Machine Learning Engineer Nano degree

Capstone Project

Yudai Furukawa
May 14th, 2017

I. Definition

Project Overview

This project is about stock investing, and I am focusing on price prediction of a stock market index. A stock index is an aggregate value produced by combining several stocks, and it helps investors to measure and compare values of the stock markets such as in the US and Japan.
The Dow Jones Industrial Average (DJIA), NASDAQ Composite index and the S&P Composite are examples of stock index.
As a wealth of information such as price, earnings, dividends, and CPI are available, I am going to use those information to do the prediction.
A dataset of S&P Composite published by Yale Department of Economics will be used in this project. For more information, please refer the link below:
https://www.quandl.com/data/YALE-Yale-Department-of-Economics (https://www.quandl.com/data/YALE-Yale-Department-of-Economics)

Problem Statement

For this project, the task is to build a stock index price predictor. A 12 month forward price change of S&P composite will be predicted by using regression. The project is going to be a supervised learning.

Following steps will be taken to make the predictor.

1. Deciding inputs that are necessary to predict a 12 month forward price change by using correlation between each feature and 12 month forward price changes. The inputs are decided on the ground of common sense in the financial industry and statistical figures such as correlation.

2. Deciding the best regression model according to the metrics defined in the next section.

In the step 1, I am expecting PE Ratio and earning growth are going to be among the dominant inputs as it is commonly used in the financial industry to justify investment. In the step 2, I am expecting that r2 score will best serve the purpose although there are other metrics such as mean absolute error , mean squared error, and explained variance score as well as median absolute error. This will be discussed in the next section.

metrics

r2 score, explained variance score, and mean squared error, as well as explained variance score and median absolute error are all going to be used to validate the result. By using all the metrics, I can overcome the risk of being biased. Expected result is the best regression model have the highest score in all the scoring metrics.

II. Analysis

Data Exploration

A dataset of S&P Composite published by Yale Department of Economics will be used in this project.
The dataset (named snp in this project) is monthly time series of S&P Composite Price, Dividend, Earnings, CPI, Long Interest Rate, Real S&P Composite Price, Real Dividend, Real Earnings, and Cyclically Adjusted PE Ration since 1831-1-31 up to date.

For more information, please refer the link below:
https://www.quandl.com/data/YALE-Yale-Department-of-Economics (https://www.quandl.com/data/YALE-Yale-Department-of-Economics)

As of 2017-04-08, the basic statistics of snp the dataset is following.
Table 1

Statistics	S&P Composite	Dividend	Earnings	CPI	Long Interes Rate
count	1756.000000	1755.000000	1749.000000	1756.000000	1756.000
mean	242.537415	5.344903	12.046968	56.433670	4.584025
std	478.579184	9.010165	22.474642	69.298402	2.290630
min	2.730000	0.180000	0.160000	6.279613	1.500000
25%	7.680000	NaN	NaN	10.100000	NaN
50%	16.005000	NaN	NaN	18.100000	3.870000
75%	115.550000	NaN	NaN	84.200000	5.240000
max	2357.000000	46.380000	105.960000	244.176000	15.32000

What can be concluded from table 1 is that there are huge deviation in most of the factors. Remarkably, the maximum price of S&P Composite is 863.3699634 times larger than its minimum price although the maximum earning is 662.25 times larger than its minimum and the maximum of dividend only 257.66 times.
This fact shows the S&P Composite historically advanced faster than earnings and dividend.

Also, as you can see some of the cells are filled by NaN as some data are missing in the dataset. Also, because when dealing with economical data, inflation has to be carefully taken into account as CPI tends to grow overtime and values of price and earnings tend to have smaller values in the past. Therefore, only real values, Long Interest Rate, and Cyclically Adjusted PE Ratio in the previous table can be taken seriously in statistical analysis without any modification.

