Analysing Economic Data from FRED in Python

Automatically Pulling and Analyzing Data from the FRED Databased in Python

John Karuitha

August 21, 2023

Background

In this analysis, I use data from the Federal Reserve bank of St Loius Database (FRED) to illustrate data analysis using Python and Pandas. As a starting point, I load the required packages- Pandas, Numpy, Matplotlib, Seaborn, and Plotly. The package `fredapi` does not come into Python by default. I hence install it and then load it.

Objectives

Summary of Results

The broad objective of the analysis is to showcase working with data from the FRED api in python.

The secondary objective of the analysis are as follows:

Data

I start by installing the packages fredapi and plotly.

```
```{python}
# !pip install fredapi
# !pip install plotly
```
```

Next, I load the required modules: matplotlib, numpy, seaborn, pandas, plotly, and fredapi.

```{python}
## Import the required modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import plotly.express as px
from fredapi import Fred

## Set options
plt.style.use("fivethirtyeight")
pd.set_option("max_columns", 20)
```

Next, we shift oir attention to our data source: FRED. FRED database is owned and maintained by the federal reserve bank of St. Louis in the United States. The database contains a vast array of economic variables from countries across the globe. The link to the database is https://fred.stlouisfed.org/. The database consists of `822,000` variables from `114` sources across the globe.

I load the data from the FRED api. To start downloading data, you have to request for an `api` key from FRED by creating an account. From your account, you can then get an api key. You will then create write code as follows:

```
fred_api = "your_fred_api_key_here"
```

I have hidden my API key for obvious reasons. Please generate your own api key from FRED.

Next, we create a Fred object to allow us to access the data.

```{python}
fred = Fred(api_key=fred_api)
```

From here, we can search or pull data from FRED.

```{python}
fred.search("Kenya")
```

