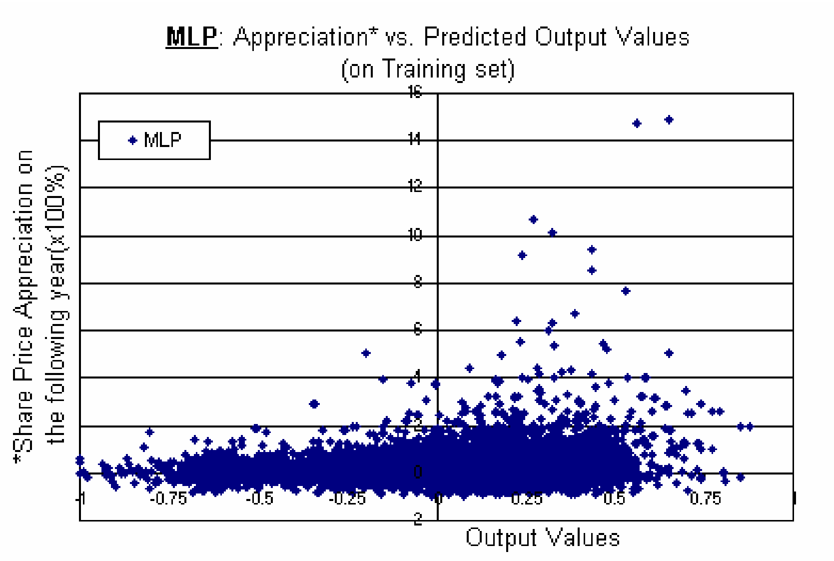

Figure 23: MLP: Actual appreciation vs. NN prediction (training set for the year 1995-2002)

As mentioned earlier, MLP trained network has predicted output values (neural network output values) in the range between -1 and +1, hence the x-axis is in the range of -1 and +1. The plot, as shown as Figure 23, skews towards +1, which reflects the good positive correlation with appreciation of equities value in the following year. We hereby expect that higher predicted output values shall reflect higher appreciation returns.

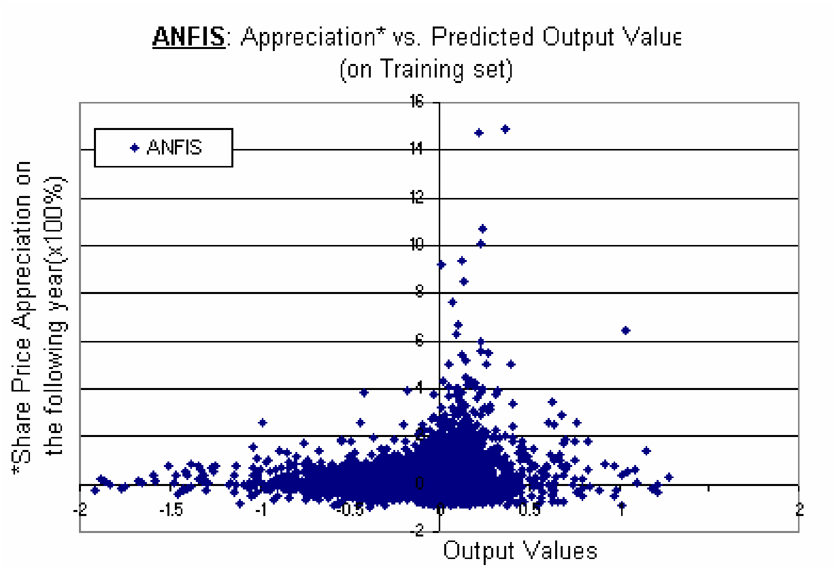

Figure 24: ANFIS: Actual appreciation vs. NN prediction (training set for the year 1995-2002)

Similarly, ANFIS model also show good positive correlation with the appreciation of the equities price in the following year. In contrast to MLP and ANFIS models, GGAP-RBF model gives a different scatter chart. It does not form a clear skewing curve, but we see some correlation relationship between the equities appreciation and the predicted output values around the x-axis values of -1 and +1 respectively, as shown as Figure 25.

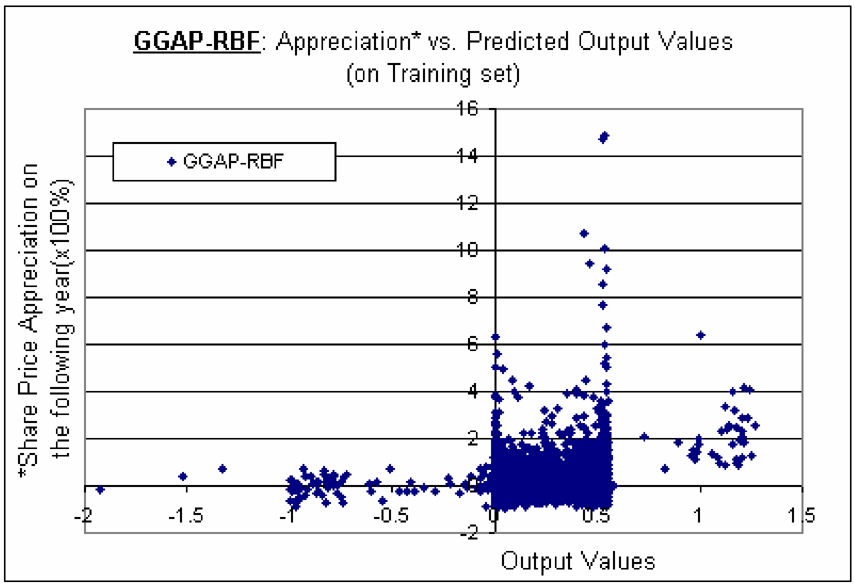

Figure 25: GGAP-RBF: Actual appreciation vs. NN prediction (training set for the year 1995-2002)

The explanation of the results presented in this section will be illustrated with numerical analysis in *Chapter 4.2.2.3*.

