Trading Strategies Predicting S&P 500

Gautham Panchumarthi

1. Introduction

In the current business perspective, for a person's economic growth, investing would be considered as the primary measures. Investors need to channel through a considerable amount of budgetary information in order to unravel the business area behavior and anticipate any future movements in order to execute wise trading decisions to gain profits. Complexity of risk analysis and building a successful trading strategy inspired us to suggest another strategy and support investor's investment decisions by "Predicting the market movement of S&P 500 index". The S&P 500, or the Standard & Poor's 500, is an American stock market index based on the market capitalizations of the 500 largest companies having common stock listed on the NYSE or NASDAQ [1]. The S&P index differs from other U.S. stock market indices, such as the Dow Jones Industrial Average or the NASDAQ Composite index, because of its diverse constituency and weighting methodology. The National Bureau of Economic Research has classified common stocks as a leading indicator of business cycles and that S&P 500 is one of the most commonly followed equity indices, therefore, maybe considered as one of the best representations of the U.S. stock market, and a bellwether for the U.S. economy.

In this case, we will build a model to predict the S&P 500 index as developing three portfolios of 20 stock, 30 stock and randomly picked 20 stock models. Using these portfolios, we develop a Data Fusion approach training models and conducting prediction techniques. Specific criteria have been used to select the 20 stock and 30 stock portfolios as follows [2]:

- Market capitalization is greater than or equal to US$ 5.3 billion
- Minimum monthly trading volume of 250,000 shares
- Diversification

The analysis will be develop over a long period of time to include enough data points necessary to improve the performance of the prediction, so the time frame is from July 2001 to November 2015 which covers a span of 15 years or 747 weeks, as the data is in weekly intervals.

1.1 Correlation and Diversification

Again, in order to increase the model's performance, the stocks are not confined to a particular sector. Instead, they were picked from various sectors. Also we compare the correlations between various stocks categorized by the sectors. When a particular sector is considered, if there is an option to choose more than one stock, we go for the stocks which have least correlation amongst them. In other words, we're likely to pick stocks which have minimal relational proportional to each other.

Now, to achieve more diversification for the models, we have increased the number of stocks from 20 to 30 stocks in the second portfolio. This analysis basically gives an overall idea of whether more stocks are needed or not for the analysis to improve model's prediction power. As well, to compare the performance of the two models, we need an additional model which has no relation with the S&P 500 and random stock portfolio was selected comprising of 20 stocks.

1.2 Implementation and Data Tuning Methods

Since the data is financial was obtained directly from Yahoo Finance, without major data preprocessing, transformation, or missing values. After obtaining the raw data, some cleaning was performed in order to obtain a required format for analysis and predictions.

For the response variable, the S&P 500 index rates are considered. For each weekly value, the corresponding change is calculated based on previous week's rate. Binary classes have been developed where when the change is positive the class is denoted as "Up" and if the change is negative, then the class is denoted as "Down".

For the predictor variables, we first find the relative change of each stock compared to its previous week value. Next, if this change is observed to be positive, we denote it as 1 and if the change is observed to be negative, we denote it as 0. In this manner, we change all data for the three stock portfolios in terms of binary values 0 and 1. These binary values will act as predictor variables to reveal any future movement of the S&P 500 which will be translated as "Up" and "Down".

After cleaning the data we classified it into training and testing data, using "createDataPartition" function in R, as 80% for the training data and remaining 20% for testing. The function usage is as follows:

```
inTrainP1 <- createDataPartition(P1$Class, p = .8)[[1]]
```

The same function is used on every portfolio to divide the data in training & testing data sets where the p-value states that .8 (80%) of the data is considered as training data and the remaining .2 (20%) of the data is automatically considered as testing data.

2. Building Models

After the datasets were changed to the binary representation, we apply Support Vector Machine (SVM) and Logistic Regression (LR) to our 20, 30 and random stock portfolios. Before moving ahead, let's walk through why we have opted SVM and LR over other models.

Even though there are plenty of statistical techniques which aim in solving binary classification, SVM is a very promising non-linear and non-parametric technique than Discriminant analysis and Neural Networks. The SVMs is a cost sensitive classifier and thus works better even if the data is unbalanced with enough positive and negative examples [4, 5]. Even though we picked the 20 and 30 stocks belonging to S&P 500, there is no rule in real world that they have to behave alike with the index. Some may move with and some may move against the index, which makes sense that SVM is good choice for our scenario.

Logistic Regression (LR) has been proved to perform better for the situation where the observed outcome for dependent variable is binomial. It gives an insight into the impact of each predictor variable to the response variable. As we are predicting the weekly forecast, it is very crucial to weigh each predictor. Moreover, logistic regression does not assume linear relationship between

the independent and dependent variables and it can handle nonlinear effects as well [6, 7]. With this flexibility we have opted this LR to achieve stable results.

Revisiting previous points, we have divided each dataset of 20, 30 and random stock portfolios into training and test set with 80% for training and 20% for testing with a 10 - fold Cross Validation with 5 repetitions. We have maintained the same parameters for both SVM and LR to find out the relative best performer.

2.1 Training the SVM

Let's create our first SVM model using the "*kernlab*" R package. The *train* function is used to select values of model tuning parameters as mentioned above. Using resampling techniques, like CV and bootstrapping, a set of virtual tuned samples will be created from the training samples. For each set of virtual tuned sample combinations, there will be a corresponding set of hold-out samples. The performance of the resampling will be calculated by aggregating the results of each hold-out sample set. The possible resampling methods for *train* function are – k-fold, bootstrapping and cross-validation.

