

The Persevering Path of the US Dollar's Value Across the Last Three Presidencies

Introduction

The first purpose of this project is to ascertain how the Euro to USD value fluctuated during the last three presidencies in the United States of America (as of January 2021). The [dataset](#) was provided by the European Central Bank and is hosted and continuously updated on Kaggle by Daria Chemkaeva. The second purpose is to engage in this project in order to particularly improve data visualization skills.

Summary of Results

The worst value for the Euro to USD conversion was almost 1.6 between 2007 and 2008, a timeframe in which a great recession took place. It later hovered around 1.4 for 5 years, from 2009 to 2014. Since 2015, it remained mostly below 1.2. However, as of 2021, it is slightly above 1.2. Lastly, the total average for the entire period was 1.22. There is a graph that demonstrates this, which was one of the objectives of this project. What follows this section is the cleaning of the data and the plotting of the main graph.

Learning About the Data's Structure

In [13]:

```
#Reading the file into pandas

import pandas as pd

exchange_rates = pd.read_csv('euro-daily-hist_1999_2020.csv')

#Inspecting the data

exchange_rates.head()
```

Out[13]:

Period\Unit:	[Australian dollar]	[Bulgarian lev]	[Brazilian real]	[Canadian dollar]	[Swiss franc]	[Chinese yuan renminbi]	[Cypriot pound]	[Czech koruna]	[Dan krør]
0	2021-01-08	1.5758	1.9558	6.5748	1.5543	1.0827	7.9184	NaN	26.163
1	2021-01-07	1.5836	1.9558	6.5172	1.5601	1.0833	7.9392	NaN	26.147
2	2021-01-06	1.5824	1.9558	6.5119	1.5640	1.0821	7.9653	NaN	26.145
3	2021-01-05	1.5927	1.9558	6.5517	1.5651	1.0803	7.9315	NaN	26.227
4	2021-01-04	1.5928	1.9558	6.3241	1.5621	1.0811	7.9484	NaN	26.141

5 rows × 41 columns

In [14]: #Inspecting the last few records

```
exchange_rates.tail()
```

Out[14]:

	Period\Unit:	[Australian dollar]	[Bulgarian lev]	[Brazilian real]	[Canadian dollar]	[Swiss franc]	[Chinese yuan renminbi]	[Cypriot pound]	[Czech koruna]	[]
5694	1999-01-08	1.8406	NaN	NaN	1.7643	1.6138	NaN	0.58187	34.938	
5695	1999-01-07	1.8474	NaN	NaN	1.7602	1.6165	NaN	0.58187	34.886	
5696	1999-01-06	1.8820	NaN	NaN	1.7711	1.6116	NaN	0.58200	34.850	
5697	1999-01-05	1.8944	NaN	NaN	1.7965	1.6123	NaN	0.58230	34.917	
5698	1999-01-04	1.9100	NaN	NaN	1.8004	1.6168	NaN	0.58231	35.107	

5 rows × 41 columns



In [15]:

#Verifying to see if there are missing rows

```
exchange_rates.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5699 entries, 0 to 5698
Data columns (total 41 columns):
 #   Column           Non-Null Count  Dtype  
 --- 
 0   Period\Unit:      5699 non-null   object 
 1   [Australian dollar ] 5699 non-null   object 
 2   [Bulgarian lev ]    5297 non-null   object 
 3   [Brazilian real ]   5431 non-null   object 
 4   [Canadian dollar ]  5699 non-null   object 
 5   [Swiss franc ]     5699 non-null   object 
 6   [Chinese yuan renminbi ] 5431 non-null   object 
 7   [Cypriot pound ]    2346 non-null   object 
 8   [Czech koruna ]    5699 non-null   object 
 9   [Danish krone ]    5699 non-null   object 
 10  [Estonian kroon ]  3130 non-null   object 
 11  [UK pound sterling ] 5699 non-null   object 
 12  [Greek drachma ]   520 non-null    object 
 13  [Hong Kong dollar ] 5699 non-null   object 
 14  [Croatian kuna ]   5431 non-null   object 
 15  [Hungarian forint ] 5699 non-null   object 
 16  [Indonesian rupiah ] 5699 non-null   object 
 17  [Israeli shekel ]  5431 non-null   object 
 18  [Indian rupee ]   5431 non-null   object 
 19  [Iceland krona ]   3292 non-null   float64 
 20  [Japanese yen ]   5699 non-null   object 
 21  [Korean won ]     5699 non-null   object 
 22  [Lithuanian litas ] 4159 non-null   object 
 23  [Latvian lats ]    3904 non-null   object 
 24  [Maltese lira ]    2346 non-null   object 
 25  [Mexican peso ]    5699 non-null   object 
 26  [Malaysian ringgit ] 5699 non-null   object 
 27  [Norwegian krone ] 5699 non-null   object 
 28  [New Zealand dollar ] 5699 non-null   object 
 29  [Philippine peso ]  5699 non-null   object 
 30  [Polish zloty ]    5699 non-null   object
```

```

31 [Romanian leu ]      5637 non-null   float64
32 [Russian rouble ]    5699 non-null   object
33 [Swedish krona ]     5699 non-null   object
34 [Singapore dollar ]  5699 non-null   object
35 [Slovenian tolar ]   2085 non-null   object
36 [Slovak koruna ]     2608 non-null   object
37 [Thai baht ]         5699 non-null   object
38 [Turkish lira ]       5637 non-null   float64
39 [US dollar ]          5699 non-null   object
40 [South African rand ] 5699 non-null   object
dtypes: float64(3), object(38)
memory usage: 1.8+ MB

```

There are a high number of null values for multiple currencies. However, the focus is going to be the US dollar, which seems to be complete, especially when comparing it to other currencies with 5,699 rows. The next steps will be to clean the dates and the currency because they have an object format.

Creating a Dataframe Centered Around the USD

The first step is to reset the index of the records for the purpose of starting with the oldest year instead of the most recent one. In addition, turning the dates into datetime objects.

```
In [16]: exchange_rates.rename(columns={'[US dollar]': 'US_dollar',
                                    'Period\\Unit': 'Time'},
                           inplace=True)

exchange_rates['Time'] = pd.to_datetime(exchange_rates['Time'])

exchange_rates.sort_values('Time', inplace=True)

exchange_rates.reset_index(drop=True, inplace=True)

print('The "Time" column\'s new datatype : ',
      exchange_rates['Time'].dtype)

exchange_rates.tail()
```

The "Time" column's new datatype : datetime64[ns]

Out[16]:

	Time	[Australian dollar]	[Bulgarian lev]	[Brazilian real]	[Canadian dollar]	[Swiss franc]	[Chinese yuan renminbi]	[Cypriot pound]	[Czech koruna]	[Danish krone]
5694	2021-01-04	1.5928	1.9558	6.3241	1.5621	1.0811	7.9484	NaN	26.141	7.4379
5695	2021-01-05	1.5927	1.9558	6.5517	1.5651	1.0803	7.9315	NaN	26.227	7.4387
5696	2021-01-06	1.5824	1.9558	6.5119	1.5640	1.0821	7.9653	NaN	26.145	7.4393
5697	2021-01-07	1.5836	1.9558	6.5172	1.5601	1.0833	7.9392	NaN	26.147	7.4392
5698	2021-01-08	1.5758	1.9558	6.5748	1.5543	1.0827	7.9184	NaN	26.163	7.4369

5 rows × 41 columns

It looks good. The next step is to take a further look at the 'Time' and 'US_dollar' columns.

```
In [17]: euro_to_dollar = exchange_rates[['Time', 'US_dollar']]
euro_to_dollar['US_dollar'].value_counts()
```

```
Out[17]: -      62
1.2276    9
1.1215    8
1.1305    7
1.3532    6
..
0.9687    1
1.4155    1
1.3195    1
1.4471    1
1.4462    1
Name: US_dollar, Length: 3528, dtype: int64
```

There are too many '-' values, so those will be dropped. In addition, the 'US_dollar' column will be converted to a float data type.

```
In [18]: rows_to_keep = euro_to_dollar['US_dollar'] != '-'

euro_to_dollar = euro_to_dollar[rows_to_keep]

euro_to_dollar['US_dollar'] = (euro_to_dollar
                               ['US_dollar'].astype(float))

print(euro_to_dollar['US_dollar'].value_counts())

print('\n', '"USD" column\'s new datatype : ',
      euro_to_dollar['US_dollar'].dtype)
```

```
1.2276    9
1.1215    8
1.1305    7
1.1268    6
1.3373    6
..
1.4639    1
1.5206    1
1.3272    1
0.8591    1
0.9375    1
Name: US_dollar, Length: 3527, dtype: int64
```