4.2.2.2 Test Results

By observation, similar pattern of correlation can be identified in test set as well for MLP and ANFIS models, as shown as Figure 26 and Figure 27. However, it appears that GGAP-RBF again has the lowest correlation or close to +0 correlations among the appreciation (Y-axis) and the predicted output values (X-axis), as shown as Figure 28.

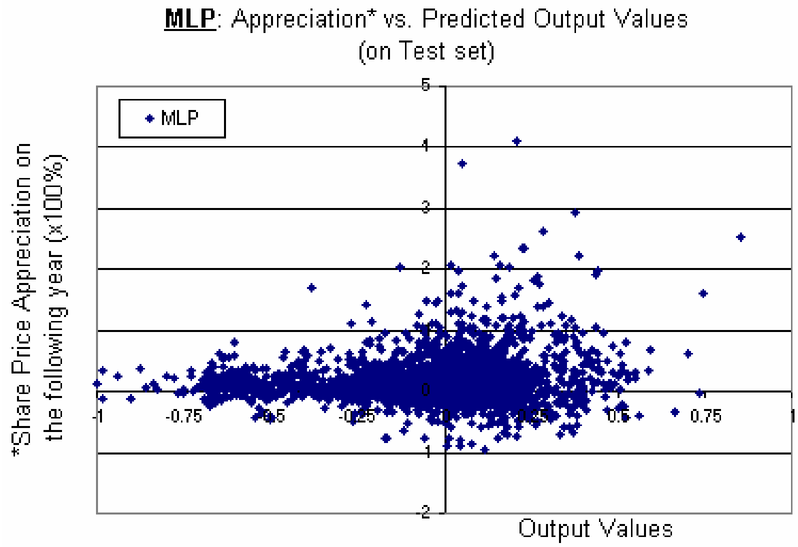

Figure 26: MLP: Actual appreciation vs. NN prediction (test set for the year 2003-2004)

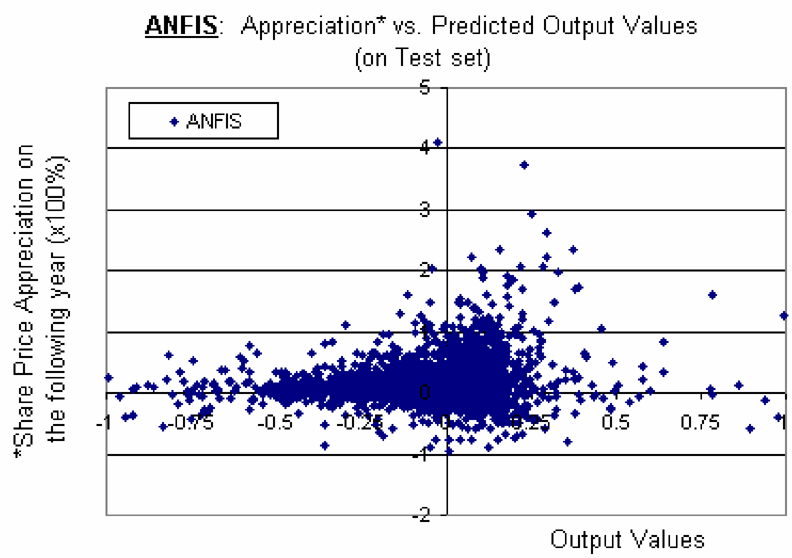

Figure 27: ANFIS: Actual appreciation vs. NN prediction (test set for the year 2003-2004)

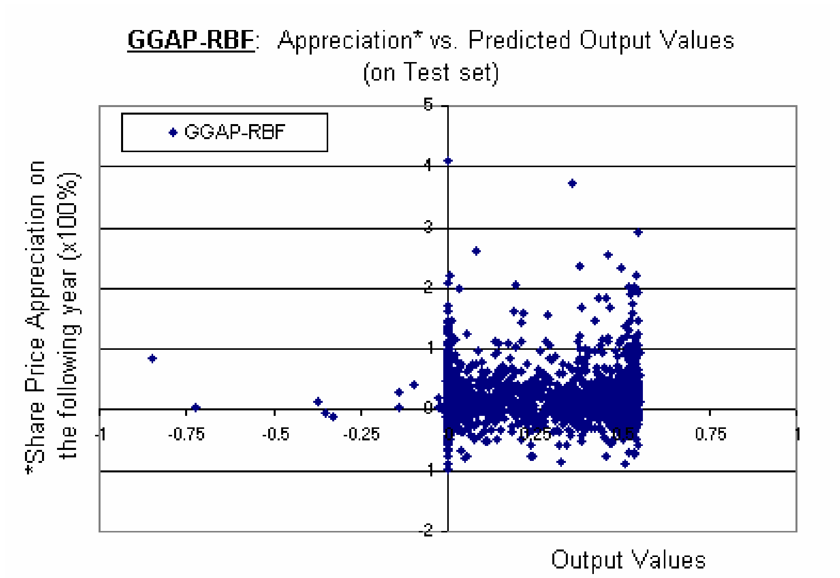

Figure 28: GGAP-RBF: Actual appreciation vs. NN prediction (test set for the year 2003-2004)

The explanation of the results presented in this section will be illustrated with numerical analysis in *Chapter 4.2.2.3*.