The *train* function for this SVM model have this following arguments:
x: The matrix or dataframe of predictors variables. It does not accept the categorical values or characters and it accepts only numeric values.
y: It is a numeric vector of response variable. And the model determines the type of problem whether it is regression or classification depend on the type of response variable given.
data: The argument for pointing out for which dataframe should it the model to be worked upon.
method: It the argument in which the model is specified. In our case its "*svmRadial*"
preProc: Its a pre-processing technique like centering and scaling etc., which will be used to transform the training data and can be applied to any data set with same variables.
trControl: Takes a list of control parameters for the function. The number of iterations for resampling can be set using this argument. The default number is 25, but for our case we set to 10.
tuneLength: Controls the default grid size tuning parameter. To expand the size of default list, *tuneLength* argument will be used. By giving the *tuneLength* = 5, the values of C in SVM ranging from 0.1 to 1,000 are evaluated. We have set this to 10 in our model.
...: The three dots can be used to pass additional arguments to the functions.

So after setting up all the parameters, our code looks like this

```
svmP1 <- train (Class ~ ., data = P1Train, method = "svmRadial", preProc = c("center", "scale"), tuneLength = 10, trControl = trainControl(method = "repeatedcv", repeats = 5))
```

Applying this for the first 20 stocks and checking the accuracy and Kappa static result had turned like this
svmP1

Support Vector Machines with Radial Basis Function Kernel

597 samples
21 predictor
2 classes: 'Down', 'Up'

Pre-processing: centered, scaled
Resampling: Cross-Validated (10 fold, repeated 5 times)
Summary of sample sizes: 537, 538, 538, 538, 537, 537, ...
Resampling results across tuning parameters:

C	Accuracy	Kappa	Accuracy SD	Kappa SD
0.25	0.8409435	0.6794121	0.04990852	0.10112330
0.50	0.8463051	0.6913873	0.04726193	0.09495938
1.00	0.8509718	0.7011142	0.04683021	0.09382806
2.00	0.8529774	0.7050885	0.03469040	0.06976173
4.00	0.8419040	0.6828032	0.03872497	0.07759463
8.00	0.8465650	0.6919790	0.03708640	0.07432518
16.00	0.8486158	0.6961969	0.04141286	0.08270018
32.00	0.8442712	0.6876300	0.04191917	0.08365007
64.00	0.8429322	0.6849757	0.04171957	0.08329862
128.00	0.8439379	0.6869142	0.04012879	0.08011404

Tuning parameter 'sigma' was held constant at a value of 0.02898779
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were sigma = 0.02898779 and C = 2

From the above result, the Accuracy and the Kappa static value were turned out to be - 0.8529774 and 0.7050885 respectively. It is to be noted that the Kappa static is a metric to measure the Observed Accuracy with an Expected accuracy. In other words, it compares the accuracy of the system to the accuracy of random system [9].

Similarly, on training for the remaining 30 and random stocks, overall result was observed as below:

SVM (Train)	
SP20	
Accuracy	0.8529774
Kappa Static	0.7050885
Sigma	0.02898779
SP30	
Accuracy	0.8784237
Kappa Static	0.7553153
Sigma	0.0187931
Random	
Accuracy	0.8549944
Kappa Static	0.7081624
Sigma	0.0265024

The result shows that the accuracy for 30 stocks is slightly higher than 20 and random stock portfolios, following the order of: 1. SP30 (0.8784237), 2. Random (0.8549944) & 3. SP20 (0.8529774).

2.2 Training Logistic regression

Let's create the training model same with logistic regression. So similar to the previous SVM model, we used the *train* function and given the same tuning parameters. The **method** argument in *train* function will be now set to "glm" in this case (the rest of the arguments will be all same as above). The final function will be like –

Train:
logisticRegP1 <- train(Class ~ ., data = P1Train, method = "glm", trControl =
trainControl(method = "repeatedcv", repeats = 5))

Now let's look at the train result and its accuracy –

logisticRegP1
Generalized Linear Model

597 samples
21 predictor
2 classes: 'Down', 'Up'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 5 times)

Summary of sample sizes: 537, 538, 538, 538, 537, 537, ...
Resampling results

Accuracy	Kappa	Accuracy SD	Kappa SD
0.8761017	0.7512252	0.0423763	0.08503845

The model gives the accuracy of 0.8761017 with Kappa value of 0.7512252 on the 20 stocks. Performing the same task for the 30 and random stocks yielded out the following results –

Logistic Regression (Train)	
SP20	
CV	10 fold; 5 rep
Accuracy	0.8761017
Kappa	0.7512252
SP30	
CV	10 fold; 5 rep
Accuracy	0.8914802
Kappa	0.7820704
Random	
CV	10 fold; 5 rep
Accuracy	0.876774
Kappa	0.7522542