Exploratory Visualization

Figure 1: Time Series for all factors

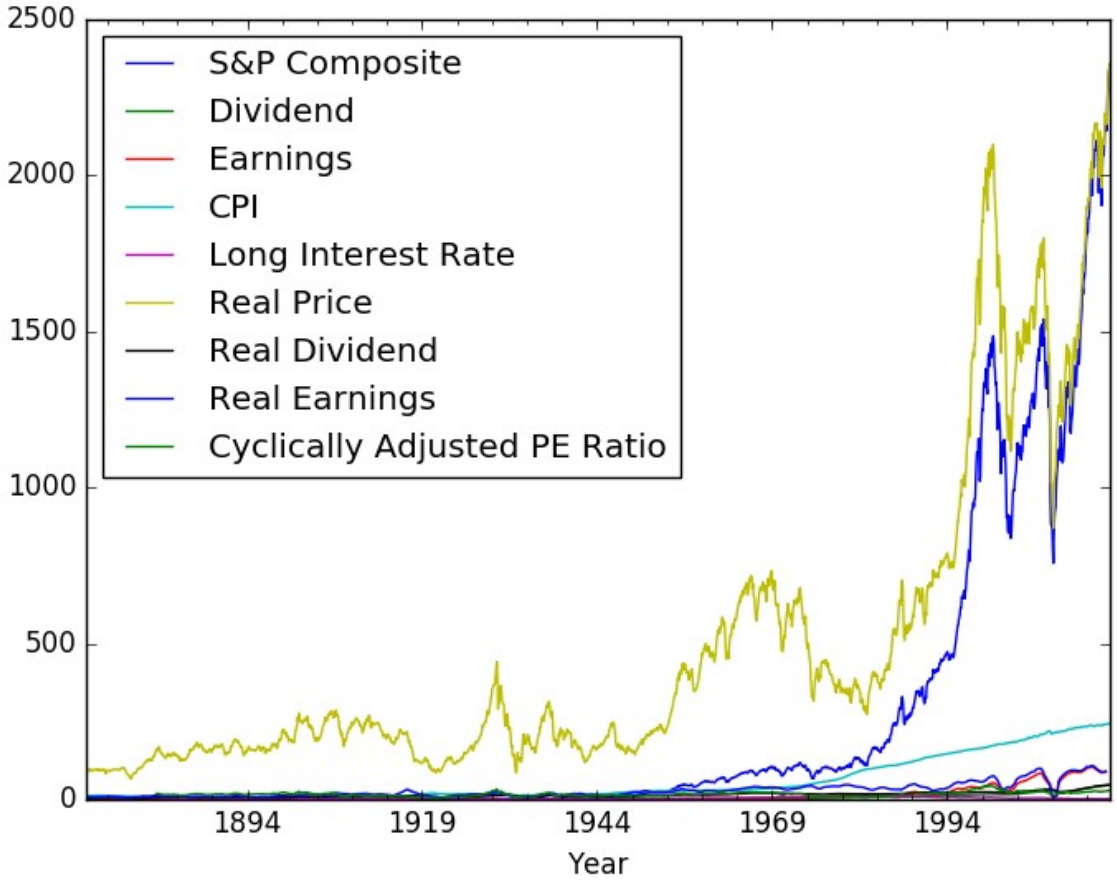


Figure 2: Time Series for Factors Except “Real Price” and “S&P Composite”