series id	id	realtime_start	realtime_end	title
PTEAUSDM	PTEAUSDM	2023-08-21	2023-08-21	Global pr
PTEAUSDA	PTEAUSDA	2023-08-21	2023-08-21	Global pr
PTEAUSDQ	PTEAUSDQ	2023-08-21	2023-08-21	Global pr
FXRATEKEA618NUPN	FXRATEKEA618NUPN	2023-08-21	2023-08-21	Exchange
DDEI06KEA156NWDB	DDEI06KEA156NWDB	2023-08-21	2023-08-21	Bank's Re
DDSI03KEA156NWDB	DDSI03KEA156NWDB	2023-08-21	2023-08-21	Bank Cap
HCIYISKEA066NRUG	HCIYISKEA066NRUG	2023-08-21	2023-08-21	Index of I
SLUEM1524ZSKEN	SLUEM1524ZSKEN	2023-08-21	2023-08-21	Youth Un
KENNGDPRPCPCPPPT	KENNGDPRPCPCPPPT	2023-08-21	2023-08-21	Real GDP
PPPTTLKEA618NUPN	PPPTTLKEA618NUPN	2023-08-21	2023-08-21	Purchasin
KENNGDPXORPCPPPT	KENNGDPXORPCPPPT	2023-08-21	2023-08-21	Real Non-
PLOINVKEA624NUPN	PLOINVKEA624NUPN	2023-08-21	2023-08-21	Price Leve
PLICPPKEA670NRUG	PLICPPKEA670NRUG	2023-08-21	2023-08-21	Price Leve
PATENT4NKEUTILITY	PATENT4NKEUTILITY	2023-08-21	2023-08-21	U.S. Gran
CIPPPGKEA156NUPN	CIPPPGKEA156NUPN	2023-08-21	2023-08-21	Investmen
PLGDPOKEA670NRUG	PLGDPOKEA670NRUG	2023-08-21	2023-08-21	Price leve
RGDPLPKEA625NUPN	RGDPLPKEA625NUPN	2023-08-21	2023-08-21	Purchasin
DDAI01KEA642NWDB	DDAI01KEA642NWDB	2023-08-21	2023-08-21	Number o
DDEI05KEA156NWDB	DDEI05KEA156NWDB	2023-08-21	2023-08-21	Bank's Re
KENEREERIX	KENEREERIX	2023-08-21	2023-08-21	Real Effec
DDSI01KEA645NWDB	DDSI01KEA645NWDB	2023-08-21	2023-08-21	Bank Z-Sc
SLEMPTOTLSPZSKEN	SLEMPTOTLSPZSKEN	2023-08-21	2023-08-21	Employme
KENNGDPRPCPPPT	KENNGDPRPCPPPT	2023-08-21	2023-08-21	Real Gros
MKTGNIKEA646NWDB	MKTGNIKEA646NWDB	2023-08-21	2023-08-21	Gross Nat
DDSI06KEA156NWDB	DDSI06KEA156NWDB	2023-08-21	2023-08-21	Liquid As
FPCPITOTLZGKEN	FPCPITOTLZGKEN	2023-08-21	2023-08-21	Inflation,
MKTGDPKEA646NWDB	MKTGDPKEA646NWDB	2023-08-21	2023-08-21	Gross Dor
WUIKEN	WUIKEN	2023-08-21	2023-08-21	World Un
SIPOVGINIKEN	SIPOVGINIKEN	2023-08-21	2023-08-21	GINI Inde
RTFPNAKEA632NRUG	RTFPNAKEA632NRUG	2023-08-21	2023-08-21	Total Fact
KENGGXG01GDPPT	KENGGXG01GDPPT	2023-08-21	2023-08-21	Total Exp
KENFCNODMFNUM	KENFCNODMFNUM	2023-08-21	2023-08-21	Use of Fir
DDDI03KEA156NWDB	DDDI03KEA156NWDB	2023-08-21	2023-08-21	Non-Bank
DDSM01KEA066NWDB	DDSM01KEA066NWDB	2023-08-21	2023-08-21	Volatility
DDOE02KEA086NWDB	DDOE02KEA086NWDB	2023-08-21	2023-08-21	Consumer
DDOE01KEA086NWDB	DDOE01KEA086NWDB	2023-08-21	2023-08-21	Consumer
DDDM01KEA156NWDB	DDDM01KEA156NWDB	2023-08-21	2023-08-21	Stock Mar
NYGDPPCAPKDKEN	NYGDPPCAPKDKEN	2023-08-21	2023-08-21	Constant
RKNANPKEA666NRUG	RKNANPKEA666NRUG	2023-08-21	2023-08-21	Capital St
PCAGDPKEA646NWDB	PCAGDPKEA646NWDB	2023-08-21	2023-08-21	Gross Dor
DDSI02KEA156NWDB	DDSI02KEA156NWDB	2023-08-21	2023-08-21	Bank Non
SEADTLITRZSKEN	SEADTLITRZSKEN	2023-08-21	2023-08-21	Literacy I
RGDPNAKEA666NRUG	RGDPNAKEA666NRUG	2023-08-21	2023-08-21	Real GDP
DDOI02KEA156NWDB	DDOI02KEA156NWDB	2023-08-21	2023-08-21	Bank Dep
KENENEERIX	KENENEERIX	2023-08-21	2023-08-21	Nominal I
KGPPPGKEA156NUPN	KGPPPGKEA156NUPN	2023-08-21	2023-08-21	Governme
DDSI04KEA156NWDB	DDSI04KEA156NWDB	2023-08-21	2023-08-21	Bank Cre
KENFCBODCKNUM	KENFCBODCKNUM	2023-08-21	2023-08-21	Geograph
MNKENA052SCEN	MNKENA052SCEN	2023-08-21	2023-08-21	Value of I
KENFCSODMFXDC	KENFCSODMFXDC	2023-08-21	2023-08-21	Use of Fir
KENDCGDPPT	KENDCGDPPT	2023-08-21	2023-08-21	Official E

We see that we have 365 variables (the rows in the search output) available for Kenya. It is now up to us to choose what we desire to analyse.