"USD" column's new datatype : float64

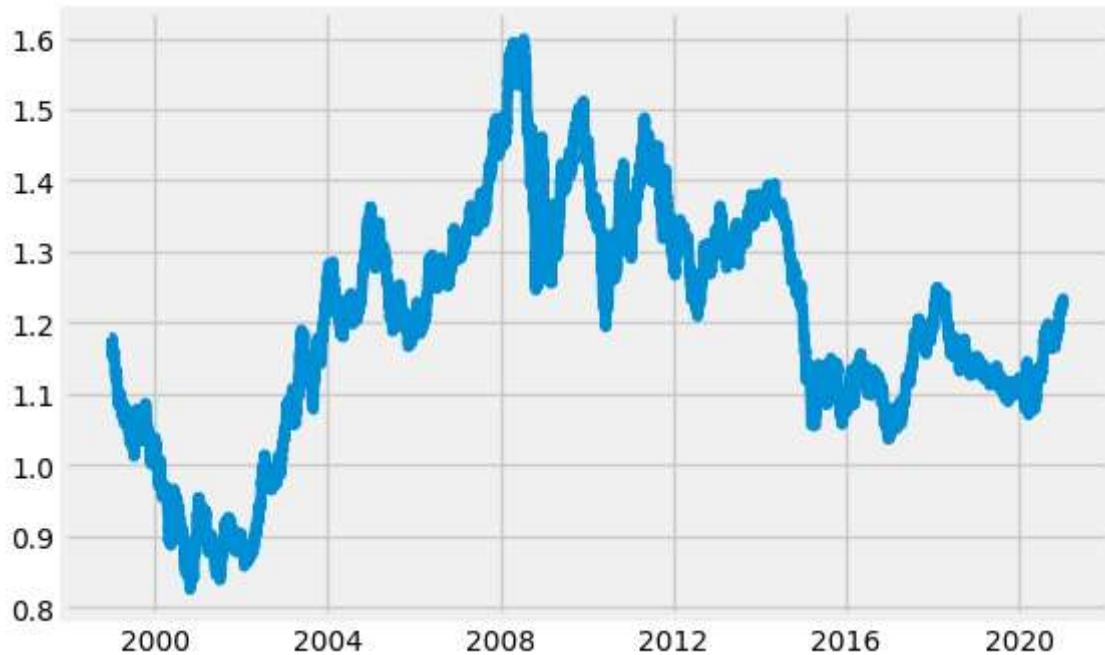
With this step complete, a test graph can be run. For this test, the change of the USD's value over time will be plotted, since the dataset starts in 1999.

```
In [19]: #Running a test plot

import matplotlib.pyplot as plt
%matplotlib inline
```

```
plt.plot(euro_to_dollar['Time'],euro_to_dollar['US_dollar'])
```

Out[19]: [`<matplotlib.lines.Line2D at 0x7f7e33987dc0>`]