4.2.2.3 Results: Correlation

A quantitative approach to analyze the correlation between appreciation and predicted output values is using the Pearson correlation coefficient, which is defined in the following equation: -

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}}$$

where: r, the correlation coefficient may take any value between -1.0 and +1.0. The larger the value is, the higher the correlation is between two data vectors.

The correlation between the percentage of appreciation of the equities share price and the predicted output values, has been shown in the following table: -

The output values of neural network models fall into the range of -1 to +1 (as explained in *Chapter 4.2*), therefore we systematically apply all possible cut-off-points from -1 to +1 in a delta step of 0.01 to classify the prediction into either "Class 1" or "Class 2". Subsequently, we compute the average appreciation for the equities which have been predicted as "Class 1" (or, "winner"). The results of these experiments are presented in Figure 29, Figure 30 and Figure 31. For all the three figures, x-axis represents the cut-of-point. Y-axis of upper plot represents the average appreciation of the predicted "Class 1" equities whereas the y-axis of bottom plot represents the total number of equities has been predicted as "Class 1" with the respective cut-off-point. For example, if we have ten equities are picked or predicted as "Class 1" (or, "winner") equities with the cut-off-point of zero, we will compute the average appreciation of those picked equities' by summing all the ten equities' actual next-year appreciation and averaging them by ten. This average appreciation is shown in upper plot. The highlighted points in the figures are further explained in subsequent paragraphs.

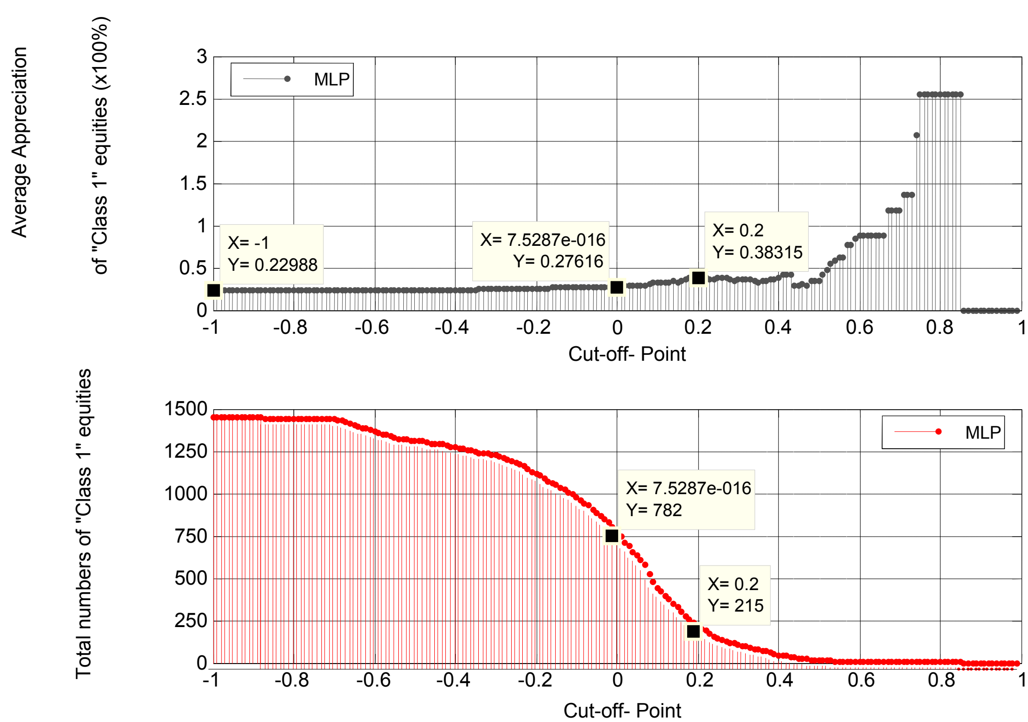

Figure 29: MLP: Appreciation vs. Cut-off-point (validation set for the year 2004)
(Average appreciation is calculated from selected "Class 1" equities)

Figure 29 demonstrates the impact of chosen cut-off-point over the average appreciation return from the picked equities. This experiment is done on validation set with trained MLP model. Let's pick the cut-off-point of zero. At the zero cut-off-point, there are in total of 782 equities have been signaled as "Class 1" equities. And the average appreciation of 782 equities is 27.616%, which is higher than the average appreciation of all the equities (22.99%). If 0.2 cut-off-point has been chosen, the total number of equities being classified as "Class 1" are further reduced to 215 from 782. The average appreciation of the 215 equities has been improved to 38.31% from 27.61%, which is much higher than the average appreciation of all the equities, 22.99%.