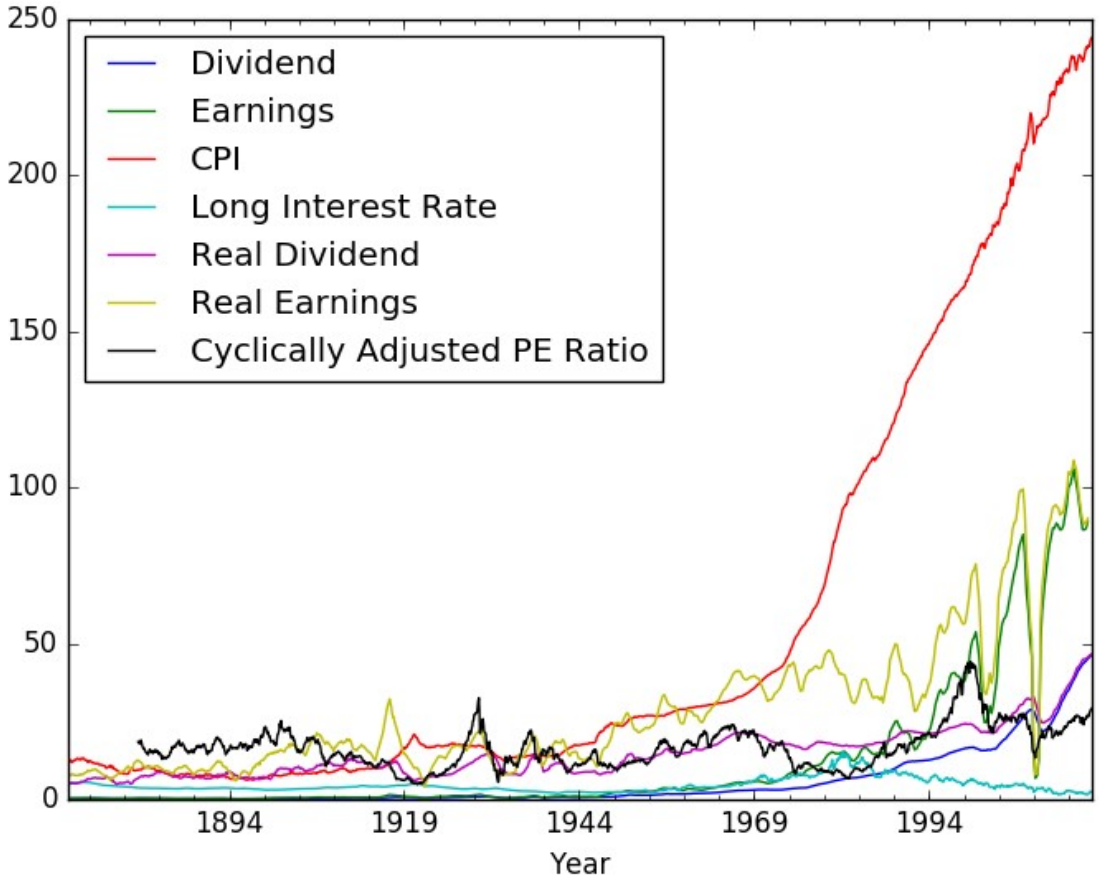


Figure 1 is a time series plot of the raw dataset. Figure 2 is also time series plot omitting “Real Price” and “S&P Composite” in order to visualize other features clearly. Time series plot was chosen as the data itself is a time series. For the both figures, the horizontal axis displays years and the vertical axis displays figures for each feature where the unit is different for each feature.

In figure 1, you can clearly see both “Real Price” and “S&P Composite” have been increasing in general as time goes. Also, in figure 2, you can see the same trends besides for “Cyclically adjusted PE Ratio” and “Long Interest Rate”. As the figures are increasing overtime, it is hard to compare the figures as it is. For example, the meaning of S&P Composite being 1000 right now and 20 years ago could have completely different meanings. In order to avoid this kind of misinterpretation, normalizing the data is necessary.

Algorithms and Techniques

In this project, following regressions will be used in predicting the 12 month forward price change in S&P Composite.

1. Linear Regression
    - Linear regression is an approach for modeling the linear relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X.
    - I chose the linear regression as it is the most commonly used regression in the financial industry. However, the weakness of the linear regression such as sensitivity to outliers and assumption of data independence should be carefully treated.
2. K Nearest Neighbors
    - In K Nearest Neighbors, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.
    - I chose KNeibors as it is robust to noisy training data and the dataset is noisy as a lot of other features that might affect the price are missing.
3. SVR
    - Support Vector Regression is very specific class of algorithms, characterized by usage of kernels, absence of local minima, sparseness of the solution and capacity control obtained by acting on the margin, or on number of support vectors, etc. It can be used to avoid difficulties of using linear functions in the high dimensional feature space and optimization problem is transformed into dual convex quadratic programmes.
    - I chose SVR as it can do both linear and non-linear regressions and it is less likely to over fit.

Benchmark

r2 score of 0.5 will be used as a benchmark. 0.5 is a reasonable benchmark as equity return is often said unpredictable. Also the result will be justified by the stability of r2 score, mean absolute error, and mean squared error, as well as explained variance score and median absolute error through back testing.

