ASSIGNMENT 2

ANDREW ID: parmenin
18-788: Big Data Science
4/17/23

Niyomwungeri Parmenide ISHIMWE

I, the undersigned, have read the entire contents of the syllabus for course 18-788 (Big Data Science) and agree with the terms and conditions of participating in this course, including adherence to CMU's AIV policy.

Signature: **Niyomwungeri Parmenide ISHIMWE**

Andrew ID: **parmenin**

Full Name: **Niyomwungeri Parmenide ISHIMWE**

LIBRARIES USED

- import numpy as np
- import pandas as pd
- import matplotlib.pyplot as plt
- import warnings
- from sklearn import svm
- from sklearn.metrics import mean_absolute_error, mean_squared_error
- from sklearn.linear_model import LinearRegression
- from sklearn.ensemble import GradientBoostingRegressor

QUESTION 1:

The two Welch and Goyal predictor datasets were required to be loaded into the environment (Jupyter Notebook). They offer data on a variety of economic variables that can be used for in-sample prediction for equity premiums. The datasets were loaded by downloading them as Microsoft Excel files and using the pandas' read_excel[1] function to read from them.

QUESTION 2:

It was required to recreate the explanatory variables using original data up to 2005 and plot labeled time series subplots for them to confirm they are correct interpretations. This was done by first, reading the Welch and Goyal paper[2] to understand their meaning and how they are implemented. Those variables are: "**Default Yield Spread, Inflation, Stock Variance, Dividend Payout ratio, Long Term Yield, Term Spread, Treasury-bill rate, Default Return Spread, Dividend Price Ratio, Dividend Yield, Long Term Return, earning price ratio, Book to market, and Net Equity Expansion**" and were recreated into a python pandas data frame using their formulas and methods provided in the paper as shown in code.

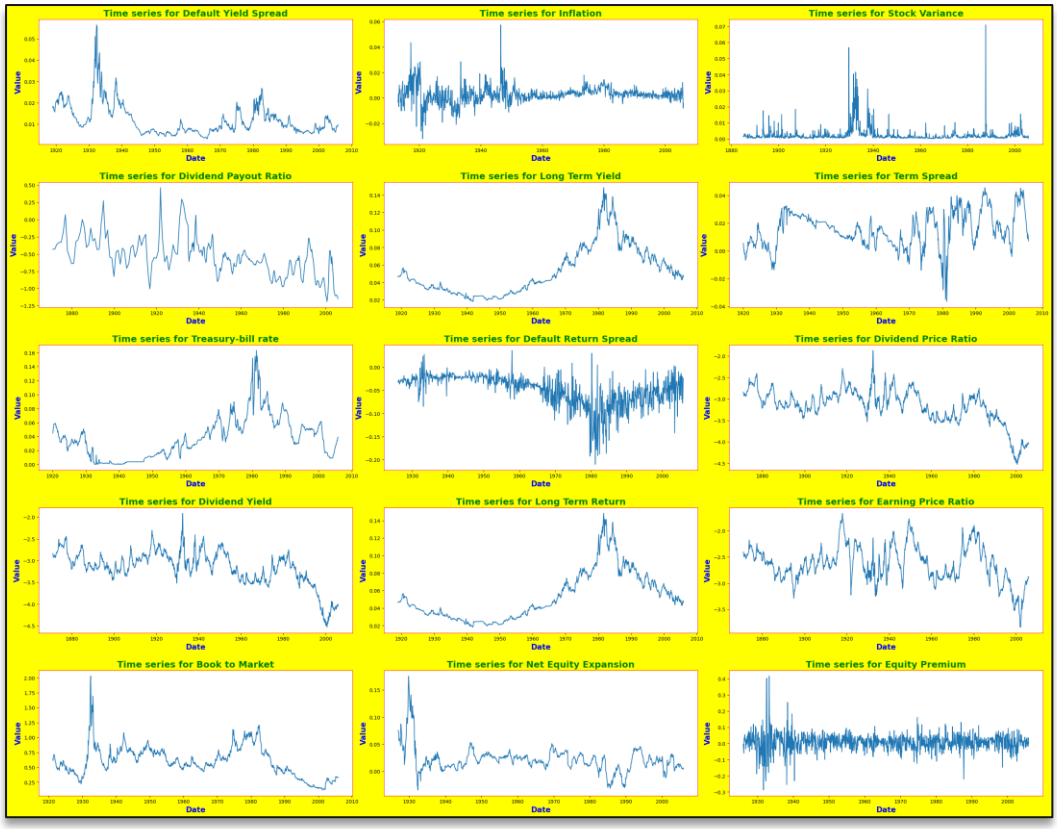

Figure 1: The subplots for explanatory variables

After recreating the variables, the above graph with subplots for each of the variables was plotted using the matplotlib.pyplot's subplots function[3] for plotting and subplots_adjust[4] for adding margins.

It can be inferred from the subplots that in the period around 1930, some dependent variables like default yield spread, dividend payout ratio, stock variance, term spread, default return spread, dividend price ratio, dividend yield, book-to-market, and net equity expansion experienced higher values. Similarly, the period around 1980 has been with greater values for many variables like long-term return, earning price ratio, default return spread, treasury-bill rate, and long-term yield.