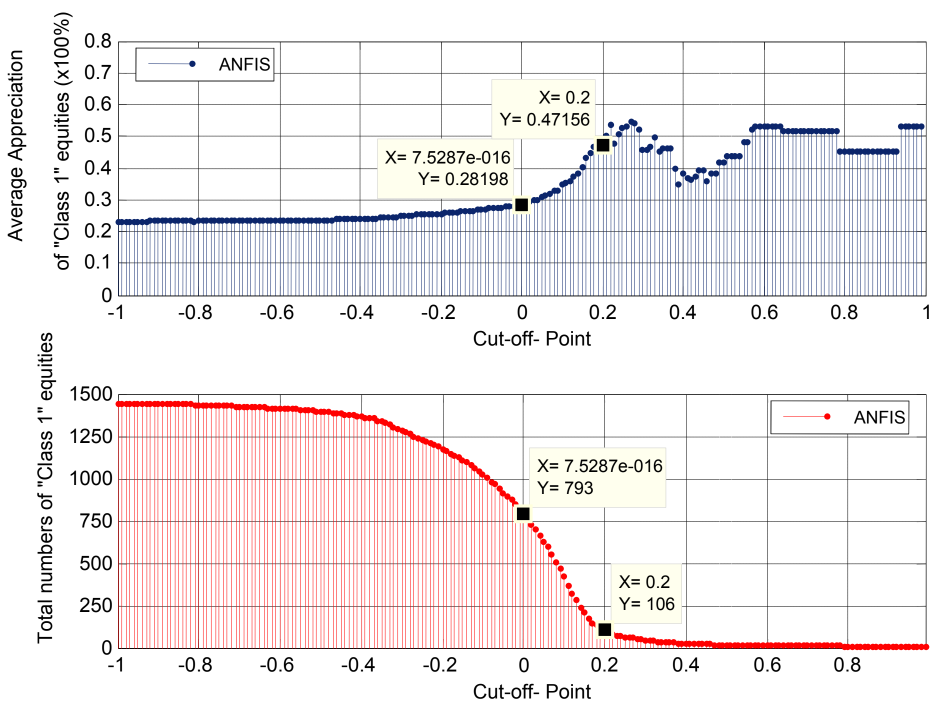

Figure 30: ANFIS: Appreciation vs. Cut-off-point (validation set for the year 2004) (Average appreciation is calculated from selected "Class 1" equities)

Figure 30 demonstrates the impact of chosen cut-off-point over the average appreciation return from the picked equities. This experiment is done on validation set with trained ANFIS model. As shown as Figure 30, the performance of ANFIS trained model is more desirable than MLP on validation set. At the zero cut-off-point, there are in total 793 equities that have been classified as "Class 1" equities. And the average appreciation of the 793 equities is 28.198%, which is slightly higher than MLP in both the number of picked equities and the average appreciation. If the cut-off-point has been chosen at 0.2, the total numbers of equities being classified as "Class 1" are further reduced to 106 from 215 as compared to MLP. The average appreciation of the 106 equities has been improved to 47.156% from 38.31% as compared to MLP. Furthermore, there are sudden average appreciation dips at around cut-off-point +0.4 and +0.8, this further supports that the stock picking problem is a stochastic and non-deterministic problem that are highly challenging.

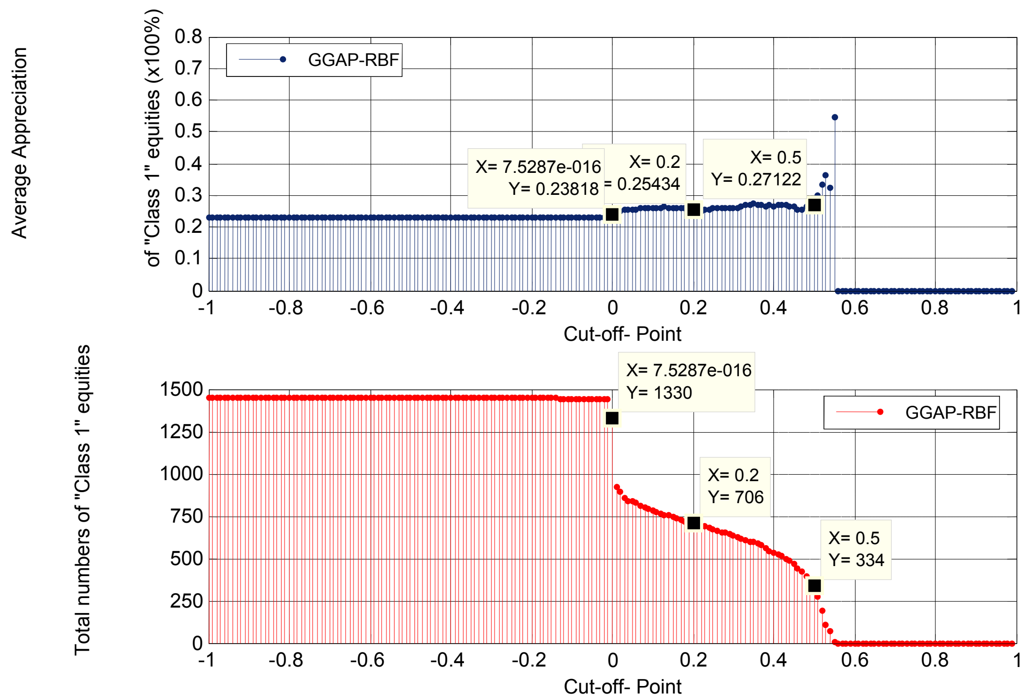

Figure 31: GGAP-RBF: Appreciation vs. Cut-off-point (validation set for the year 2004)
(Average appreciation is calculated from selected "Class 1" equities)

Figure 31 demonstrates the impact of chosen cut-off-point over the average appreciation return from the picked equities. This experiment is done on validation set with trained GGAP-RBF model. As shown as Figure 31, the cut-off-point of zero is meaningless as it signaled almost all the equities, which is 1330 out of 1448, and the average appreciation is about the average of total equities available. At cut-off-point

0.2, the total number of equities signaled has been half reduced to 706 (with average appreciation of 27.12%), which is comparable to cut-off-point zero for MLP and ANFIS. As illustrated in Figure 25, there are none or near-to-zero data that have been clustered in the range of more than +0.6. This explains that there is a dip to zero beyond value of +0.6 in the upper plot of Figure 31. This supports a future research direction of GGAP-RBF to fine-tune the clustering process for handling of similar stochastic data, which is beyond the scope of this research.