III. Methodology

Data Preprocessing

As discussed, the original dataset needs to be normalized in order to predict % change of

S&P Composite price with a return horizon of 1 year (named snp_changes). I first removed the columns for real values as this project aims to predict change of nominal S&P Composite price change and CPI in the dataset can be used to calculate the real value when needed. After this procedure, instead of using raw dataset, 12 months changes for each feature and the original figures of “Cyclically adjusted PE Ration” and “Long Interest Rate” will be used. The target feature will be 12 month forward changes in S&P Composite price.

The following modifications were made on the dataset.

1. Removed features with real values as the project is focusing on predicting % change of nominal S&P Composite price.
    - Features [‘Real Price’,’Real Earnings’,’Real Dividend’] were removed
2. Generated a dataset called snp_changes which shows 1 yea3 change of each feature in order to see the relationship of changes of 1 year S&P Composite price and other features

Before Change

Year	S&P Composite	Dividend	Earnings	CPI
1882-01-31 00:00:00	5.92	0.320000	0.439200	10.180580
1882-02-28 00:00:00	5.79	0.320000	0.438300	10.275745
1882-03-31 00:00:00	5.78	0.320000	0.437500	10.275745
1882-04-30 00:00:00	5.78	0.320000	0.436700	10.370911
1882-05-31 00:00:00	5.71	0.320000	0.435800	10.465995

Year	Long Interest Rate	Cyclically Adjusted PE Ratio
1882-01-31 00:00:00	3.620000	15.678764
1882-02-28 00:00:00	3.620833	15.153862
1882-03-31 00:00:00	3.621667	15.091670
1882-04-30 00:00:00	3.622500	14.916997
1882-05-31 00:00:00	3.623333	14.567103

After Change

Year	S&P Composite	Dividend	Earnings	CPI
1882-01-31 00:00:00	-0.043619	0.207547	-0.095924	0.080808
1882-02-28 00:00:00	-0.061588	0.185185	-0.090098	0.079999
1882-03-31 00:00:00	-0.073718	0.163636	-0.083770	0.079999
1882-04-30 00:00:00	-0.070740	0.142857	-0.077329	0.079216
1882-05-31 00:00:00	-0.121538	0.122807	-0.071185	0.099995

Year	Long Interest Rate	Cyclically Adjusted PE Ratio
1882-01-31 00:00:00	-0.021622	-0.151304
1882-02-28 00:00:00	-0.019630	-0.164950
1882-03-31 00:00:00	-0.017631	-0.173970
1882-04-30 00:00:00	-0.015625	-0.168975
1882-05-31 00:00:00	-0.013612	-0.228017

1. Added a target feature “y” which is 12 months forward return of S&P Composite.

Year	y
1882-01-31 00:00:00	-0.061588
1882-02-28 00:00:00	-0.073718
1882-03-31 00:00:00	-0.070740
1882-04-30 00:00:00	-0.121538
1882-05-31 00:00:00	-0.136778

2. Added following features that seem to be have an effect on the prediction.
    - The real value of PE Ratio as the ratio is considered as a good indicator on predicting return in general. As investpedia says, “The P/E ratio is a much better indicator of the value of a stock than the market price alone, since it allows investors to make a better apples to apples comparison”.
3. Removed outliers
    - Outliers are omitted to avoid having a misleading r2 score - it has a weakness as the score could be greatly affected by unusual data points

Implementation

TimeSeries Split Validator was used to test the models. This validator provides train/test indices to split time series data samples that are observed at fixed time intervals, in train/test sets. This cross-validation object is a variation of KFold. In the kth split, it returns first k folds as train set and the (k+1)th fold as test set.

Refinement

At the beginning I used algorithms with default settings to determine which algorithm best serves the purpose of this project. After determining the algorithm, I used grid search in order

to optimize the algorithms in this project. I used the default number of splits which is 3.