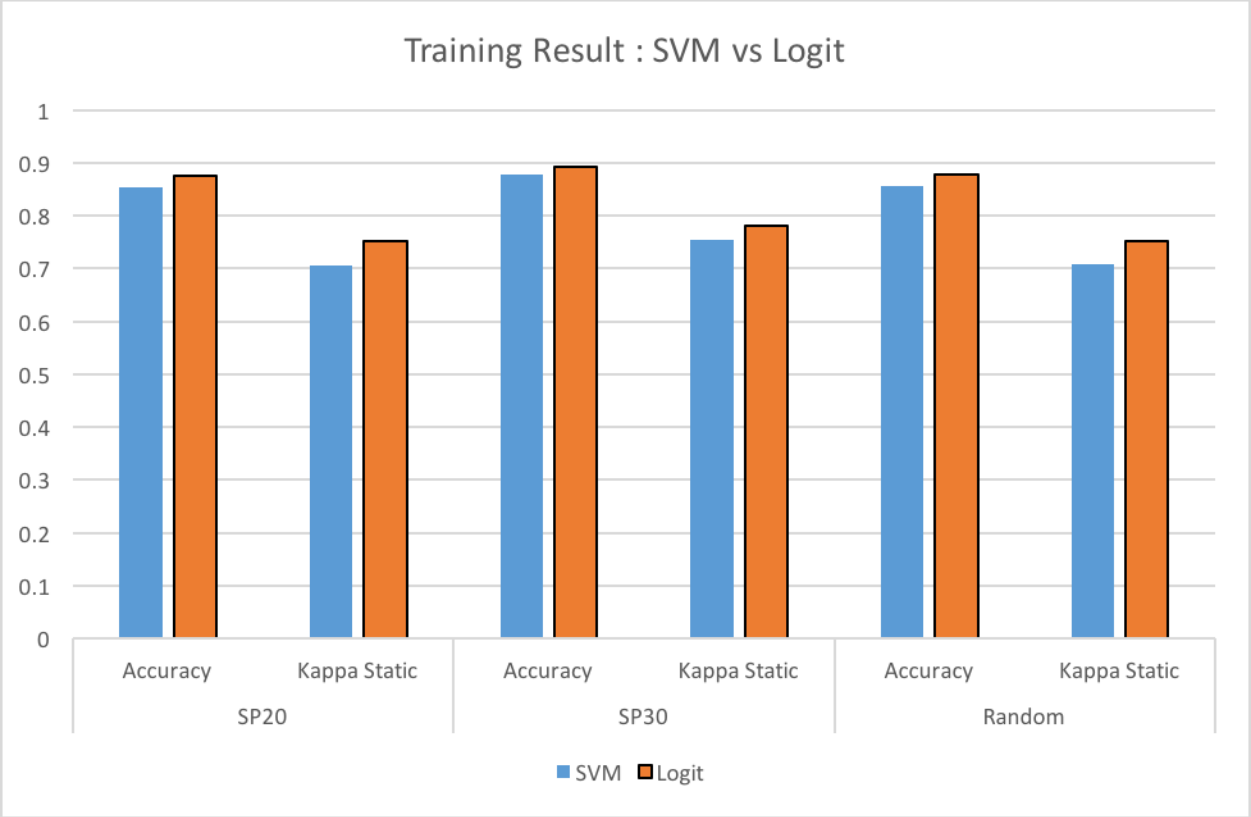
The result shows that the accuracy for 30 stocks is higher than 20 and random stock portfolios, and whereas tossing the accuracy of 30 and random stock almost to a closure result. The accuracy order is as follows: 1. SP30 (0.8914802), 2. Random (0.876774) & 3. SP20 (0.8761017).

2.3 Comparing the test results of SVM and Logistic Regression -

Now let's evaluate all the training results of both SVM and Logistic by putting as shown below –

Stock Portfolio	**Results**	**SVM**	**Logit**
SP20	Accuracy	0.8529774	0.8761017
	Kappa Static	0.7050885	0.7512252
SP30	Accuracy	0.8784237	0.8914802
	Kappa Static	0.7553153	0.7820704
Random	Accuracy	0.8549944	0.876774
	Kappa Static	0.7081624	0.7522542

Let’s plot a simple histogram for an intuitive insight over the above training result.



From the above histogram, the logistic regression performed better than the SVM for all three portfolios.

Now picking up the logistic regression and evaluate its result on the test set. Since our response variable is binomial, let’s measure its test accuracy based on sensitivity and specificity using confusion matrix.

3. Testing the the Model

Let’s test the logistic regression model based on the confusion matrix technique and walk through the Sensitivity and Specificity on the test data.

This can be calculated by using the function *confusionMatrix* from “caret” package in R. The confusion matrix focus is on the predictive capability of the model based on the predicted values instead of how fast the model takes to perform the classification. Whereas the Sensitivity is the percentage of correct predictions for the the stocks which are “Up” (true positive rate) and specificity is the percentage of correct predictions for the stocks which are “Down” (true negative rate) [8].

3.1 Test results of Logistic regression

Now let’s explore the classifier accuracy as per the confusion matrix metric -

confusionMatrix(data = TestResultsP1$pred, reference = TestResultsP1$obs)

```
Confusion Matrix and Statistics
          Reference
Prediction Down Up
      Down   75  9
      Up      4 61

               Accuracy : 0.9128
                 95% CI : (0.8554, 0.9527)
    No Information Rate : 0.5302
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8241
 Mcnemar's Test P-Value : 0.2673

            Sensitivity : 0.9494
            Specificity : 0.8714
         Pos Pred Value : 0.8929
         Neg Pred Value : 0.9385
             Prevalence : 0.5302
         Detection Rate : 0.5034
   Detection Prevalence : 0.5638
      Balanced Accuracy : 0.9104

       'Positive' Class: Down
```

The result implies the sensitivity of 0.9494 and specificity of 0.8714 for the 20 stocks portfolio.

Similarly, let's summarize the all results of 20, 30 and random stocks from the below table along with training result for an overall benchmark comparison –

Logistic Regression			
SP20			
Train		Test	
CV	10 fold; 5 rep	CI	(0.8554, 0.9527)
Accuracy	0.8761017	Accuracy	0.9128
Kappa	0.7512252	Kappa	0.8241
		Sensitivity	0.9494
		Specificity	0.8714
SP30			
CV	10 fold; 5 rep	CI	(0.8554, 0.9527)
Accuracy	0.8914802	Accuracy	0.9128
Kappa	0.7820704	Kappa	0.8244
		Sensitivity	0.9367
		Specificity	0.8857
Random			
CV	10 fold; 5 rep	CI	(0.7927, 0.9106)
Accuracy	0.876774	Accuracy	0.8591
Kappa	0.7522542	Kappa	0.7164
		Sensitivity	0.8861
		Specificity	0.8286

The test results surprisingly showed the same accuracy of 0.9128 for the 20 and 30 stocks including the value for confidence intervals (0.8554, 0.9527) but the Kappa static with slight variation (0.8241 for 20 stocks and 0.8244 for 30 stocks). On the other hand, the accuracy of Random stocks is 0.8591 with Kappa static of 0.7164 which is less than than the accuracy of other portfolio models.

Interestingly the Sensitivity is higher for 20 stocks showing the result of 0.9494 and whereas the Specificity is higher for the 30 stocks of 0.9367.

The relationship between the sensitivity and specificity with the performance for the classifier can be visualized and studied using the ROC curve in the coming sections.