Chosen cut-of-point	Neural Network Model	Average Appreciation of the predicted "Class 1" equities	Number of equities predicted as "Class 1"
0	MLP	27.616%	782
0	ANFIS	28.198%	793
0	GGAP-RBF	23.818%	1330
+0.2	MLP	38.315%	215
+0.2	ANFIS	47.156%	106
+0.2	GGAP-RBF	25.434%	706
+0.5	GGAP-RBF	27.122%	334

Table 11: Impacts of the chosen cut-of-points over the average appreciation of predicted "Class 1" (or, "winner") equities

We summarize the findings in Figure 29, Figure 30 and Figure 31 into Table 11. Based on the findings, GGAP-RBF model is best work for higher value of cut-off-point. For example, it has the average appreciation of +27,122% with just 334 equities being picked, at the cut-of-point level of +0.5. On the other hand, both MLP and ANFIS give average appreciation of as high as 38.315% and 47.156% respectively with cut-off-point of +0.2. This concludes that we can choose higher value of cut-off-point for GGAP-RBF and lower value cut-of-points for MLP and ANFIS.

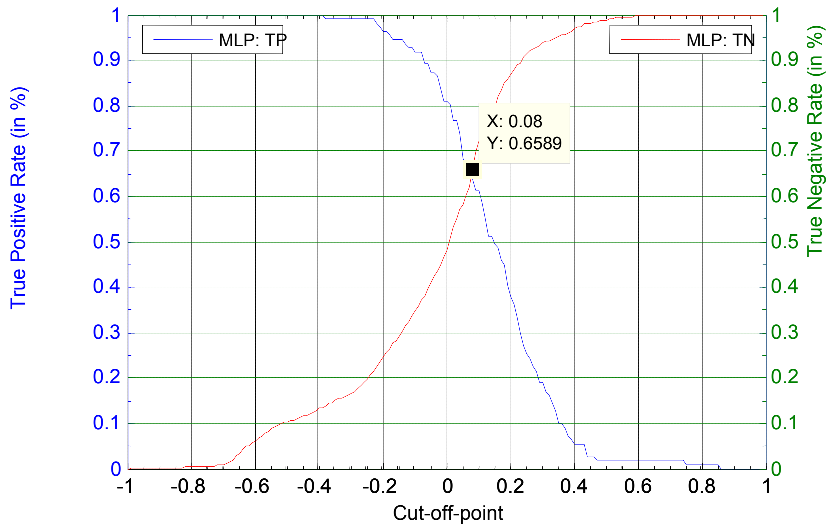

Figure 32: MLP: True Positive Rate vs. True Negative Rate

Figure 32 shows the True Positive rate and True Negative rate of the output of trained MLP on validation set. The horizontal axis shows the chosen cut-off-point for the output values of trained network to predict the data as either “Class 1” or “Class 2”. The two vertical axes show the True Positive rate and True Negative rate respectively. Thus, the cut-off-point at the intersection point of these two curves is the optimal cut off value for best configuration. The best cut off value for MLP is 0.08.

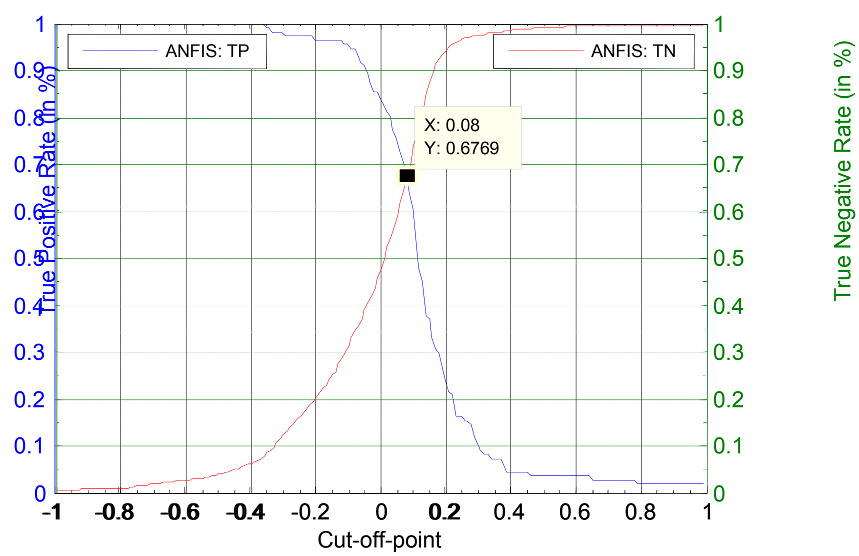

Figure 33: ANFIS: True Positive Rate vs. True Negative Rate

Figure 33 shows the True Positive rate and True Negative rate of the output of trained ANFIS on validation set. Similarly to MLP, the best cut off value for ANFIS is 0.08.