Step 1: Choose Algorithm

```
estimator = SVR()
estimator = KNeighborsRegressor()
estimator = LinearRegression()
```

Step 2: Grid Search

```
estimator = grid_search.GridSearchCV(SVR(kernel='rbf', gamma=0.1), cv=5, param_grid={"C": [1e0, 1e1, 1e2, 1e3],"gamma": np.logspace(-2, 2, 5)})
estimator = grid_search.GridSearchCV(KNeighborsRegressor(), param_grid={"n_neighbors": [2,3,4,5,6,7,8,9,10]})
estimator = grid_search.GridSearchCV(LinearRegression(), param_grid =  {'fit_intercept':[True,False], 'normalize':[True,False], 'copy_X':[True, False]})
```

IV. Results

Model Evaluation and Validation

Followings are the result of each scores with split

SVR
Split 1
('r2 score:', 0.46203561013012351, 'explained variance score:', 0.49076547652335589, 'mean_squared_error', 0.027909821203550064, 'mean_absolute_error', 0.13322021924068453, 'median_absolute_error', 0.11494323190727181)
Split 2
('r2 score:', 0.83734051188973591, 'explained variance score:', 0.8507121009308416, 'mean_squared_error', 0.0036388683253097179, 'mean_absolute_error', 0.048233671571316319, 'median_absolute_error', 0.042793838457600555)
Split 3
('r2 score:', 0.56601185038138779, 'explained variance score:', 0.57365252325183336, 'mean_squared_error', 0.011315896786256328, 'mean_absolute_error', 0.072512443085194361, 'median_absolute_error', 0.047410680757838497)

K Nearest Neighbor
Split 1
('r2 score:', 0.11366052722544762, 'explained variance score:', 0.27650168616411541, 'mean_squared_error', 0.045983668578453943, 'mean_absolute_error', 0.16797549806049597, 'median_absolute_error', 0.14124764701469139)
Split 2
('r2 score:', 0.6262866037854673, 'explained variance score:', 0.63084140197008631,

‘mean_squared_error’, 0.0083603720633077076, ‘mean_absolute_error’, 0.071785369871896237, ‘median_absolute_error’, 0.062301103917071922)
Split 3
(‘r2 score:’, -0.082982563354474292, ‘explained variance score:’, 0.10866035141914021, ‘mean_squared_error’, 0.028237911378465357, ‘mean_absolute_error’, 0.12316307176270737, ‘median_absolute_error’, 0.089980760066028259)

Linear Regression
Split 1
(‘r2 score:’, 0.88629761910370441, ‘explained variance score:’, 0.88728464199424062, ‘mean_squared_error’, 0.0058989278491112474, ‘mean_absolute_error’, 0.051686380139549411, ‘median_absolute_error’, 0.037683001046082007)
Split 2
(‘r2 score:’, 0.89932003229322677, ‘explained variance score:’, 0.90247836633119305, ‘mean_squared_error’, 0.0022523195525675855, ‘mean_absolute_error’, 0.036520785100067572, ‘median_absolute_error’, 0.028205368886686434)
Split 3
(‘r2 score:’, 0.9020251503591914, ‘explained variance score:’, 0.90603497761453722, ‘mean_squared_error’, 0.0025546164962307679, ‘mean_absolute_error’, 0.037571621770448774, ‘median_absolute_error’, 0.02698937431809878)

As you can see Linear Regression scored better in almost all the score. Also more stability in scores overtime is observed for Linear Regression compared to other regressions. For example, the r2 scores of linear regression stays around 0.9 whereas obvious instability of r2 score is observed for SVR and K Nearest Neighbors.

Therefore, I concluded linear regression best serves the purpose for this project.

After using grid search, not much improvement was observed. The scores are followings.

Split 1
(‘r2 score:’, 0.89664579541310629, ‘explained variance score:’, 0.89691708583413254, ‘mean_squared_error’, 0.0061842954759759924, ‘mean_absolute_error’, 0.051265662455921866, ‘median_absolute_error’, 0.037261406664000601)
Split 2
(‘r2 score:’, 0.89986981562675339, ‘explained variance score:’, 0.90286244646868152, ‘mean_squared_error’, 0.0022638811791995807, ‘mean_absolute_error’, 0.036607046703997399, ‘median_absolute_error’, 0.028266085435894095)

Split 3
('r2 score:', 0.90054123332737157, 'explained variance score:', 0.90377986316285341, 'mean_squared_error', 0.0027188916770536373, 'mean_absolute_error', 0.038394690588562007, 'median_absolute_error', 0.026809489549802112)

