Definition (approx. 1-2 pages)

Project Overview

Historically hedge funds have been using market knowledge and intuition to gather signals that are helpful in the prediction of the returns of stocks. Lately an onslaught of evidence has been gathering that machine learning could be a valuable tool in profitable analysis. This brings rise of the machine learning quantitative analyst and this project takes a look into a hypothetical problem within the profession. It is important to note, that this dataset is given by a real quantitative analysis firm distributed by the data scientist recruitment group Correlation One. Please open MachineLearningTest.pdf for further details. I turn their assignment into a capstone project with a scenario described below.

In this scenario, I am a machine learning engineer working as a quantitative analyst for a hedge fund. Furthermore this hedge fund trades in the S&P 500 index but can use data from other indexes for analysis in these trades. The company provides the machine learning engineer data to use for profitable analysis. More specifically the data contains 10 columns, each one representing a open close prices for a stock ordered according to dates. Each column name is named S1 through S10. More specifically, S1 trades in the united states as part of the S&P 500 Index while stocks S2, S3 through S10 trade in Japan in the Nikkei Index. S1 is missing the latter portion of the dataset in comparison with stocks S2 through S10. More specifically, S1 contains 50 data points, while S2 through S10 contains 100 data points each. Each of these data points correspond to a date.

The goal for the company in providing the machine learning engineer this dataset is for the engineer to build a model that forecasts the latter portion of S1,

as a function of S2 through S10. The model should predict the latter 50 rows for S1. In addition, fund researchers point out not all of the columns S2 through S10 are important in predicting S1.

Problem Statement

In summary the training set is columns S2 through S10. The testing set is column S1 which has 50 data points corresponding to the initial 50 dates. This is in comparison with the training set which has a 100 data points for each stock or feature (S2 through S10). Cross validation will be implemented on the dataset to make sure the testing set is not a statistical anomaly that is generally not representative of the real environment. Furthermore this will assure that we have a more averaged training error that has more potential to represent actual environment testing. For each cross validated training, the dataset will go through dimensionality reduction using principal component analysis. After we have a new dataset which was transformed by principal component analysis, I will apply an ensemble technique. An ensemble is a collection of various machine learning algorithms each of which could be weighted and summed into a final classification or regression solution. The ensemble will contain linear regression or general linear model, logistical regression, and k nearest neighbors. This ensemble will return the model which should predict the actual stock open to close changes for the latter portion of S1. This is the solution to the problem: Predict the 50 latter data points of the stock S1.

Metrics

To check for accuracy or testing error I will be using mean squared error. Basically this metric takes the difference from actual values and predicted values,

squares it, and then returns a sum of all these values for each training point. The squaring is done so that negative differences don't cancel positive differences during the summing process. The reason I'm fond of this particular metric is because mean squared error is the second moment of the error. Thus it includes both variance of the estimator and its bias.

Analysis (approx. 2-4 pages)

Data Exploration

A data set is present for this problem. The features are daily stock open close prices. For example, if a stock opens at 5 dollars and then closes at 9 dollars, the open close price would be 4 dollars. Basically we have 9 columns each representing a stock open close price, for the training set. S1 is a stock open close price that represents both the testing set and the value we are building a model to predict. More specifically the first 50 dates for S1 are known and will be used to validate the training set. While the latter portion of S1 is unknown and must be predicted. The testing column is S1 (a s&p stock) and the training set contains 'S2' through 'S10' each of which is an open close price for a stock in the Nikkei index. For the first fifty points, we have the values for S1, and this subset of the dataset will be used to validate the training. The latter fifty data points, S1 is missing its values. Furthermore I have provided a few rows below, of the training data so that the reader may understand better on what we're working with.

