Project Report

# GitHub URL

<https://github.com/BarryMcPadden/UCDPA_BarryMcPadden.git>

# Abstract

What I have looked at in my project is the impact on FX rates for the first couple of days in 2016. There was a lot going on in the world in 2016 which had major impacts

# Introduction

With the US being the largest Economy in the world since 1871, the last 6 years have created a lot of negative publicity about the stability of the Economy largely due to events both internally & Externally. As the Fed has reported inflation of 7% year to December 2021, what is the impact on the US Exchange Rate & what has the impact been as a result of events which have unfolded over the past number of years.

Add this into the Brexit situation which dominated 2015 & 2016 in advance of the referendum & also the upcoming presidential race in the US, I wanted to look at the impact on the markets in the early part of 2016 & see what signs were given to the market.

I wanted to use the data to see how the markets for each currency reacted to the uncertainty & wanted to compare that against a more stable currency such as Australian Dollar & Danish Krone. Danish Krone is pegged against the Euro so this would show how a steady currency should react.

# Dataset

The dataset I am using as part of my project is the Central Bank of Ireland Exchange rates. These can be found in the following location;

[Euro Exchange Rates | Central Bank of Ireland](https://www.centralbank.ie/statistics/interest-rates-exchange-rates/exchange-rates)

The above file what I used for uploading a CSV file via Pandas. The reason I have chosen this source is that it provides the spot rate at a given time for the Exchange rates to the Euro.

I used the following link to upload via API into JupyterLab

<https://api.coingecko.com/api/v3/exchange_rates>

This was the current real time exchange rates

# Implementation Process

First part of my project was to decide on the data I was going to use as part of my project. I decided to go with both a CSV upload which contained Fx rates from 2016 & also did an API plug in which was of the current exchange rates.

The first thing I did was I found an API of the current exchange rates. As the code shows, I created a link to bring these into my project.

I then went to the Central Bank website & downloaded the FX rates from here. I filtered these in Excel to show the Jan 2016 rates & converted these to CSV. Once converted I then uploaded these through Pandas into python. These formed part of the Data Extraction stage.

Once the files were in Python I worked on Sorting, Indexing & grouping the data. The data I worked off was the data I imported via csv. I started off by sorting the data in ascending order. I did this by using the command sort\_index. Once I did this, I then sorted the data by Column.

I then used the function groupby to try & see what the max & min of the dataset turned out to be.

Once I had that done I then ran functions to run loops & iterrows in the data set. These functions can be used to treat the data frame like a dictionary.

I then went on in trying to merge both the data frames I had together in Python. This was done using pandas & I used the function ‘merge’. I tried to merge them using USD as a common value.

Once I had done the above, I then imputed function into my workflow to try & define a custom function to create a useable code. I used the code def(AUD2EUR) which was to try & workout out what a single unit of AUD was in Euro. This would allow me to compare other currencies & it was giving a Euro value of 1 unit of the other currency. This would feature later in the project for what I was trying to achieve.

With the Data I had in, I converted it using NumPy. With this I used the Dictionary function to assign values to both the USD & AUD in order to come up with a value for both look ups.

I then used the list function to create a list of values for AUD. I was able to use these lists to filter through the list & create a value for the reference assigned to each value for e.g. the references were 0-8.

Once I was happy with this I then proceeded to create charts through Matplotlib. Unfortunately the reports would not work as per the formula I was using so I inputted the formulas & for illustration purposes I have created the charts in Excel.

The first chart I created was to try and track the AUD movement over a 4 day period in January 2016. The purpose of this chart was to try & see how the currency started off the year.

The Second chart I created was to try & check the movement on the Euro values of DKK, USD, GBP & AUD currencies & compare the movements against each other. This would give an indication of how volatile the currency is & see if that was reflective of the time back in 2016.

I also tried to run a stack-plot as part of my project. I was unable to but the function I have used is within the project.

The choice of my dataset & the charts I have used allow me to check what the volatility of the different currencies were at a time when there was a lot of uncertainty in the global market.

# Results

With the 2 graphs I ran the following results came from them;

AUD Movement

With this graph we can see that there was a good start to the year when the price went from €0.66 per AUD$1 up to €0.668 per AUD$1.

However we seen that over the next few days there was a drop down to €0.645 per AUD$1. This may not seem like a lot in monetary terms but if you break this down into percentage terms this leads to a percentage drop between the 5th – 7th January of 3.44% over 2 days.

Comparing this to the bigger picture & looking at Danish Krone (DKK), US Dollar (USD) & Pound Stirling (GBP), we can see the following;

We see in that 4 day period the currency with the biggest change in position was the GBP as the movement between both the first & the last day was the most.

USD & AUD stayed roughly the same with a slightly bigger move between USD & Euro.

We can see that DKK has remained roughly static which was to be expected given that the Euro & the DKK are pegged against each other.

# Insights

* The graph shows the Exchange Rate with the lowest movement in comparison to the Euro is DKK. This would be expected as DKK is pegged to the Euro
* The graph shows the highest movement is GBP. This data is 6 months before Brexit referendum where there was uncertainty as how the UK was going to vote
* AUD & USD were steady around that time
* API’s can be used to link in real time information to a database
* The use of Panda’s can be used to bring in data in CSV files

# References

Import Pandas

<https://youtu.be/ENhGz1HkzvY>

Import Numpy

<https://youtu.be/l4nE2WUb8vQ>

**API Import**

[**https://api.coingecko.com/api/v3/exchange\_rates**](https://api.coingecko.com/api/v3/exchange_rates)

Creating custom function in python

<https://youtu.be/TF_uoQ8Aw54>

Dictionary’s in Python

<https://youtu.be/ZEZdys-fHDw>

Lists in Python

<https://youtu.be/ohCDWZgNIU0>