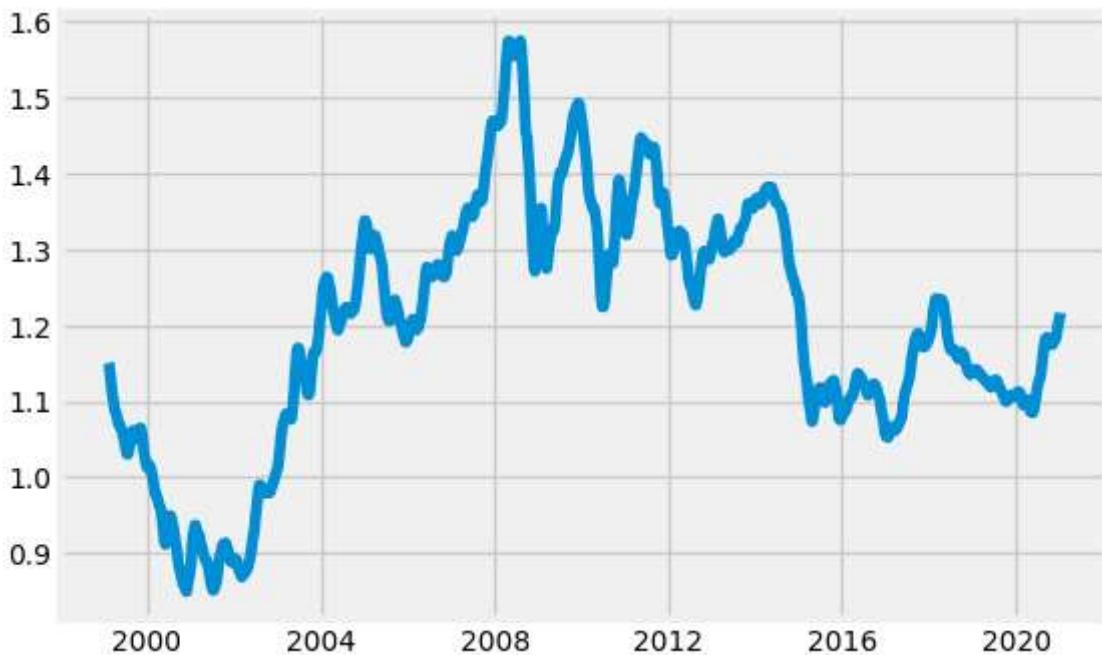


The USD's Value Since 1999 : Rolling Mean Test Plot

Smoothing out the lines of the plot above is necessary for focusing on long term trends. To address this, a rolling mean with a window of 30 days will be utilized.

```
In [20]: euro_to_dollar['Rolling_mean_30'] =(euro_to_dollar  
['US_dollar'].rolling(30).mean())  
  
plt.plot(euro_to_dollar['Time'],euro_to_dollar['Rolling_mean_30'])
```

Out[20]: [`<matplotlib.lines.Line2D at 0x7f7e2acd4070>`]



Since this change will help the legibility and usability of the graphs, the focus will now be shifted to making the final graph.

The Euro to USD Value's Change Throughout the Last Three Presidencies

The final plot is meant to portray the change of the Euro-USD value across the tenures of the last three US Presidents: George W. Bush, Barack Obama, and Donald Trump. It will capture a period that begins in 2001 and ends in 2021, which amounts to a total of 20 years. To accomplish this, the main objective is to create two snapshots of sorts, one section with a line plot for each of the tenures and another with a line plot combining all of the tenures.

In [21]:

```
#Importing the style module
import matplotlib.style as style
style.use('fivethirtyeight')

#Preparing the figure
fig = plt.figure(figsize = (12,13))

#These are the individual plots for each of the Presidents,
#the first section of the graph.
#The labels were placed in plots due to image size errors.

ax1_label = fig.add_subplot(3,3, 1)
ax2_label = fig.add_subplot(3,3, 2)
ax3_label = fig.add_subplot(3,3, 3)
ax1 = fig.add_subplot(3,3, 4)
ax2 = fig.add_subplot(3,3, 5)
ax3 = fig.add_subplot(3,3, 6)

#The combined plot to cover the second section of the graph
ax4 = fig.add_subplot(3,3, (7,9))

#Preparing the pairings used for the Loop
```

```

label_spots = [ax1_label,ax2_label,ax3_label]
label_x = [0.2, 0.25, 0.235] #The x_coordinates of the tenure labels
ax_list = [ax1,ax2,ax3]
labels = ['BUSH', 'OBAMA', 'TRUMP']
tenures = [(2001,2009),(2009,2017),(2017,2021)]
colors = ['#9467bd', '#ff7f0e', '#45b8b0']

pairings = zip(label_spots, label_x, ax_list, labels,
               tenures, colors)

for label_spot, x, ax, name, tenure, colour in pairings:

    #Plotting a line plot per ax for ax1, ax2, and ax3
    #as well as ax4, which will house all three line plots

    timeframe = '{}-01-01, {}-12-31'.format(tenure[0],tenure[1])

    beginning, end = timeframe.split()

    Time_bool = euro_to_dollar['Time'].between(beginning,end)

    ax.plot(euro_to_dollar[Time_bool]['Time'],
            euro_to_dollar[Time_bool]['Rolling_mean_30'],
            c = colour, linewidth = 4)

    ax.set_xlim(bottom = 0.8,top = 1.6)

    ax.set_yticklabels(['',1.0,1.2,1.4,1.6], alpha = 0.4)

    ax4.plot(euro_to_dollar[Time_bool]['Time'],
              euro_to_dollar[Time_bool]['Rolling_mean_30'], c = colour,
              linewidth = 4)

    #Editing the axes ax1_Label, ax2_Label, ax3_Label
    #to accomodate the Presidents' name and tenure labels

    label_spot.grid(b=None)

    label_spot.set_xticklabels([])

    label_spot.set_yticklabels([])

    label_spot.text(0.3, 0.3, name, c = colour,
                   fontsize = 25, fontweight = 'bold')

    tenure_label = '({}-{})'.format(tenure[0], tenure[1])

    label_spot.text(x, 0.1, tenure_label, c = 'grey',
                   fontsize = 19, fontweight = 'bold', alpha = 0.5)

    #Tweaking the xticks to improve the readability of Labels
    #along the x-axis of the graph

    tenure_gap = tenure[1]-tenure[0]

    n = tenure[0]

    #n is used to improve the readability of the argument
    #of ax.set_xticklabels