3.2 Probability Histogram Plot

Moving forward, as the resulting model is applied to predict the S&P direction for the next trading cycle, we will plot the probability of truly identifying S&P 500 moving Down and Up. In particular, since any trading strategy is exposed to a high risk of losing money when S&P 500 is going down, we will be interested in correctly predicting downward movement as opposed to upward. First, let's look into the probability histogram plot for the 20 stocks portfolio (See Figure.1). As seen, the model performs relatively well identifying S&P 500 moving downward vs upward with above 40 times correctly Down and just a little bit over 30 times Up. If we look into the histogram chart for the 30 stock portfolio (See Figure.2), we observe similar results in terms of correctly identifying downward movement as opposed to upward but it is clear that the 30 stock portfolio achieves better results correctly identifying Down with almost 50 times and just a little below 40 times Up then the 20 stock portfolio. Surprisingly, the random-stock portfolio, although underperforming, also achieves relatively good results as well correctly identifying downward movement at about 35 times and upward at about 31 times (See Figure. 3). Considering the fact that stocks are randomly picked further investigation is required to discover if there's any underlying relation or this is due to some one-time situation which is outside the scope of this project.

3.3 Receiver Operating Characteristic Curve.

Receiver Operating Characteristic (ROC) curve can be used to compare multiple models thru measuring the area under the curve (AUC). Optimal model would have the highest number of area under curve and move toward the upper left corner of the plot with AUC close to 1. ROC curve is a general method to determine an effective threshold for event detection, as discussed earlier our interest event detection is downward movement of the S&P 500 to prevent losses. Each point on the curve represents an "operating point" responding to a resultant threshold, and a specificity and sensitivity trade-off. As seen in Figure. 4, ROC curve of 20 stocks and 30 stocks portfolios cover most area under the curve as both curves move towards the upper left corner of the plot. Random stock portfolio underperforms although still displaying good results in terms of area under the curve.
It is very difficult to distinguish just by looking at the charts which one would be the optimal model, therefore, we need to look what is the actual percentage of the AUC and compare, using the following formula[3]:

$$A_{ROC} = \int_0^1 \frac{TP}{P} \quad d\frac{FP}{N} = \frac{1}{P\,N} \int_0^N TP \;\; dFP$$

```
> auc(P1ROC)
Area under the curve: 0.9714
> ci.auc(P1ROC)
95% CI: 0.9491-0.9937 (DeLong)
>
> auc(P2ROC)
Area under the curve: 0.9696
> ci.auc(P2ROC)
95% CI: 0.9457-0.9935 (DeLong)
>
>
> auc(RPROC)
Area under the curve: 0.9383
> ci.auc(RPROC)
95% CI: 0.9031-0.9736 (DeLong)
```

As seen above, the 20 stock portfolio covers the most AUC slightly outperforming the 30 stock and random stock portfolio with 0.0018 and 0.0331. Would that be considered the optimal model? Recalling a previous discussion that the random stock portfolio would require a much deeper investigation to support these results, we would exclude it as a possible optimal model at this point. Also, recalling Figure. 2, where we saw that 30 stock portfolio performs much better in terms of correctly identifying downward movement of the S&P 500, we conclude that this is a very small difference which will not be taken into an account. Indeed, the 30 stock portfolio would be considered the optimal model in this particular case.

3.4 Lift Curves

Another visualization tool for assessing the classification ability of a model, similar to the ROC curve, is the Lift Curve (LC). It predicts the testing samples and rank them by scores as probability of interest and plots the cumulated “lift” against the cumulated percentage of samples. By varying the threshold of the probabilistic classifier we get a set of points. The curve we get by drawing a convex hull of the given (binary) points is called a lift chart. Just like in the ROC curve, it holds true that each point on the convex hull corresponds to a combined classifier and the probabilistic classifier that corresponds to the convex hull is always at least as good as the original classifier from which the hull was derived.

In this case, we will observe the 30 stock portfolio performance as we already identified it as the optimal model. Nevertheless, we would like to see how close it is to the perfect model. For that purpose, we will use the Lift curve that represents the best possible model for this particular data set, covering the most area under the curve. As seen from Figure. 5, the 30 stock portfolio model is very close to the perfect model. Plotted is a formula as an input where the true class is on the left-hand side of the formula, and one or more columns for model class probabilities are on the right.

Conclusion

In conclusion, we developed a data fusion approach using Logistic Regression and Support Vector Machine models to predict market movement of the S&P 500 and support investing decisions. The models described in this report have displayed highly optimistic results.

Considering that the 20 and 30 stock portfolios were carefully chosen based on earlier specified selection criteria, such as market capitalization is greater than or equal to US$5.3 billion and minimum monthly trading volume of 250,000 shares, allows for good correlation with S&P 500 (as it is well known that it consists of current 500 best stocks in the market) as intended. Surprisingly, the random stock portfolio, although underperforming, displayed almost similar results which raises some questions of the validity and accuracy of the optimal model. Therefore, further analysis and computation would be required to support the results and reveal more patterns from the data itself.

References:

1. *"Standard & Poor's 500 Index - S&P 500". Investopedia. Retrieved 11 June 2012*
2. Renshaw, Edward. *The Stock Market, Oil Price Shocks, Economic Recessions and the Business Cycle With An Emphasis on Forecasting*, December 2002
3. *Metodoloski, Zvezki. ROC Curve, Lift Chart and Calibration Plot. Vol. 3, No. 1, 2006, 89-10*
4. *Auria, L. and Moro, R. (2008). Support Vector Machines (SVM) as a Technique for Solvency Analysis. Berlin: Deutsches Institut für Wirtschaftsforschung.*
5. *"Pros of SVM Classifier." - ResearchGate. N.p., n.d. Web. 07 Dec. 2015.*
6. *"Advantages and Disadvantages of Logistic Regression." Victor Fangs Computing Space. N.p., 10 May 2011. Web. 07 Dec. 2015.*
7. *"What Are the Advantages of Logistic Regression over Decision Trees?" - Quora. N.p., n.d. Web. 07 Dec. 2015.*
8. *"Artificial Intelligence in Motion." Tools for Machine Learning Performance Evaluation: Confusion Matrix -. N.p., n.d. Web. 07 Dec. 2015.*
9. *Arora, A. (2011). Confusion Matrix – Another Single Value Metric – Kappa Statistic | Software Journal. [online] Standardwisdom.com. Available at: http://standardwisdom.com/softwarejournal/2011/12/confusion-matrix-another-single-value-metric-kappa-statistic/..*

