Project Report – John Enright

Github URL

https://github.com/DeerparkMul/SpecialistProject-UCDPA_John-Enright

Abstract

The money supply increased by 40 percent between March 2020 and December 2021 (American Farm Bureau Foundation 2022) meaning we are living in a time of unprecedented growth in the money supply. This will most likely affect the economy in a number of ways. However, I just wanted to look at how the increase in money supply effects the stock market. Therefore, I put forward the hypothesis that an increase in the money supply leads to an increase in the stock market. I also wanted to test just how strongly related factors such as earnings and dividends affect the level of the stock market.

Introduction

The issue of quantitative easing has become more and more important recently as central banks pump more and more money into the system to insure economic stability during covid and before that during the eurozone and financial crises. I will measure the effect the expansion of the monetary base has had on the stock market. Also, if this turns out to have such a strong positive effect on the stock market then the period of monetary tightening which we are now entering may have a strong negative affect on the stock market and therefore the economy. I will also examine the relationship between dividends and earnings and the level of the S&P500. Given that the economy effects very important aspects of everyone's life it is important to understand the effects of certain policy decisions on the economy. Monetary policy is one of the main tools governments have at their disposal to manage the economy and therefore we should analyse its effects on key economic indictors such as stock market performance.

Dataset

The first dataset I used was from Kaggel and it was based on information from the St Louis Fed- this shows the monetary base since 1959 but I will only use it from 1960. I used this datasource as it is from the federal reserve and therefore the most reliable source of information relating to money supply. The second dataset I got from data hub and has information about the S&P500 going back to 1871. This is a reliable source of data. The last dataset I used from Kaggel came from and was parsed from twitter and so is reliable. The tweets related to 6 different stocks between 2015 and 2020.

Definitions

- The S&P500 is one of the largest stock markets in America and is made up of 500 large companies.

- Earnings are the amount of money made by a company. For the purposes of this dataset earnings amounts to the total amount of earnings for the entire S&P500

- Dividends amount to the figure that are distributed by the company. For the purposes of this dataset dividends refer to the entire amount of dividends for the S&P500.

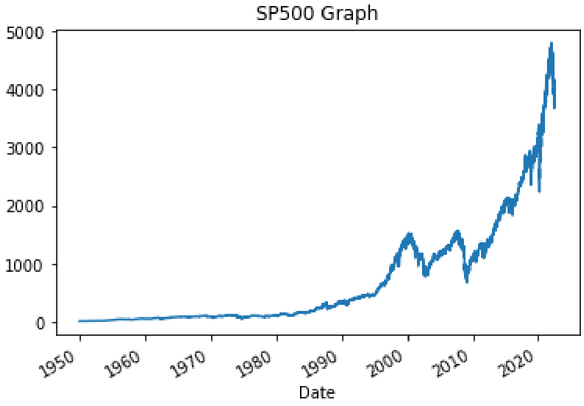
Implementation process

- Import packages such as pandas, numpy, seaborn and matplotlib
- Read first csv file relating to monetary base – df
- Use .iloc and .columns to select only the years I want and the columns I want
- Use .iloc again to select every 12th row so I have the value from the start of the year
- Use .reset_index() to reset index
- Read the dataset which has data from the S&P 500 -dfs
- I used head, tail, info, describe and nunique to analyse the dataframe.
- I went through the same process with iloc and columns and reset_index as the last dataframe
- I then did an index join
- I then checked if there were any na's
- I then used seaborn to print a scatterplot showing the relationship between the size of the monetary base and the value of the S&P500
- Imported packages such as yfinance , requests, io to webscrape prices from yahoo finance for the S&P500 since 1950
- Webscraped a graph from yahoo finance for the S&P500 since 1950
- I created a graph of the cumulative returns on the S&P 500 since 2000
- I then created a graph showing the breakdown of the S&P500 by industry in July 2020
- I used a stats.linregress to put a regression line through the scatterplot showing the relationship between the size of the monetary base and the value of the S&P500
- I imported sklearn.
- I ran train_test_split
- I ran a Decision Tree Classifiers
- I fitted and predicted the model. Then I got its accuracy score.
- I then predicted the S&P 500 levels based on dividends and earnings
- I then got the coeffeients of the regression model
- I then produced a scatterplot with SP500 vs Dividends and with earnings as the hue