```

```

if tenure_gap > 4:

    simplified_xticks = [ ' ', n, ' ', n + 2, ' ', n + 4, ' ',
                           n + 6, ' ', n + 8]

else:

    simplified_xticks = [n, ' ', n + 1, ' ', n + 2, ' ',
                           n + 3, ' ', n + 4]

ax.set_xticklabels(simplified_xticks, alpha = 0.4)

#Final aesthetic changes for ax4, the ax with all the line plots

ax4.grid(b=None, which='major', axis='x')

ax4.set_yticklabels([' ',1.0,1.2,1.4,1.6], alpha = 0.4)

ax4.set_xticklabels([])

#This part of the code covers the calculation of the averages
#to be used for the titles.

all_tenures_bool = euro_to_dollar['Time'].between('2001-01-01',
                                                '2021-12-31')

all_tenures_avg = (euro_to_dollar[all_tenures_bool]
                   ['Rolling_mean_30'].mean())

#The creation of the titles, subdivided for the code's readability

title_part_1 = 'EURO_USD rate averaged {:.2f} under the last three'
title_part_2 = ' US presidents'

title_string = (title_part_1 + title_part_2).format(all_tenures_avg)

ax3_label.text(-2.7,0.92,title_string,
               fontweight = 'bold', fontsize = 22)

subtitle_part_1 = 'EURO-USD exchange rates under George W. Bush '
subtitle_part_2 = '(2001-2009), Barack Obama (2009-2017),'
subtitle_part_3 = ' and Donald Trump (2017-2021)'

#The reason for the following split is to maintain the readability
#of the subtitle in the graph as well as the code.

subtitle_A = subtitle_part_1 + subtitle_part_2

subtitle_B = subtitle_part_3

ax3_label.text(-2.7, 0.8, subtitle_A, fontsize = 18)