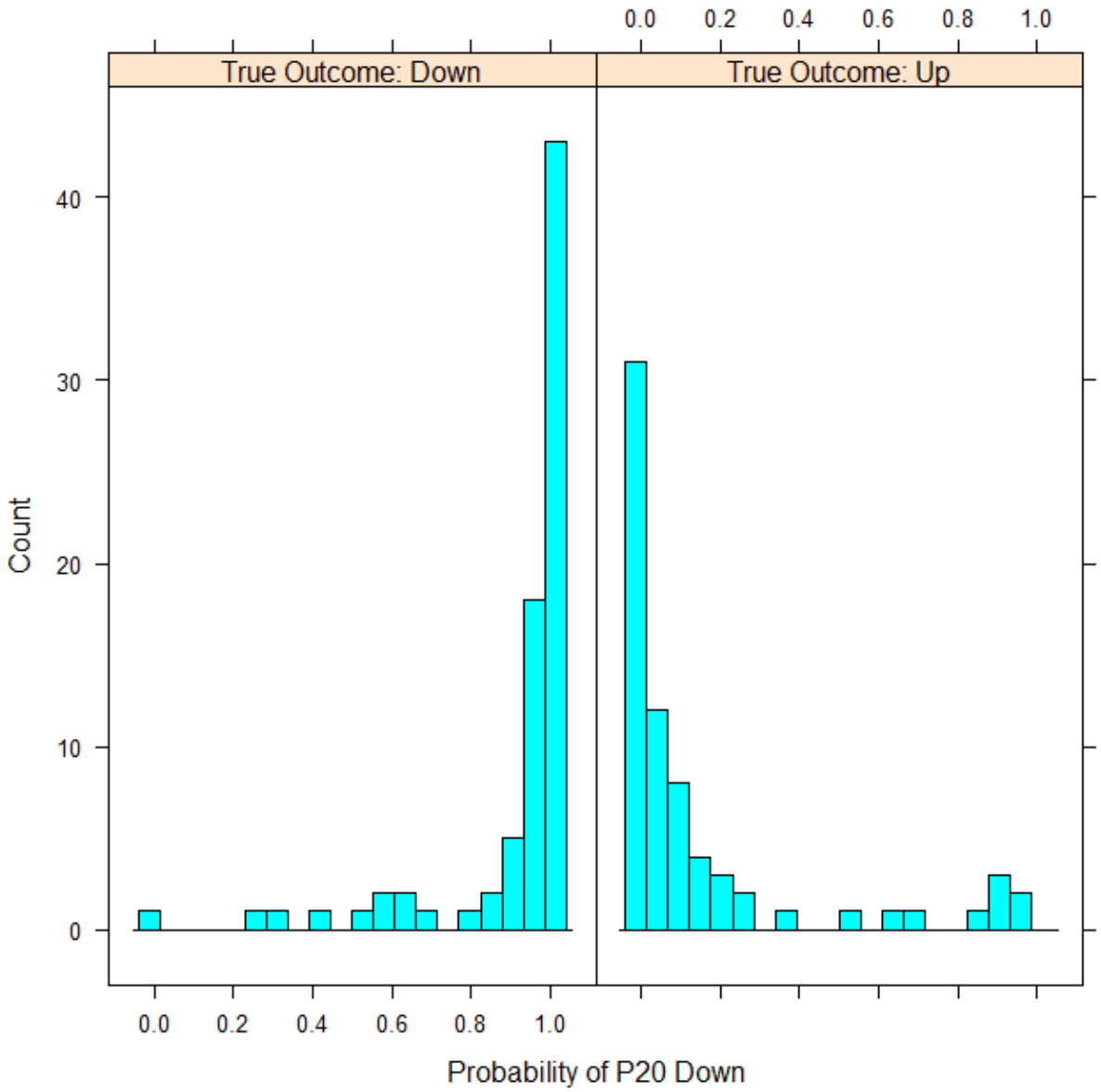

True Outcome: Down
True Outcome: Up
0.0
0.2
0.4
0.6
0.8
1.0
Count
0
10
20
30
40
Probability of P20 Down