Now let us get the data on the net issues of international debt securities by the government of Kenya. This is the last item in our search list. In downloading the data, we provide python with the serach id which is `IDSOFAMRINIKE` in this case.

```{python}
kenya_debt = fred.get_series(series_id="IDSOFAMRINIKE")

kenya_debt.shape
```

```
(112,)
```

Let us peek into the first few rows of the series.

```{python}
kenya_debt.head()
```

	0
1987-01-01	NaN
1987-04-01	NaN
1987-07-01	NaN
1987-10-01	NaN
1988-01-01	NaN

Let us also see the last few rows of the series.

```{python}
kenya_debt.tail()
```

	0
2013-10-01	NaN
2014-01-01	NaN
2014-04-01	NaN
2014-07-01	NaN
2014-10-01	NaN

Let us see the null values in the data.

```{python}
kenya_debt.isna().sum()
```

```
109
```

There appears to be 109 null values in the data out of the possible 112 values. We are out of luck here.

As an alternative, lt us look at the Kenya shilling-US Dollar exchange rate.

```{python}
exchange = fred.get_series('FXRATEKEA618NUPN')
```

We examine the first and last rows of the series. It appears we have some reasonable data here from 1950 to 2010.

```{python}
exchange.head()
```

	0
1950-01-01	7.140861
1951-01-01	7.140861
1952-01-01	7.140861
1953-01-01	7.140861
1954-01-01	7.140861

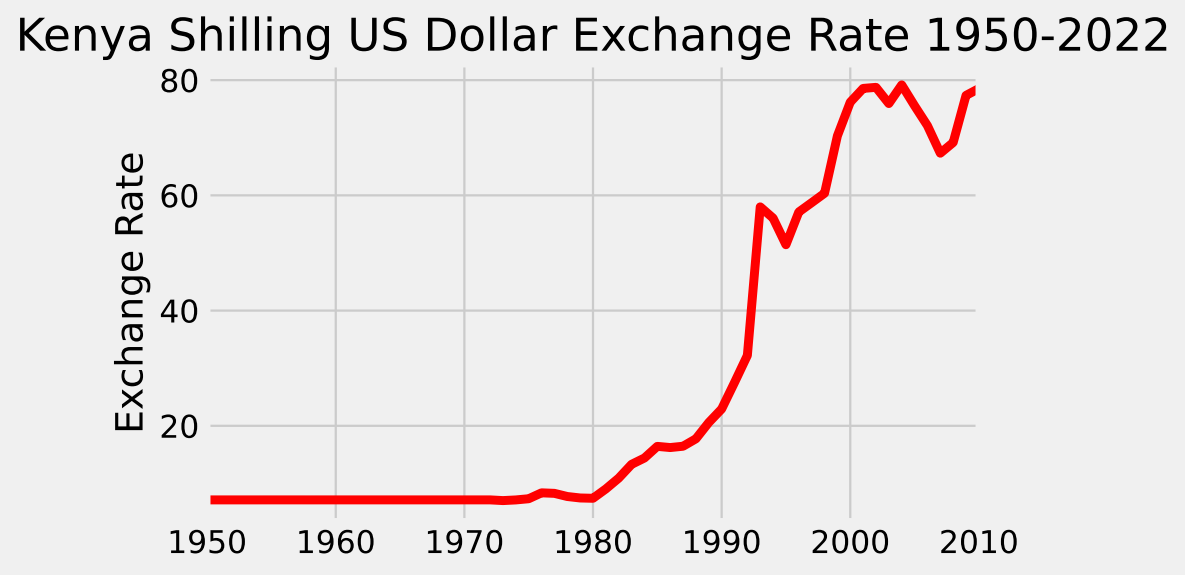
```{python}
exchange.tail()
```

	0
2006-01-01	72.100835
2007-01-01	67.317638
2008-01-01	69.175320
2009-01-01	77.352012
2010-01-01	78.539167

Analysis

```{python}
exchange.plot(title = "Kenya Shilling US Dollar Exchange Rate 1950-2022", ylabel = "Exchan
```

```
<AxesSubplot:title={'center':'Kenya Shilling US Dollar Exchange Rate 1950-2022'}, ylabel='Ex
```



Exploration

In-depth analysis

Conclusion

References