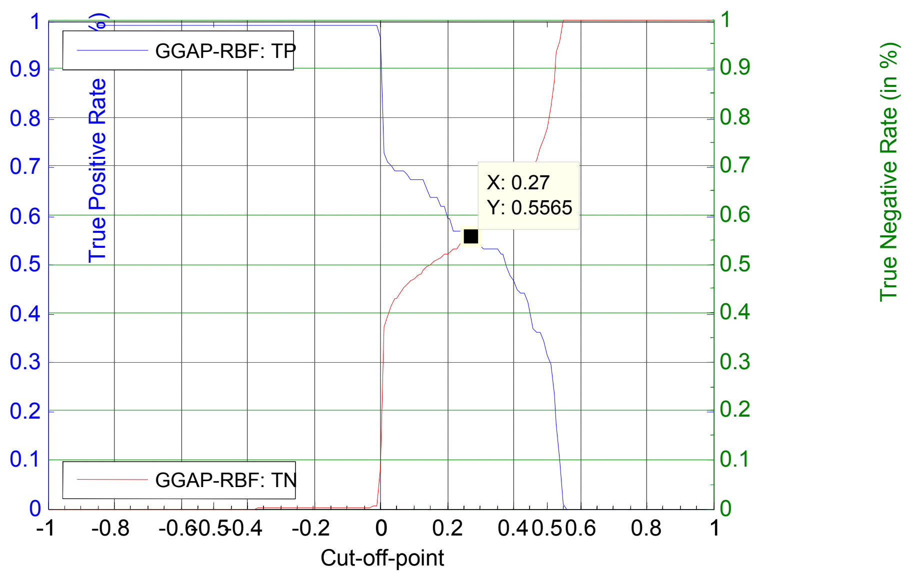

Figure 34: GGAP-RBF: True Positive Rate vs. True Negative Rate

Figure 34 shows the True Positive rate and True Negative rate of the output of trained GGAP-RBF on validation set. The best cut off value for GGAP-RBF is 0.27.

Based on the above findings, we choose the cut off values of +0.08, +0.08 and +0.27 for MLP, ANFIS and GGAP-RBF respectively. And, their numerical results for validation set are shown in Table 13, Table 14 and Table 15.

The results are presented in a customized confusion matrix table format as shown in Table 12. The summary results of each model are presented in the caption of below each table.

TP instances	FN instances	(TP+FN) instances	True Positive Rate	False Negative Rate
FP instances	TN instances	(FP+TN) instances	False Positive Rate	True Negative Rate
(TP+FP) instances	(FN+TN) instances	(TP+FP+FN+TN) instances	**Accuracy Rate***	**Precision Rate***

Table 12: Format of Customized Confusion Matrix
(*Accuracy Rate and Precision Rate is explained in *Chapter 3.3*)

71	40	111	63.964 %	36.036%
456	881	1337	34.106 %	65.894%
527	921	1448	**65.746%**	**13.472%**

Table 13: Customized Confusion Matrix for MLP (Validation set)

*(**Cut- off-point = 0.08**. Total signaled equities = **527**. **Average appreciation = 30.35%.** The average return of All 1,448 Training set equities = **22.99%)***

76	35	111	68.468%	31.532%
432	905	1337	32.311%	67.689%
508	940	1448	**67.749**	**14.961%**

Table 14: Customized Confusion Matrix for ANFIS (Validation set)

*(**Cut-off-point = 0.08**. Total signaled equities = **508**. **Average appreciation = 32.63%.** The average return of All 1,448 Training set equities = **22.99%)***

62	49	111	55.856%	44.144%
593	744	1337	44.353%	55.647%
655	793	1448	**55.663%**	**9.4656%**

Table 15: Customized Confusion Matrix for GGAP-RBF (Validation set)

*(**Cut-off-point = 0.27**. Total signaled equities = **655**. **Average appreciation = 25.75%.** The average return of All 1,448 Training set equities = **22.99%)***

Refer to Table 13, Table 14 and Table 15, all the average appreciation of the models out-perform the average appreciation of the market (22.99%). GGAP-RBF has least accuracies, the most number of equities being selected, a total of 655 equities are picked, and with the least average appreciation (25.75%).

Now, we have configured our trained soft-computing models with the best cut off values to obtain the optimal performance. The cut off values are 0.08, 0.08 and 0.27 for MLP, ANFIS and GGAP-RBF respectively. These settings will be applied on testing set to validate and compare the performance of each model. Before that, let's compare the ROC (Relative Operating Characteristics) curves of three soft-computing models.

The relative operating characteristics (ROC) curve is a highly flexible method for representing the quality of dichotomous, categorical, continuous, and probabilistic forecasts [24]. The ROC is a representation of the skill of a forecast system in which the hit rate and the false-alarm rate are compared [25]. True Positive rate and False Positive rate are explained in *Chapter 3.3*.We illustrate the relationship between True Positive rate and False Positive rate with the following Figure 35: -