Figure. 1

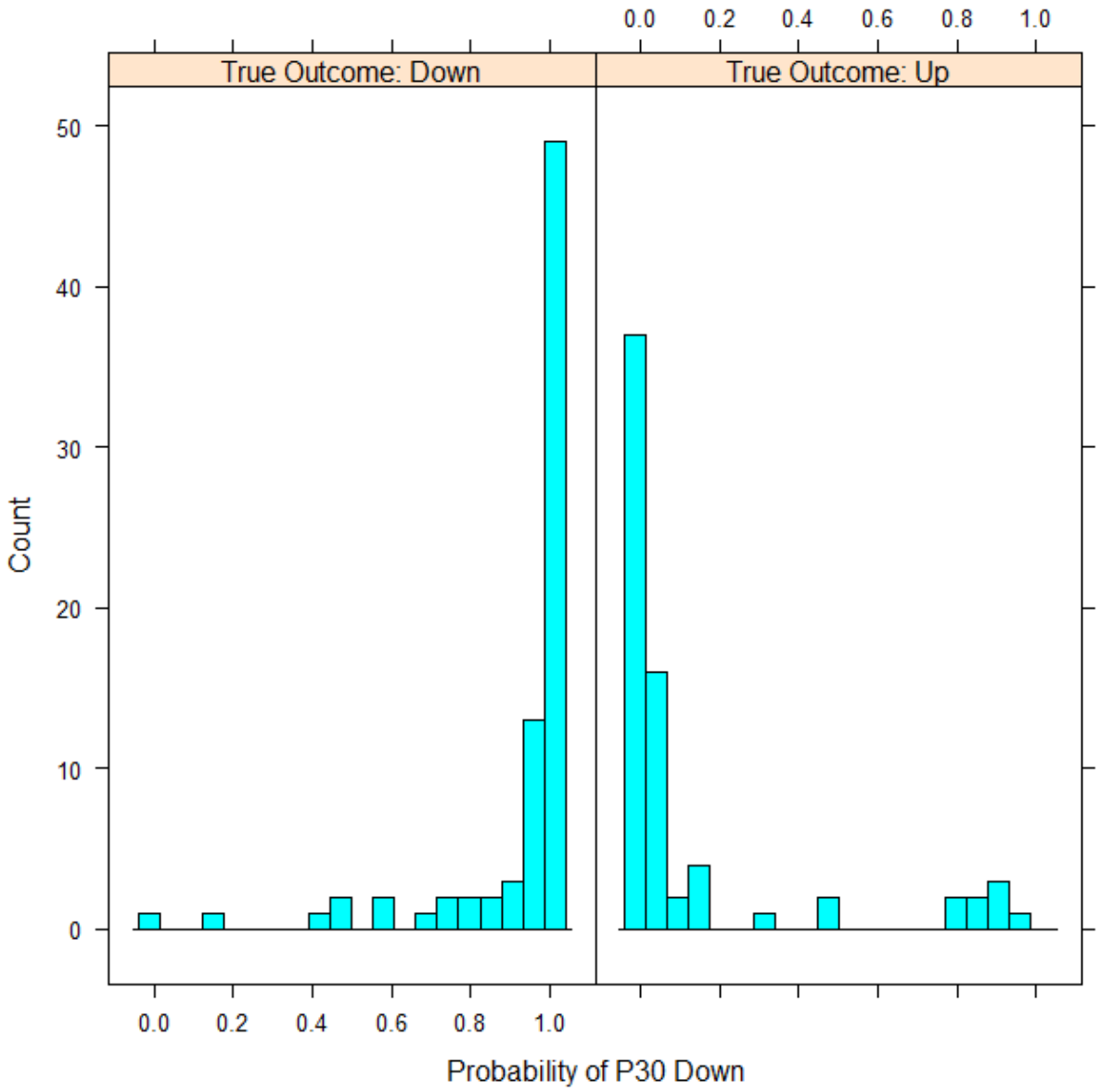

True Outcome: Down
True Outcome: Up
0.0
0.2
0.4
0.6
0.8
1.0
Count
0
10
20
30
40
50
Probability of P30 Down