	S2	S3	S4	S5	S6	S7 \
newDates						
2014-05-30	-0.828505	1.268874	1.468834	1.207883	0.622798	1.821350
2014-06-02	-0.245839	-0.558069	-0.990142	-0.497502	-0.137380	0.158878
2014-06-03	-0.745564	0.254992	0.322715	0.832613	0.091484	-0.405513
2014-06-04	-0.636639	-0.840802	0.643013	0.167639	0.017852	0.293241
2014-06-05	-0.710122	1.396653	0.135177	1.888105	0.583958	0.949694

```
                 S8        S9       S10
newDates
2014-05-30 -0.699674  1.890508  0.523177
2014-06-02  0.081083 -1.311177  0.175995
2014-06-03 -0.323873  0.920877 -0.066132
2014-06-04  0.871516  0.201038 -0.166004
2014-06-05  0.428746  1.050444  1.196328
```

The gaps in the above data points are because of omission of holidays and weekends. Using the pandas dataframe, I was able to call the describe function for the training set and testing set which is shown below.

Training Set Description

	S2	S3	S4	S5	S6	S7	S8	S9	S10
count	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000	50.000000
mean	-0.250501	0.351924	0.296563	0.453432	0.160605	0.278186	-0.036940	0.493267	0.009384
std	1.062886	1.879886	1.137385	2.801194	0.902715	2.013317	0.641218	1.947678	0.645085
min	-3.248847	-3.859708	-1.457443	-4.658581	-1.416375	-3.618538	-1.650624	-2.892026	-1.350912
25%	-0.807769	-0.740356	-0.577430	-1.797284	-0.588935	-1.147446	-0.491008	-1.055960	-0.315425
50%	-0.214763	-0.072577	0.205831	0.527903	0.075039	0.047589	-0.053798	0.600966	0.033475
75%	0.382425	1.325070	0.933386	2.316513	0.618160	1.371763	0.429557	1.632827	0.427829
max	1.953226	5.201204	3.320352	7.275904	2.503466	5.576043	1.272966	4.988727	1.326313

The mean shows the average for each stock in terms of open to close changes per day. All of these are positive except for S2 and S8. S5 seems to be the most profitable stock with an average open to close change of .35 and a maximum daily change of 9.69. In particular S1 has a positive mean with value 0.118352. This mean is most similar to S6. Furthermore it's standard deviation was 0.930020 which is most similar to S6. Simply reading the data,

Testing set description

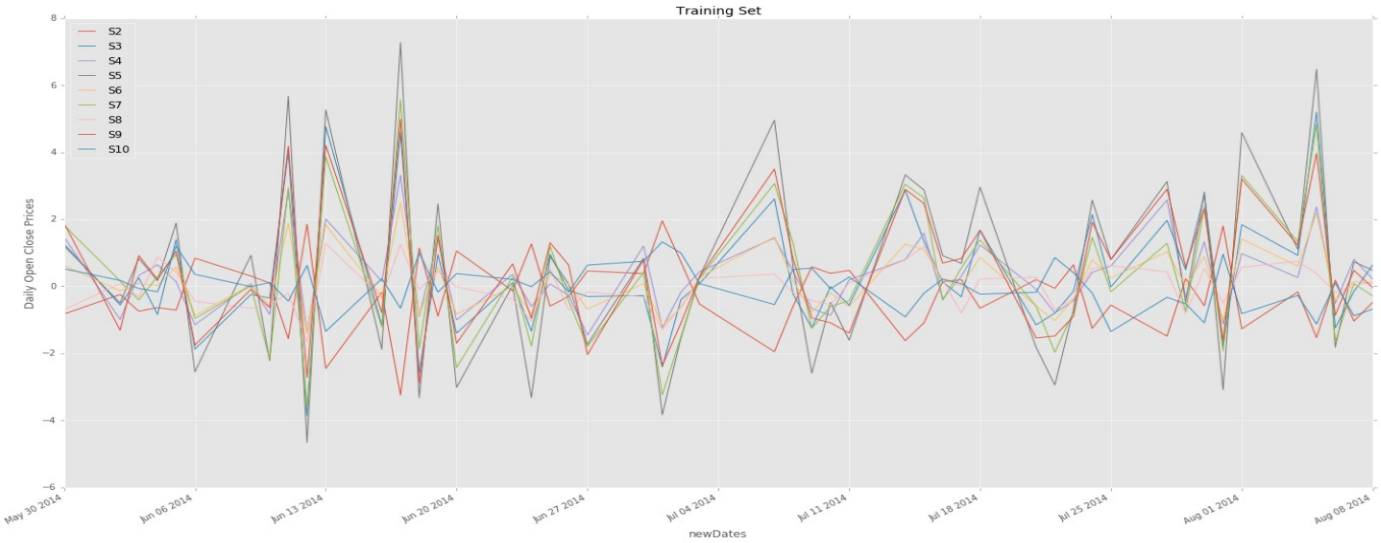
	S1
count	50.000000
mean	0.118352
std	0.930020
min	-1.537622
25%	-0.627997
50%	0.090518
75%	0.581462
max	2.435610