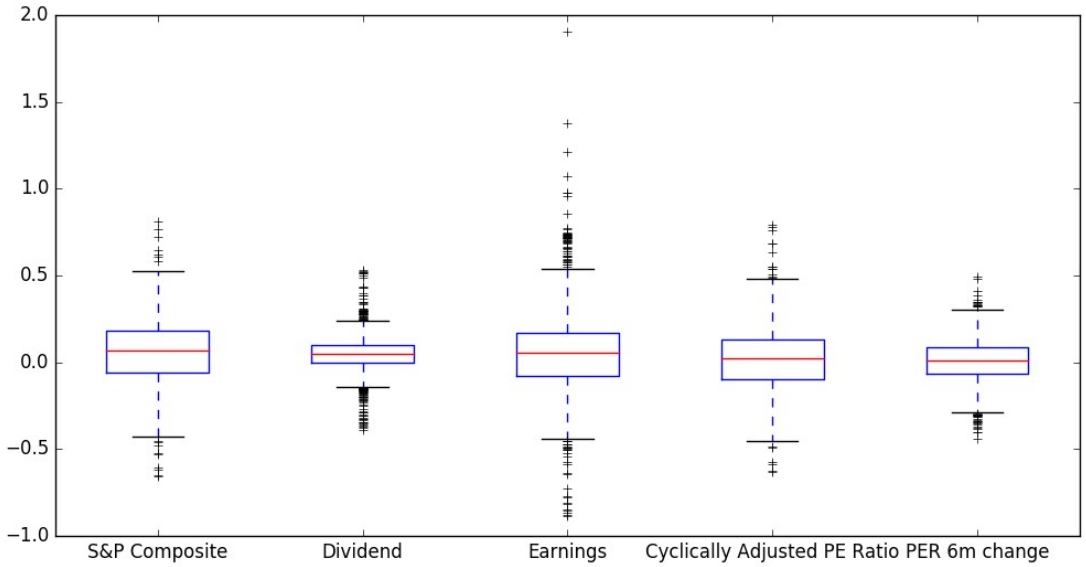
Justification

r2 score of 0.91 with the selected linear regression is definitely higher than benchmark which is r2 score of 0.5.
The result is reasonable as the scores for linear regression were stable through Split 1, 2, and 3. r2 score, mean absolute error, and mean squared error, as well as explained variance score and median absolute error were reasonably stable compared to other regressions. This also shows the model seems not to be overfitting it has similar score over back-testing (Split 1, 2, and 3)
Also the result could be justified as linear regressions are historically working well in economics and finance field. This result is significant as the project shows existing factors can be good predictors of the future S&P Composite return although it is often said that the it is nearly impossible to predict the future return of S&P Composite.

V. Conclusion

Free-Form Visualization

Figure 3: % change in each factor (24 month % change for 'price 24m change'. 12 months % change for other factors)



This graph is a box plot of the dataset. As you can see mean values of all the features are above 0. Especially as the mean value of y is above 0, you can make money on average when you invest in S&P Composite.

Reflection

I started this project with searching for datasets available in order to understand what kinds of datasets I can use. I checked morning star, Google finance, and yahoo finance as well as quandl, and decided to use quandl database as it seemed to have more various datasets compared to other sources.

After searching data, I defined the goal for this project as the problem could be either classification and regression. I was thinking of using classification as well (for example I could divide the returns into some categories such more than 10% increase, less than 10% change etc.) but decided to use regression.

I analyzed the characteristics of the dataset in order to understand how I can use the dataset to achieve the goal and decided to normalize the data in order to make it usable for regressions. Decided to use parametric returns for normalizing the dataset.

After having the dataset ready, I chose what regressions and scoring metrics I will use for the project, then compared each regressions according to the scores. I had to be careful about the score as the model could be over-fitting and the score could be biased. Finally I tried to refine the regression having the best score, and concluded the model which best serves the purpose.

I found it interesting that I could define the problem as any of supervised, unsupervised, and reinforcement learning although the data being used is the same. I found it difficult to find data that I need for the project as most of datasets are not free.

The final result fit my expectations for the problem as the linear regression is the most commonly used regression in the financial industry, and I think

Improvement

I think I could have more factors such as fundamental data (book value, working capital etc.) and economic data (GDP, NFP, etc.) in the dataset in order to make a better stock price predictor. With more factors in the dataset, I expect to have a better regression with higher r2 score.