- I created a correlation map of the merged dataframe
- I created a correlation map of the S&P500 dataframe
- I removed the na's from the dfs dataframe
- I used the reg_all score to test the model where dividends and earnings are used to predict S&P500
- I then calculated the R^2
- I then got the Ridge score
- I then got the tuned elasticnet R squared
- Converted from regression into classification- the monethary base versus SP500
- Got roc_auc_score of classification
- Did Grid Search on the classification
- Did grid Search best estimator
- Did Random Search best estimator
- Calculated metrics.accuracy_score
- I then got F1 score
- Printed knn score
- I read the tweets csv file into python
- I used iloc to just take the first 100 rows
- I then imported packages such as nltk and SentimentIntensityAnalyzer
- I created a new data frame with "id" and "comment" fields, removed all non-alphabet characters, converted to lower-case
- I then got a sentiment score for each tweet and merged this back onto the original dataframe and then graphed this by date
- I then imported packages like cv2, imutils, Image, Glob, Pil etc
- I read in the two images of the head of the ECB and the Federal Reserve
- I resized them
- I got the differences between the two images and created an image that only showed difference between the two images. Almost the entire image was bright which shows that these images were very different.
- I created a new column in dfs which shows just the year
- I used a function to create a column that divides Dividends over Earnings that shows the percentage of Earnings that are going into dividends

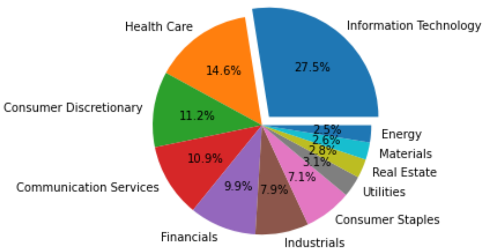
- I cleaned the Date column using regex so that all rows would be in the format yyyy-mm-dd
- I used an iterator comparing dividends to long term interest rates
- I then created a word cloud showing the most frequent PE ratios as per the data
- I created a new column which measured the dividend yield
- I then compared the dividend yield to the long term interest rate
- I ran a function that computed the average increase in the SP500

Results

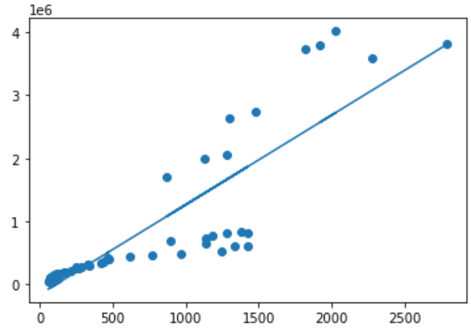


This chart shows the level of the S&P500 since 1950. It should be noted that much of the gains have occurred in recent years.

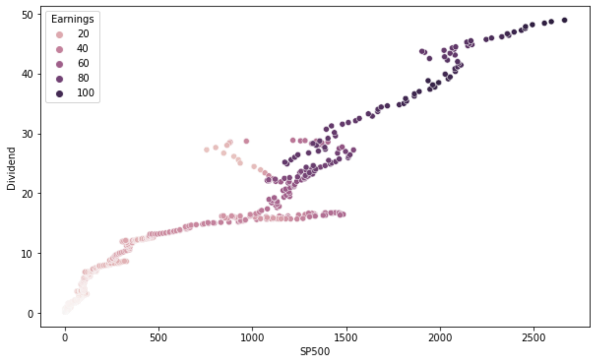
This chart shows the cumulative gains of the S&P500 since 1/1/2000. It should be noted that much of the gains happen in recent years.



