<https://github.com/nrodgers05/UCDPA_Natalia.git>

**Project Report**

Prior to the start of my project, I had no inclination toward any financial dataset I wanted to analyse. During our course, it became apparent that COVID lockdown restriction were to be extended in Ireland and that the vaccine rollout in Europe was going to be slightly slower than once anticipated. While browsing for dataset ideas on Kaggle, I came across a dataset for vaccinations by country over various time periods (from whichever date countries began vaccinating) and this piqued my interest.

In the same time period, watching Bloomberg TV made me realise that the share markets of developed countries were doing comparatively well considering the global recession on the back of certain technology shares which were able to take advantage of the move to online learning like that we are undertaking. Considering that technology companies have a larger weighting in some indices like the US, this was not surprising but when looking at more traditional indices such as the UK which has a higher proportion of mining and brick-and-mortar companies, it was. Remembering the dataset from Kaggle, I decided I would use this project to try to simply analyse any relationship between the rate at which vaccinations were given in countries and the rise in the leading share market indices of each country. In order to maintain the coding as the focal point of the project, I decided to keep the financial analysis relatively simple and choose only three countries and their share indices to analyse, being the UK (& FTSE 100), the US (& S&P500) and Australia (&S&P/ASX 200).

In order to do concentrate my analysis, I used a location function (df.loc) in order to locate the three countries, I wanted to focus on from the list of 135 countries available from Kaggle. I then dropped the unnecessary columns from the dataset, leaving me with dates, the country names and the number of people vaccinated per 100 (which is akin to the share of the population vaccinated as a percentage). In order to do this, I used the numpy array to create a list which allowed me to order the sequence of columns which I wanted to drop and define them in one object. I could then use the .drop function in order to drop the specific columns I listed in the numpy array without worry that non-specified columns would also drop out of the table. I opted to use the list over the dictionary function because of its ability to keep the order of column names and therefore remove the probability of error.

I then realised that there were a number of days in which there were no values given for the share of vaccinated population. This has been a well-known issue whereby days such as Christmas day or New Years have had statistical lags because of fewer people working on those days. In order to be factual, I decided to replace these no-value days with a 0 value using the fillna(0) function. After looking back at the graphs created, I realise it would have been better to use a previous value instead of 0 as the share of the population vaccinated would not have suddenly dropped – only stayed the same. Only the first derivative of population vaccinated would have been 0. After filling missing data, I indexed the table imported to give each date and country its own unique id code.

Rather than have the data begin from 0, I updated the index so that the first row would be indexed as 1. This was to stop any accidental deletion or manipulation of the title bars which I wanted to keep as they were. When I initially tried to create sub-tables for graphing for each of the countries from the finaltable, this would have proved useful however due to the graphing functions need to have a CSV file to refer to, I had to change approach. This is detailed further later in the report.

I visited Yahoo.com as I knew from previous experience that historical data was available there for any number of shares or indices and that it was exportable to excel. I found the price history for the days of which I had vaccine data (which granted due to its contemporaneous nature was not significant), and I exported these and saved them as CSV files. This meant that I could use the merge function to put the two tables together and create finaltable. This table shows the date, country, number of people vaccinated and the closing price of each of the three major share indices. In order to merge the two files, I had to use the ‘id’ column created when indexing due to the fact that a number of countries shared the same dates, and a number of dates shared the same country. This meant that in order to match the correct country and date in each table, I needed to use a unique code or identifier that was equivalent to the two datapoints.

In order to create country-specific graphs, it was necessary for me to import the matplotlib.pyplot function as plt and to create country specific CSV’s from the original CSV in order to import them. I originally used the table already created and a df.loc function again to isolate only one country at a time from the table containing three countries. I then intended to plot using these user-defined sub-tables however the column named ‘closing\_price’ consistently gave me errors in finding it. After thoroughly searching, I decided to recreate the CSV files in three separate country-specific versions. Using the matplotlib function, I created graphs from the three imported CSV files for each of the countries. Initially, I had trouble as both the vaccines per hundred of population and the closing prices of indices were being plotted on the same axis. This was obviously a problem due to the fact that vaccines per hundred were comparatively tiny (around 0.5) while closing prices varied between 3-7 thousand. In order to visualize both sets of data on an appropriate axis, I used the twinx function to create a second y-axis for the closing price. This means that both vaccines delivered and closing price were on appropriate scales and therefore we could draw some conclusions from this.

Given more time, I would have like to learn how to code a regression which wul allow me to use a larger dataset and give a statistical figure for the amount of correlation between the vaccine rollout and the index price increase. I might also have liked to have focussed on one share index and shown how the vaccine rollout disproportionately aided companies which have been harder hit during the pandemic’s restrictions (which one would expect given the loosening of restrictions and the economic recovery to brick-and-mortar style stores one would expect from it). This would also have allowed me to give some form of causality argument between the loosening of restrictions and the increase in index price.

**Insights from visualisation**

When looking at the data you can see there is a positive correlation between the vaccination numbers and the closing prices for stock prices, thus indicating that countries with higher vaccinations numbers are likely to have a higher index closing price. Obviously a causality argument is not able to be put forward from this graph as there are too many extraneous variables however one can surmise that a vaccine rollout coincides with an increase in index closing price.