a reader realizes S5 is the most different. It is sporadic in terms of the other stock price changes with a standard deviation of 3.198, while still being the most profitable.

Exploratory Visualization

The S1 data is plotted below

Next the training set is plotted below:



Comparing both graphs, it is important to note, that the y axis is scaled differently. Also some stocks shown in the second plot are more sporadic while others are less, in comparison with S1. Although they have different open close prices everyday, it is easy to see they all center around the x axis or 0. These graphs show the training data on one end and then the resulting testing space. It gives the data scientist a birds eye view of the data, in order to create models. By analyzing new predictions in comparison with the input space, one might be able to understand right away that the model did something right or wrong visually.

Algorithms and Techniques

The algorithms I will test in an ensemble are linear regression or general linear model, logistical regression, and k nearest neighbors. The dataset will first be transformed principal component analysis for each cross validated training and testing sets. Principal component analysis is a powerful mathematical tool which reduces dimensionality, so that one knows what new features, from the transformed dataset, brings the most variation. Furthermore, it rearranges the data on new axises, which represents the new features. For example if one of the stock brings little variation to the overall stock price, it's variation could be dropped in a mathematically efficient manner saving both time and energy for the computation while conserving an effective dataset to build the model. The ensemble will be applied with the listed machine learning algorithms using exhaustive parameter tuning with the help of gridsearchcv, an sklearn function. To start I apply the ensemble to 4 of the most valuable pca components. Next pca_inverse transform is used to bring the model dataset back to the original

space. Similarly the general linear model, logistical regression and reinforcement learning will be applied. General linear model or linear regression is a model where each feature is weighed differently and therefore contributes differently to the final classification. The formula is y = x1w1+x2w2+...xnwn+b. Logistic regression is similar but fits a logistic polynomial curve instead of a linear one. Finally the last part of the ensemble is the k nearest neighbors . For a new or training vector, the k nearest neighbors algorithm finds the k nearest vectors relative to the training or new vector. It then picks the mode classification of the set of the found k nearest vectors. These algorithms will be combined into an ensemble, specifically the voting classifier where each algorithm has a vote to decide the final regression.

Benchmark

Unfortunately there's no benchmark to compare the model to an actual result. This capstone project is given by a quantitative analyst firm looking for a machine learning engineer. Although the date for the submission (to the firm) has passed, I find this project valuable since a real firm uses this to seek a valuable skill set.

Methodology (approx. 3-5 pages)

Data Processing

In this section, all of your preprocessing steps will need to be clearly documented, if any were necessary. From the previous section, any of the abnormalities or characteristics that you identified about the dataset will be addressed and corrected here. Questions to ask yourself when writing this section:

•If the algorithms chosen require preprocessing steps like feature selection or feature transformations, have they been properly documented?

•Based on the **Data Exploration** section, if there were abnormalities or characteristics that needed to be addressed, have they been properly corrected?

•If no preprocessing is needed, has it been made clear why?

Implementation

Refinement

Results (approx. 2-3 pages)

Model Evaluation and Validation

Justification

Conclusion (approx. 1-2 pages)

Free-form Visualization

Reflection

Improvement

Work Cited:

1. Mean squared error *http://www.vernier.com/til/1014/*
2. Understanding PCA *http://setosa.io/ev/principal-component-analysis/*
3.