Figure. 2

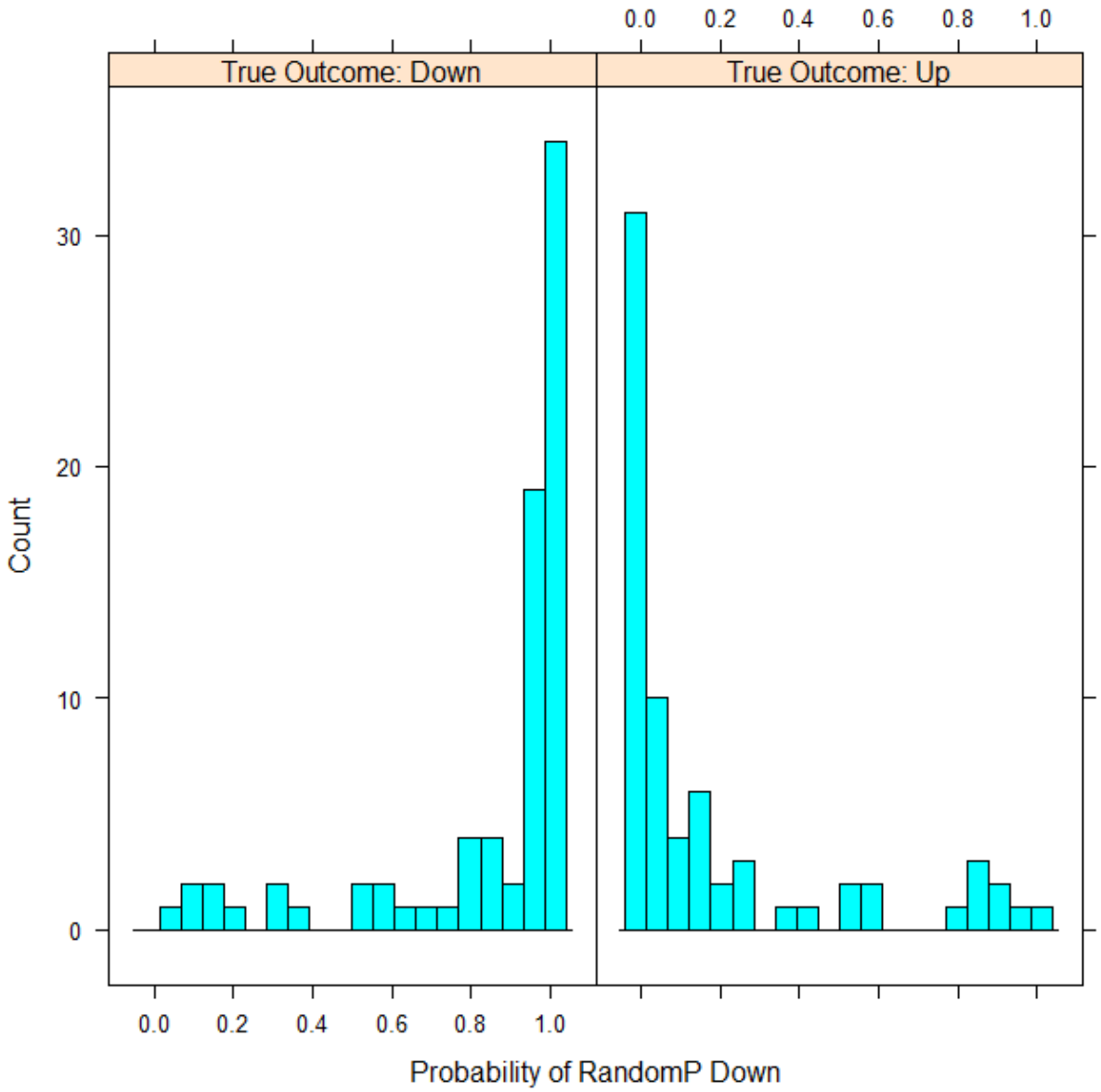
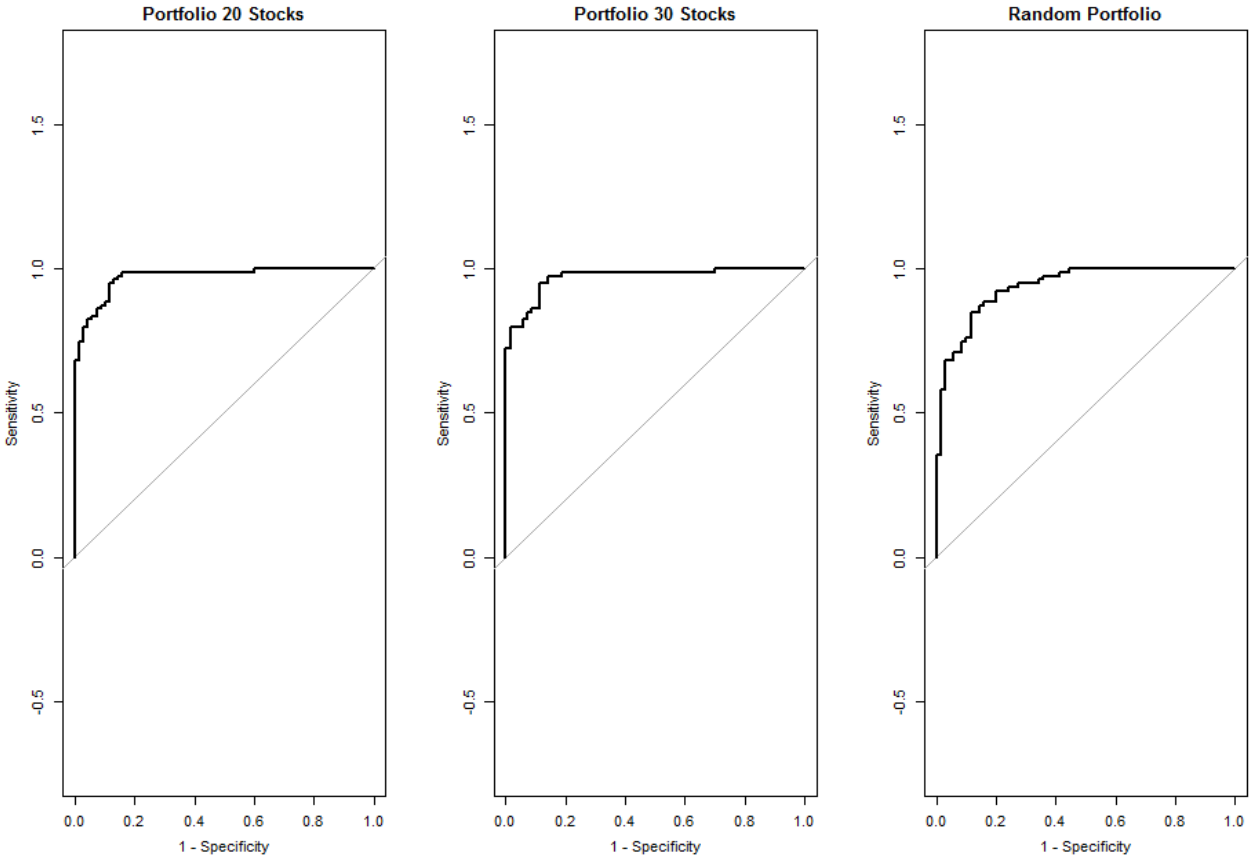

True Outcome: Down
True Outcome: Up
Count
0
10
20
30
0.0
0.2
0.4
0.6
0.8
1.0
Probability of RandomP Down

Figure. 3

Figure. 4


Portfolio 20 Stocks
Portfolio 30 Stocks
Random Portfolio
Sensitivity
1 - Specificity
-0.5
0.0
0.5
1.0
1.5
0.0
0.2
0.4
0.6
0.8
1.0