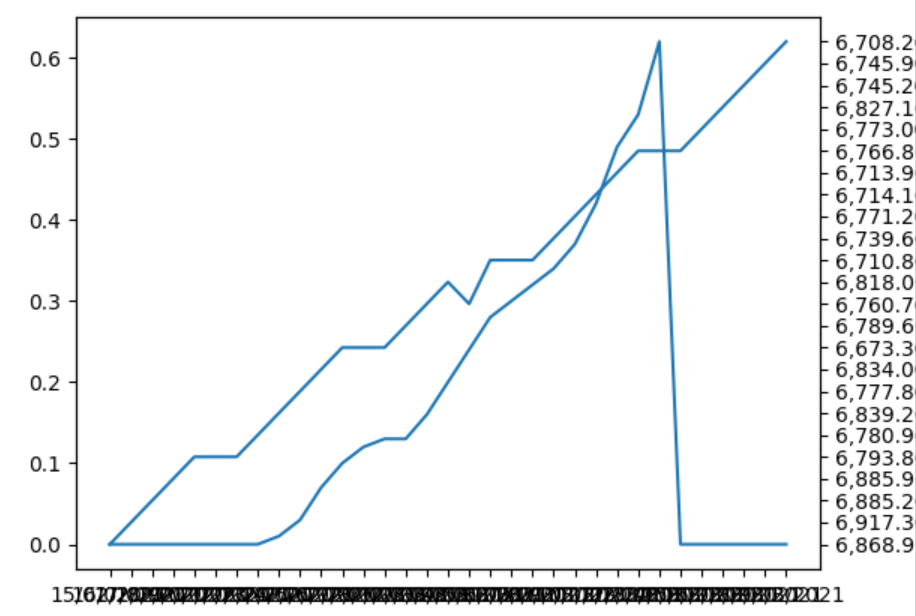
This is most obvious in the Australian country chart, which shows no vaccine data for the final days o the time frame. One could argue that the restrictions being successful in Australia meant that no vaccines were needed to be given for the economy to normalise and share prices to increase. This however would be erroneous due to the fact that the country cannot function on domestic demand alone and will need to open borders at some point – this would be baked into share market prices.

One important insight gained from the comparison of these charts is the difference in the rate of vaccination even throughout the developed world. One can see, knowing that the data is contemporaneous, that the UK has vaccinated far higher a percentage of the population than the US and Australia has.

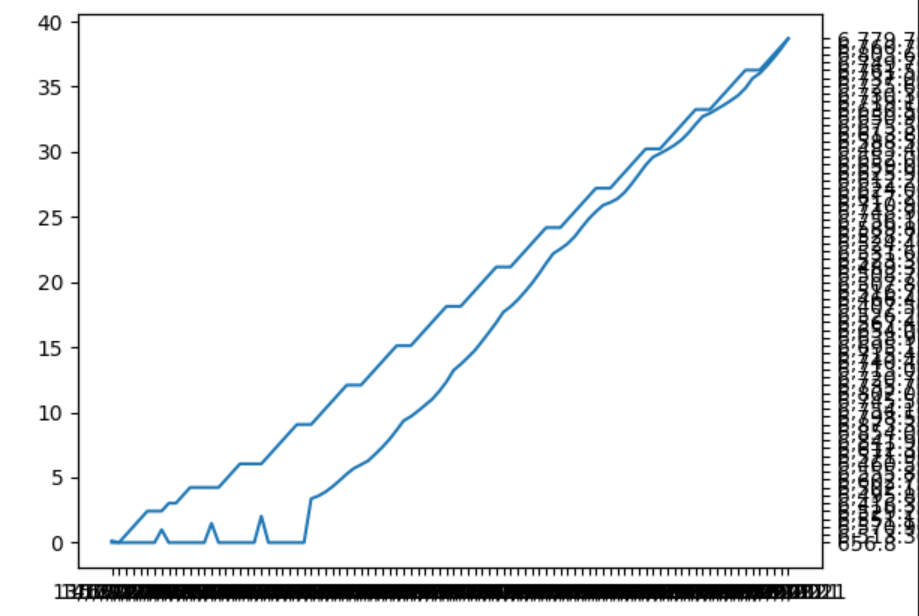
Looking at the visual outputs, it is obvious that using a 0 value for days in which no vaccinations were reported, although technically correct, creates outliers to the data and does not accurately represent the share of the population that have been vaccinated and a prior-value or a moving average of the prior data should have been used. A moving average would have been slightly skewed due to the short span of the data used (COVID only having begun in 2020) however it should have been at least considered as a better alternative to using 0.

In future, there are two things I would add to these simple visualisations – firstly, I would tidy the axis by defining their limits and their spacing. A key would also be helpful or axis labelling although in the case below, it is rather intuitive that one cannot have 6000 cases per 100 people of vaccination. It would also be helpful to be able to colour code the lines of the graph for aesthetic reasons, say if one were to do a presentation.

**Australia data**



**UK data**



**US data**