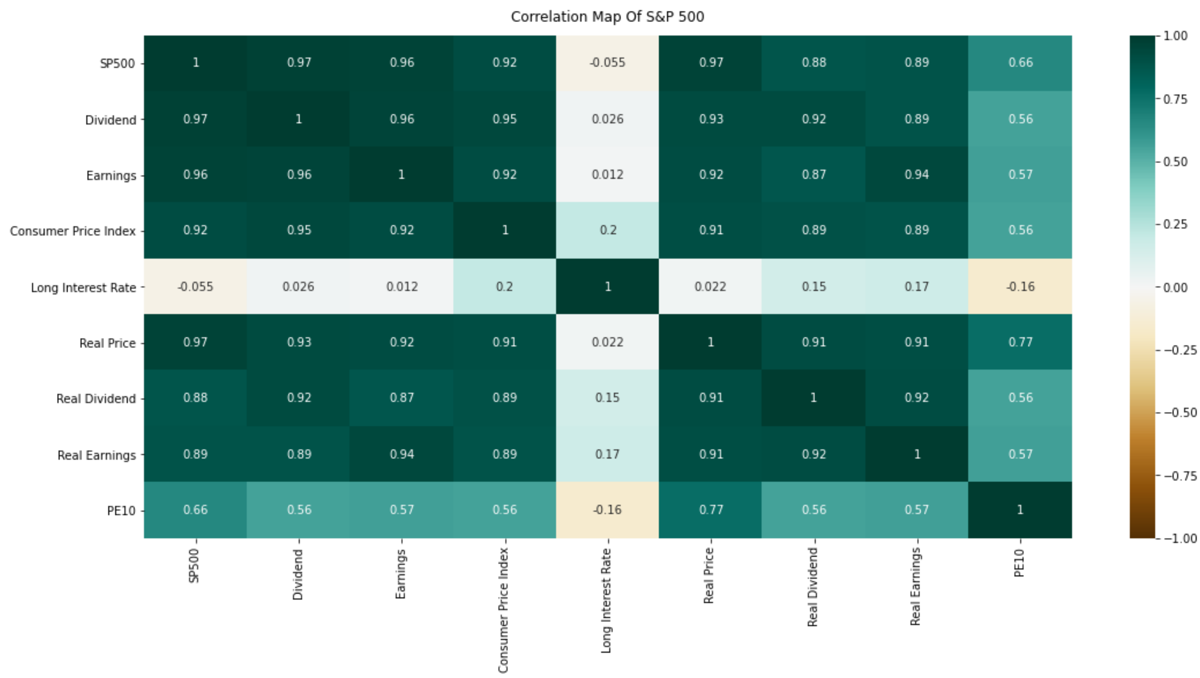
This pie chart shows the breakdown of S&P500 by industry in July 2020



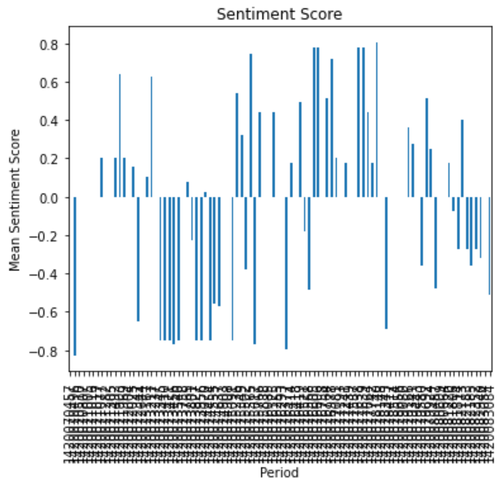
This scatter plot with regression line shows the relationship between monetary base and S&P500. It shows positive correlation.



This scatterplot shows the relationship between S&P500 and Dividends and Earnings(hue). You can see all these relationships are positive and hue(Earnings) gets stronger as the other two increase.



This heatmap shows the different correlation between the different variables. There exists a strong correlation between all variables except Long Term Interest rates.



This graph of sentiment scores shows the sentiment of texts on different dates

This wordcloud shows the most common PE ratios. A PE ratio is the price earnings ratio and is the multiple of company's current earnings that its market value currently stands at.