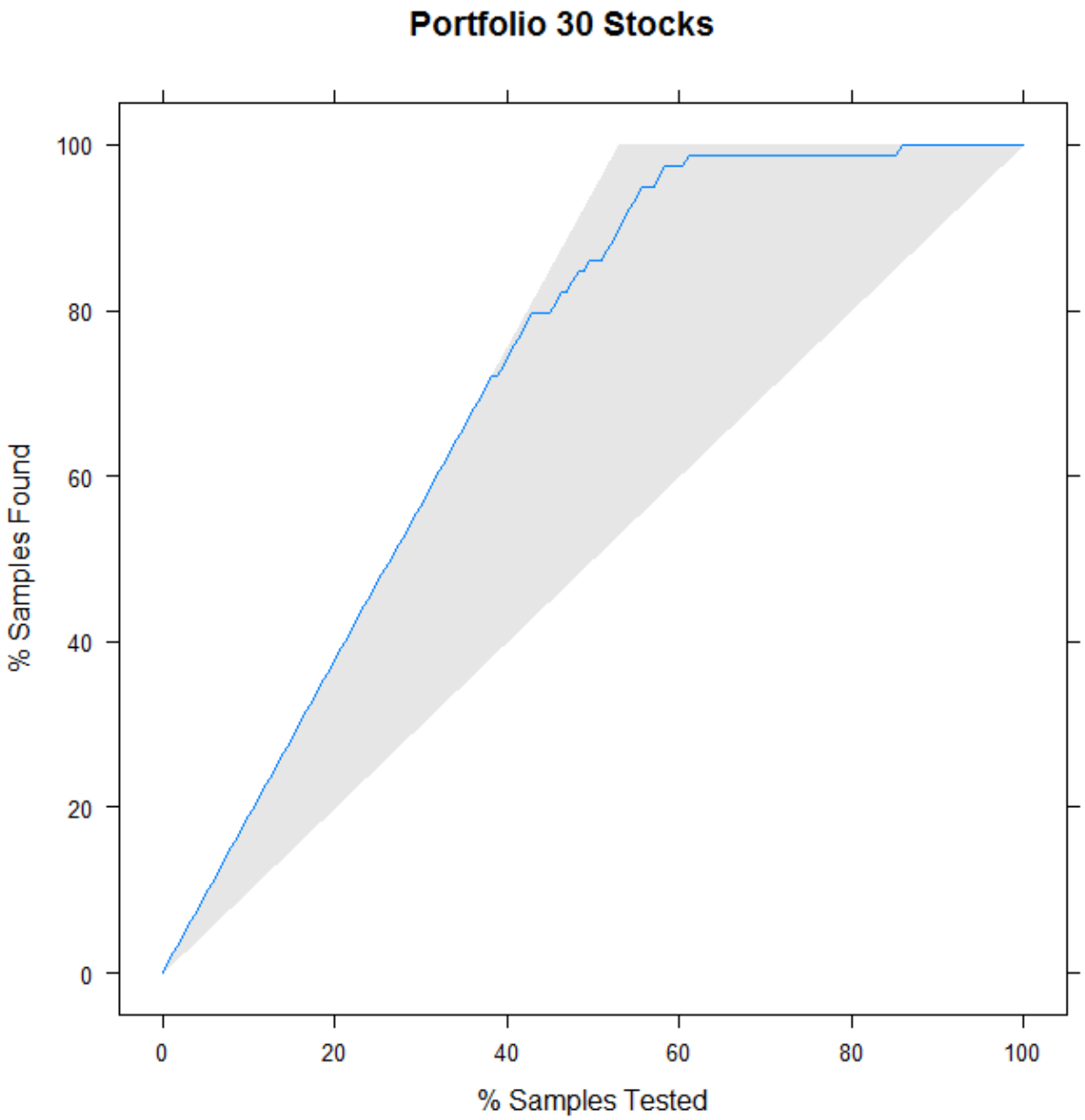

Portfolio 30 Stocks
% Samples Found
% Samples Tested
0
20
40
60
80
100

Figure. 5

Portfolios Stock List:

20 Stock Portfolio:

ACN'- Accenture plc - Information Technology
MMM - 3M Company - Industrials
ABT – Abbott Laboratories - Health Care
ACE - ACE Limited - Financials
ATVI - Activision Blizzard Information Technology
ADBE - Adobe Systems Inc - Information Technology
AES - AES Corp - Utilities
AET - Aetna Inc – Health Care
AFL - AFLAC Inc - Financials
AMG – Affiliated Managers Group Inc - Financials
A – Agilent Technologies Inc – Health Care
GAS - AGL Resources Inc – Utilities
APD - Air Products & Chemicals Inc – Materials
ARG - Airgas, Inc. – Materials
AKAM – Akamai Technologies Inc - Information Technology
AA - Alcoa Inc – Materials
AGN - Allergan plc – Health Care
ALXN – Alexion Pharmaceuticals – Health Care
ADS - Alliance Data Systems – Information Technology
ALL - Allstate Corp – Financials

30 Stock Portfolio:

ACN'- Accenture plc - Information Technology
MMM - 3M Company - Industrials
ABT – Abbott Laboratories - Health Care
ACE - ACE Limited - Financials
ATVI - Activision Blizzard Information Technology
ADBE - Adobe Systems Inc - Information Technology
AES - AES Corp - Utilities
AET - Aetna Inc – Health Care
AFL - AFLAC Inc - Financials
AMG – Affiliated Managers Group Inc - Financials
A – Agilent Technologies Inc – Health Care
GAS - AGL Resources Inc – Utilities
APD - Air Products & Chemicals Inc – Materials
ARG - Airgas, Inc. – Materials
AKAM – Akamai Technologies Inc - Information Technology
AA - Alcoa Inc – Materials
AGN - Allergan plc – Health Care
ADS - Alliance Data Systems – Information Technology
ALL - Allstate Corp – Financials
ALTR - Altera Corp – Information Technology
MO - Altria Group Inc - Consumer Staples

AMZN - Amazon.com Inc – Consumer Discretionary
AEE - Ameren Corp – Utilities
AEP - American Electric Power – Utilities
AXP - American Express Co – Financials
AIG – American International Group, Inc. – Financials
AMT - American Tower Corp A – Financials
ABC – AmerisourceBergen Corp - Health Care
AME - Ametek – Industrials
AMGN - Amgen Inc - Health Care

Random 20 stock Portfolio:

REDF - Rediff.com India Limited
SIFY - Sify Technologies Limited
SINA - SINA Corporation
EEP - Enbridge Energy Partners, L.P**.**
EMF - Templeton Emerging Markets Fund
EPR - EPR Properties
EQR - Equity Residential
LZB - La-Z-Boy Incorporated
AZZ - Azz Inc **-** Consumer Durables
AZO – Autozone - Consumer Services
AXR - Amrep Corp – Finance
AXE - Anixter International Inc - Consumer Non-Durables
AWR - American States Water Company - Public Utilities
AVY - Avery Dennison Corp - Consumer Durables
AVX - Avx Corp - Capital Goods
AVD - American Vanguard Corp - Basic Industries
ASR - Grupo Aeroportuario Del Sureste – Transportation
ASG - Liberty All-Star Growth Fund – Finance
ASA - ASA Gold and Precious Metals – Finance
ARL - American Realty Investors - Finance