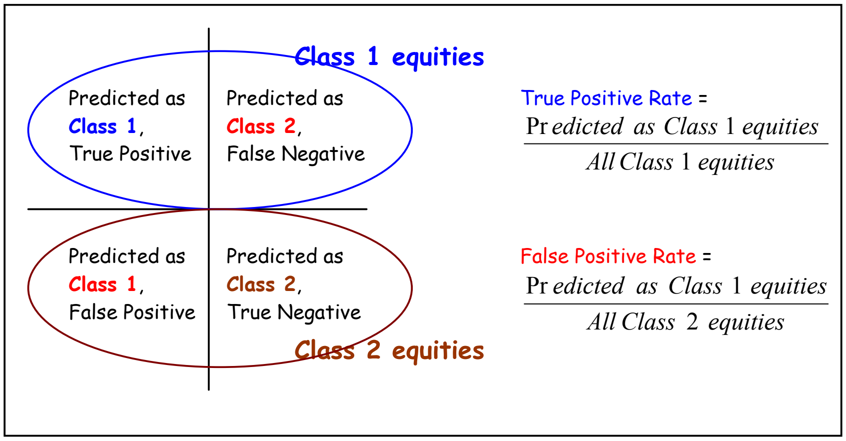

Figure 35: The relationship between True Positive rate and False Positive rate

Equities that have neural network predicted value higher than the cut off value, we classify them as Class 1 (or, "winner"), otherwise Class 2 (or, "loser"). If we apply a very high cut off value, none of the equities will be classified as Class 1 and all equities will be classified as Class 2. Refer to Figure 35, if all the equities are predicted or classified as Class 2 equities, and we predict 0 equities as Class 1 equities, then we will have both True Positive rate and False Positive rate of 0. On the other hand, if we apply a very low cut off value, result in all the equities are classified as Class 1. Then, we will have both True Positive rate and False Positive rate of 1. As we classify more and more equities as Class 1, either True Positive rate or False Positive rate, or both increase. Figure 36 illustrates the resulting True Positive rate and False Positive rate pairs when moving from high cut off values to low cut off values.

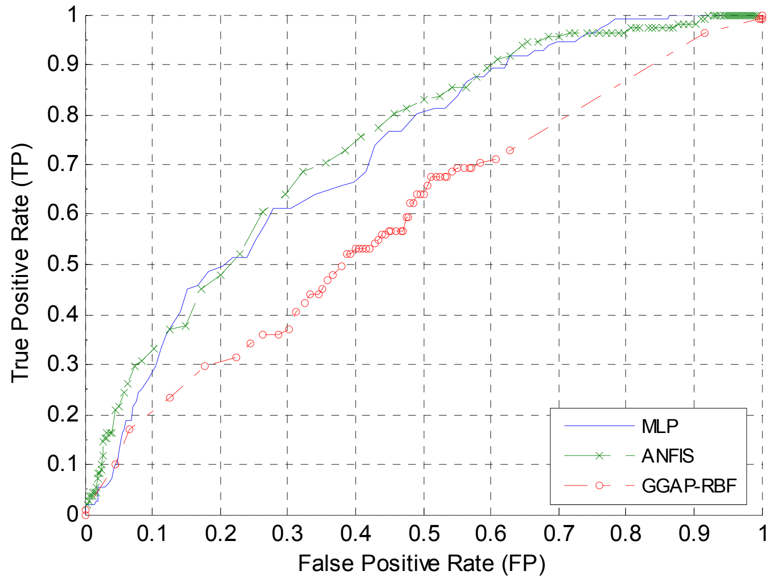

Figure 36: ROC (Relative Operating Characteristics) curves for MLP, ANFIS & GGAP-RBF

In general, for good forecast systems, the ROC curve bends toward the top left, where the True Positive rates are larger than False Positive rates. Where the curve lies close to the diagonal, the forecast system does not provide any useful information. If the curve lies below the line, the system performs negatively for the prediction problem [25].

Figure 36 is ROC curves of three soft-computing models applying on validation set. The horizontal axis of ROC curve measures error rate on “Class 2” instances while the left vertical axis measures accuracy on “Class 1” instances. The curve shows to what extent accuracy on “Class 1” instances drops with reduced error rate on “Class 2” instances. The larger the area below the ROC curve, the higher the classification potential of the algorithm. Obviously, GGAP-RBF has the lowest area below its ROC curve; MLP and ANFIS appear to be comparable.

4.3.3 Test Results

In Setting 2, we have test set of 974 instances, with known output. We will use this set of data as input of the trained soft-computing models, and apply the preconfigured cut off values for the prediction outputs to classify the instances to either “Class 1” or “Class 2”. The results are presented in a customized confusion matrix table format as shown in Table 16. The summary results of each model are presented in the caption of below each table.

TP instances	FN instances	(TP+FN) instances	True Positive Rate	False Negative Rate
FP instances	TN instances	(FP+TN) instances	False Positive Rate	True Negative Rate
(TP+FP) instances	(FN+TN) instances	(TP+FP+FN+TN) instances	**Accuracy Rate***	**Precision Rate***

Table 16: Format of Customized Confusion Matrix
(*Accuracy Rate and Precision Rate is explained in *Chapter 3.3*)

12	10	22	54.545%	45.455%
269	683	952	28.256%	71.744%
281	693	974	**71.355%**	**4.2705%**

Table 17: Customized Confusion Matrix for MLP (Test set)
*(**Cut- off-point = 0.08**. Total signaled equities = **281**. **Average appreciation = 13%.** The average return of All 974 Test set equities = **11.22%**)*

11	11	22	50%	50%
234	718	952	24.58%	75.42%
245	729	974	**74.846%**	**4.4898%**

Table 18: Customized Confusion Matrix for ANFIS (Test set)
*(**Cut-off-point = 0.08**. Total signaled equities = **245**. **Average appreciation = 14.93%.** The average return of All 974 Test set equities = **11.22%**)*

9	13	22	40.909%	59.091%
360	592	952	37.815%	62.185%
369	605	974	**61.704%**	**2.439%**

Table 19: Customized Confusion Matrix for GGAP-RBF (Test set)
*(**Cut-off-point = 0.27**. Total signaled equities = **369**. **Average appreciation = 11.15%**. The average return of All 974 Test set equities = **11.22%**)*

Refer to Table 17, Table 18 and Table 19, ANFIS model has the highest Precision Rate and average appreciation of the signaled stocks. On the other hand, GGAP-RBF has demonstrated its low ineffectiveness in picking valuable equities, with the best cut-off-point 0.27. MLP model beats the average return 11.22% by 1.78% whereas ANFIS model beats the average appreciation with 3.71%.