Furthermore, the year 2000 has been giving smaller values for various explanatory variables including book-to-market, dividend yield, earning price ratio, dividend price ratio, and dividend payout ratio.

Moreover, despite fluctuating in its values, the independent variable, the equity premium has shown both its peak value and its slowest value during the year 1930 of around 0.4 and 0.3 respectively.

QUESTION 3:

It was asked to use the rolling multiple regression model described in Welch and Goyal's paper by using the updated data up to 2021 for the period from January 1965, January 1976, and January 2000 to December 2008 and provide a table about out-of-sample R^2, RMSE, and MAE. In-sample data were used for training and out-sample data are used for testing. This was done by first selecting data for required periods, then making a function to compute the rolling multiple regression using the linear regression model with the help of the LinearRegression function from sklearn.linear_model[5]. Then, call the function by passing each of the required period data frames and the linear regression model to provide R^2, RMSE, and the MAE depicted below in the tables.

R SQUARED	RMSE	MAE
-0.361384	0.050988	0.037396

Figure 2: R^2, RMSE, and the MAE for the 1965 – 2008 period.

R SQUARED	RMSE	MAE
-0.533698	0.054348	0.038846

Figure 3: R^2, RMSE, and the MAE for the 1976 – 2008 period.

R SQUARED	RMSE	MAE
-2.349439	0.083578	0.054089

Figure 4: R^2, RMSE, and the MAE for the 2000 – 2008 period.

From the above tables R^2 for the three periods are -0.361384, -0.533698, and -2.349439 which seems to be high and highly changing between those three periods. In addition, the errors, RMSE is 0.050988, 0.054348, and 0.083578, and MAE are 0.037396, 0.038846, and 0.054089, which seems to be giving the low value of errors even though they are not consistent with those three periods of time.

QUESTION 4:

It was required to improve the results of the models on R^2, RMSE, and MAE values. This was done using two methods which are Support Vector Regression from the svm[6] function, and boosting to improve prediction using the GradientBoostingRegressor function from sklearn.ensemble[7]. To do this, the same function from the above question was used by passing its time frame's data frame and each of the Support Vector Regression and GradientBoostingRegressor models. All that process provided the following tabled results:

R SQUARED	RMSE	MAE
-0.18778	0.047626	0.03709

Figure 5: R^2, RMSE, and the MAE for SVR's 1965 – 2008 period.

R SQUARED	RMSE	MAE
-0.288057	0.049806	0.038927

Figure 6: R^2, RMSE, and the MAE for SVR's 1976 – 2008 period.

R SQUARED	RMSE	MAE
-0.028397	0.046311	0.035603

Figure 7: R^2, RMSE, and the MAE for the 2000 – 2008 period for SVR.

R SQUARED	RMSE	MAE
-0.345292	0.050686	0.037496

Figure 8: R^2, RMSE, and the MAE for the 1965 – 2008 period for GradientBoostingRegressor

R SQUARED	RMSE	MAE
-0.39926	0.051911	0.03893

Figure 9: R^2, RMSE, and the MAE for the 1976 – 2008 period for GradientBoostingRegressor

R SQUARED	RMSE	MAE
-0.368819	0.053429	0.03944

Figure 10: R^2, RMSE, and the MAE for the 2000 – 2008 period for GradientBoostingRegressor

From the above results, after comparing with that LinearRegression, we can infer that the prediction accuracy has been lifted a bit since the R^2 is being increased, the RMSE is being decreased, and the MAE is also being increased for each time frame (1965, 1976 and 2000 until 2008) for both Support Vector Regression (SVR) and GradientBoostingRegressor.

In addition, the SVR model is performing well compared to the GradientBoostingRegressor model since it has larger R squared, and small values for both RMSE and MAE errors.

REFERENCES

[1] ‘pandas.read_excel — pandas 2.0.0 documentation’. https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html (accessed Apr. 15, 2023).

[2] A. Goyal and I. Welch, ‘A Comprehensive 2021 Look at the Empirical Performance of Equity Premium Prediction II’, presented at the Review of Financial Studies, Jul. 2008.

[3] ‘17. Creating Subplots in Matplotlib | Numerical Programming’. https://python-course.eu/numerical-programming/creating-subplots-in-matplotlib.php (accessed Mar. 25, 2023).

[4] ‘Matplotlib Subplots_adjust - Python Guides’, Sep. 16, 2021. https://pythonguides.com/matplotlib-subplots_adjust/ (accessed Mar. 25, 2023).

[5] ‘sklearn.linear_model.LinearRegression’, *scikit-learn*. https://scikit-learn/stable/modules/generated/sklearn.linear_model.LinearRegression.html (accessed Apr. 15, 2023).

[6] A. Sethi, ‘Support Vector Regression Tutorial for Machine Learning’, *Analytics Vidhya*, Mar. 27, 2020. https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/ (accessed Apr. 15, 2023).

[7] A. Jain, ‘Complete Machine Learning Guide to Parameter Tuning in Gradient Boosting (GBM) in Python’, *Analytics Vidhya*, Feb. 21, 2016. https://www.analyticsvidhya.com/blog/2016/02/complete-guide-parameter-tuning-gradient-boosting-gbm-python/ (accessed Apr. 15, 2023).