Quantitative results

Correlation between Monetary base and S&P500 is 0.87

Regression model based on Monetary base vs S&P500

- Accuracy score= 0.9166666666666666

Regression model based on Earnings, Dividend vs S&P500

- reg_all.score= 0.9537183900532339
- R^2: 0.9537183900532339
- ridge.score= 0.9531436637135209
- Tuned ElasticNet R squared: 0.9551305614897474

Classification based on Monetary base vs S&P500

- roc_auc_score= 0.9285714285714286
- Gridsearch= .93333
- roc_auc_score(grid_search.best_estimator)= 0.7142857142857143
- roc_auc_score(random_search.best_estimator)= 0.928571
- metrics.accuracy_score = 1
- F1 score=.96
- knn.score=0.95833

Insights

- The heatmap shows a very strong correlation between the level of the S&P500 and other factors such as dividends and earnings in both real and nominal terms. This is to be expected as a company is really only worth what it earns. The correlation between S&P500 and earnings and dividends are over 0.9 in nominal terms and over 0.8 in real terms.This means that if the earnings of companies increase the value of the stock market should increase.
- Long term interest rates seem to have a very weak relationship with the level of the S&P500 and also other factors such as earnings and dividends. This is surprising as you would think higher interest rates would reduce earnings and dividends and therefore the level of the S&P500.
- A sentiment analysis of a small number of tweets relating to a selection of 6 companies show that sentiment was mixed in the case of these tweets
- An analysis of the difference between the photo's of Fed Chairman and President of the ECB show the photos to be completely different
- The correlation between the monetary base and the S&P500 is 0.87. This is a strong positive correlation and suggests an increase in monetary base will lead to an increase in the S&P500. This means that the period of monetary tightening we are now entering into may have a negative effect on the stock market and possibly by association the economy.
- Wordcloud shows the most common PE ratios are 18 and 17 but with a significant range. This is to be expected as stock markets tend to go from boom to bust where they are overvalued to where they are undervalued. The current PE ratio of the S&P500 is 30.5(Current Market Valuation 2022) which is high if you analyse the wordcloud.
- The accuracy tests run on both regression models and the classification model suggest the models are accurate
- Random Search parameter hyper tuning seems to produce a better result than Grid search parameter hyper tuning for the classification model
- I created a new column showing dividends over earnings. This shows that on average 61.3% of earnings were distributed as dividends but this varied

widely since 1871 as the interquartile range was between 48.5% and 68.4%

- I created a column which shows the dividend yield of the SP500 over time and then compared this to the long term interest rate. Surprisingly dividend yields were higher 60.69 per cent of the time. Given the average annual increase of the SP500 since 1871 has been 7.15 which is positive it would suggest placing your money in the S&P500 has historically on avearge been a better strategy to maximise returns on capital than placing it in a bank.

When and why to use trees:

There are both positive and negative aspects of using trees as well as how ensemble learning can improve the process

Benefits - Trees are more flexible with gives them the ability to describe nonlinear relationships (Datacamp 2022)

Negatives - Trees are "very sensitive to small variations in the training set. Sometimes, when a single point is removed from the training set, a CART's learned parameters may change drastically." (Datacamp 2022)

Ensemble learning can be summarized as follows:

1. Different models are trained on the same data
2. Each model makes its own prediction
3. Then a meta model puts the predictions of individual models together and creates a final prediction

Doing this reduces the chance that the model will suffer from the problems of just running a normal tree model.

References

Current Market Valuation 2022 Available at https://www.currentmarketvaluation.com/models/price-earnings.php#:~:text=OV%20Mean%20Reversion-,Overview,P%2FE%20Ratio%20is%2030.5. (Accessed 12 August 2022)

American Farm Bureau Foundation 2022 Available at https://www.fb.org/market-intel/inflation-and-money (Accessed 12 August 2022)

Datacamp 2022 Available at https://campus.datacamp.com/courses/machine-learning-with-tree-based-models-in-python/the-bias-variance-tradeoff?ex=9 (Accessed 12 August 2022)

Datacamp 2022 Available at https://campus.datacamp.com/courses/machine-learning-with-tree-based-models-in-python/the-bias-variance-tradeoff?ex=9 (Accessed 12 August 2022)