4.3.4 Pick the Best Ten

The above experiment assumes that we have unlimited resources. With such assumption, we can trade as many equities as possible. What if we want to focus on a certain number of equities only, say top 10 equities? We have early demonstrated that there indeed has positive correlation between the outputs of trained models and the appreciation value. Based on this finding, we select the equities based on the strength of the signals from the neural network models. The signals essentially refer to the neural network output values. Figure 37 illustrates the impacts on the average appreciation based on the number of equities that have strongest signals, and are those equities that are being picked.

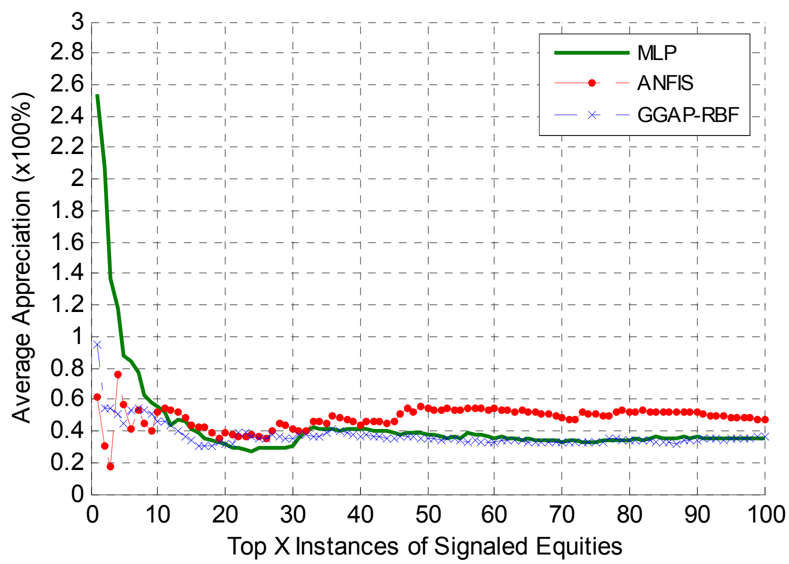

Figure 37: Average Appreciation of selected numbers of Equities with highest output values (validation set for the year 2003) (x-axis represents selected *x* numbers of equities)
(This is based on the strength of predicted output values. The average return of all 1,448 Validation set equities = **22.99%**)

Refer to Figure 37, intuitively we can choose the top 10 of the signaled equities as the average appreciation is about 40% to 60% for all three soft-computing models, which is about doubling the average market appreciation, 22.99%. We apply this on test set subsequently, which is illustrated in Figure 38.

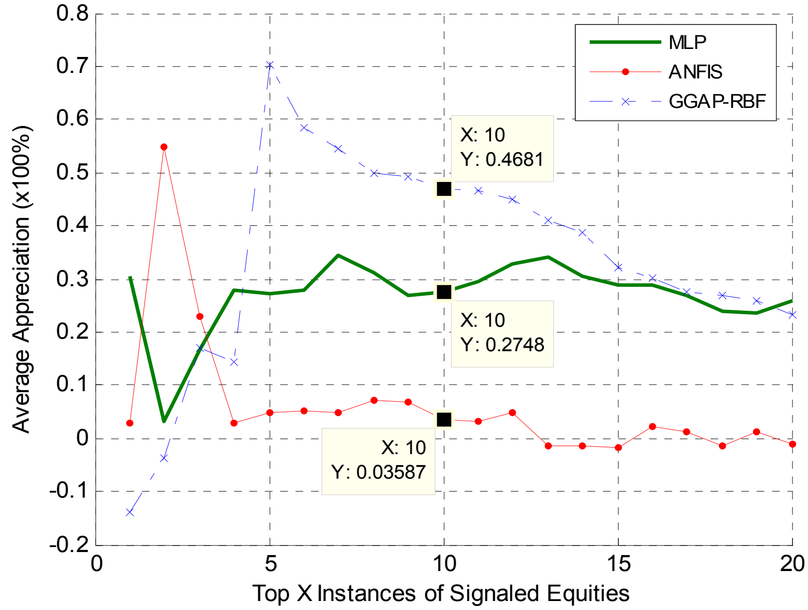

Figure 38: Average Appreciation of selected numbers of Equities with highest output values (test set for the year 2004) (x-axis represents selected *x* numbers of equities)
(This is based on the strength of predicted output values. The average return of all 974 test set equities = **11.22%**)

Let us assume that we want to trade on the top ten equities that have been signaled on trained soft-computing models, out of the 974 instances of test set. As shown in Figure 28, MLP produces 27.48% of appreciation of it top 10 equities; ANFIS produces 3.587% of appreciation and GGAP-RBF produces 46.81% of appreciation in one year period. If we combine the results of all the three models, the average appreciation of the 30 signaled equities will be 25.959%, which is much higher than the average market appreciation of 11.22% by 14.73%.

GGAP-RBF has picked several counters that are giving negative return (or, lost in appreciation). However, GGAP-RBF is capable of giving us the highest appreciation if we only focus on top ten equities. We have an equity which has next-year appreciation as high as 300% in this test set. This special counter is not picked by both MLP and ANFIS in their top tens, but only GGAP-RBF. This results in the good average appreciation of GGAP-RBF. Therefore, we could not conclude that GGAP-RBF is the best in this experiment on test set as this is an outlier case.