```

```

ax3_label.text(-2.72, 0.7, subtitle_B, fontsize = 18)

# Finalizing the graph with the signature

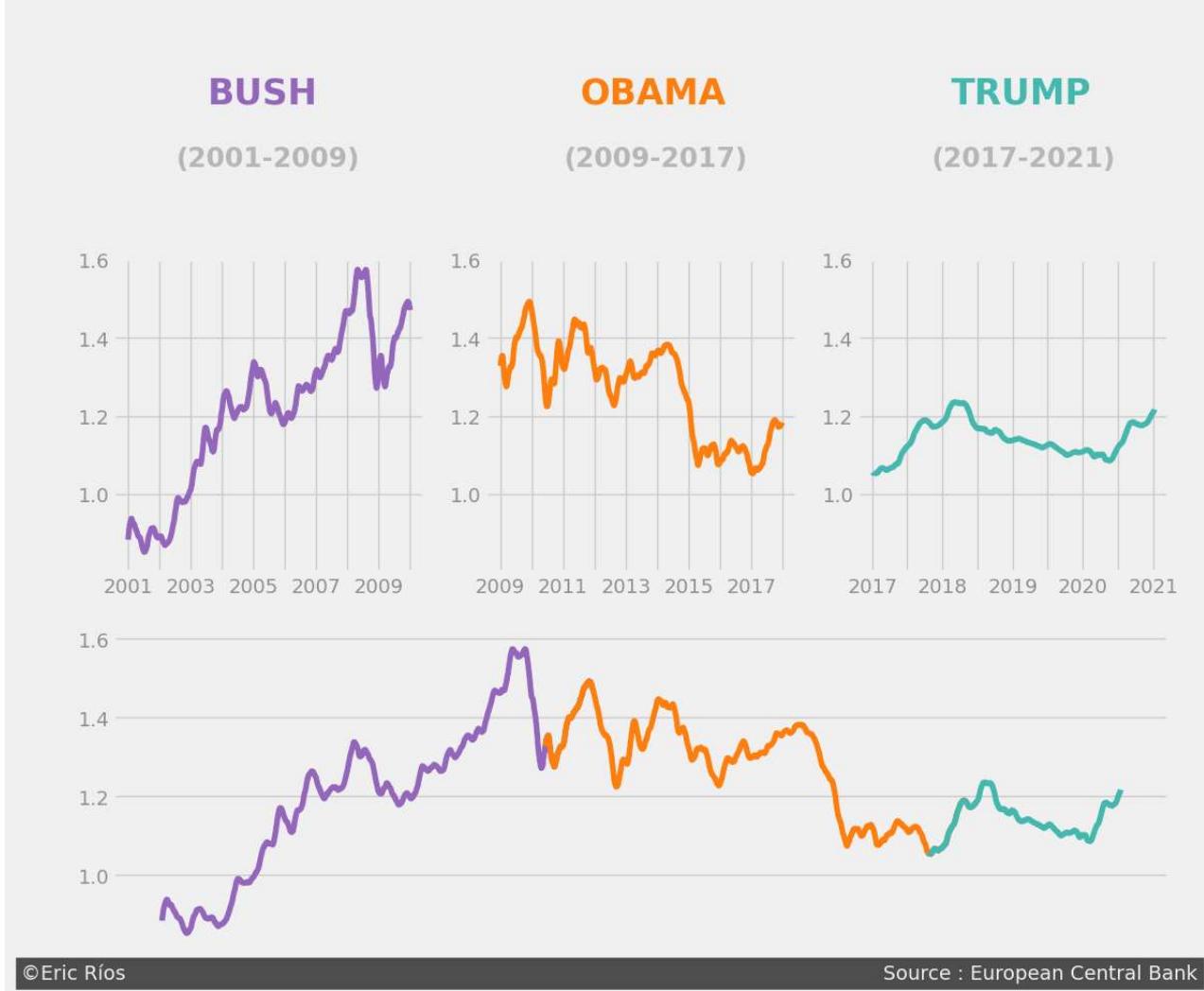
ax2_label.text(-1.5,-2.5,'@Eric Ríos' + ' '*120 +
               'Source : European Central Bank',
               backgroundcolor = '#4d4d4d', c = '#f0f0f0')

plt.show()

```

EURO_USD rate averaged 1.22 under the last three US presidents

EURO-USD exchange rates under George W. Bush (2001-2009), Barack Obama (2009-2017), and Donald Trump (2017-2021)



The worst conversion value for the USD peaked near 1.6 between 2007 and 2008, notable as it was a time of great recession. It remained somewhat precariously near the 1.4 mark from 2009 to 2014. As of 2015, it has stayed somewhat above or below 1.2. In addition, the total average for the entire period is 1.22. Lastly, the last presidency's tenure is a particularly interesting period, due to the world's citizens and economies suffering from Coronavirus pandemic's various ramifications. During that specific period, the value stayed below 1.2 and rose slightly above it in 2021.

Conclusion

The objective was to track and show the Euro-USD value's changes throughout the combined tenures of the last three presidents, ranging from 2001 until 2021. The approach consisted of cleaning the data by fixing formats, removing empty records, and applying the use of the rolling mean. All of these steps paved the way for the main graph. Said graph consisted of plots showing the USD values changing during each President's tenure as well as collectively across them.

At its worst state, the USD value was almost 1.6 (2007-2008), and its worst overall period was between 2008 and 2012. After roughly thirteen to fourteen years, it managed to place near 1.2, although it was slightly rising towards the beginning of 2021. In addition, it also averaged a value of 1.22 for the entire timeframe. Lastly, as food for thought, a potential and future variant for this study could examine the changes of the currency's value during the Coronavirus pandemic.