Exchange Rate Prediction

by Evrim Akgul

Sunday, October 18, 2020

MSDS 692 - Data Science Practicum I
Christy Pearson

I. About the Project

As a student of Regis University, MSc. Data Science program, I was obligated to get two practica classes and turn in a project for each of the classes. I am coming from a background of Economics and Finance. For my first practicum class, considering my backgroud as well, I picked a project something that I can contribute from the finance point of view. This way, I am hoping that I can provide some basic information regarding the field besides the other technical information of the data science aspect, as well.

II. About Exchange Rates and Time Series

In finance, an exchange rate is a measure between two currencies. At this rate one currency is exchanged for another. It is also regarded as the value of one country's currency in terms of another currency. Exchange rates are determined in the foreign exchange market, which is open to buyers and sellers, and where currency trading is continuous. From this perspective, Exchange Rates are financial time-series, but at the same time a major role player for economic decisions due to its importance and effect on all sorts of economic activities. Thus, exchange rates forecasting has a substantial role in the economic decision making processes. Accurate estimation of the rate has significant influence on successful decisions.

In terms of modelling forecast methods, it is essential to understand features of exchange rates in order to develop decent models. First and foremost, exchange rates are sequenced data. The transactions executed one after another and with a timestamp, hence they produce time-series data. Besides the sequenced nature of them, their other notable feature is that they are nonlinear and nonstationary, meaning they are nondirectional and ever changing without presenting any regularity.

In terms of modelling time-series data, the literature suggests some statistical (or econometric) modelling methods such as (S)ARIMA, ETS (for univariate series), or VAR (for multivariate series) to be employed. In general, machine learning and more precisely deep learning applications are highly successful to map irregular data. Regarding this phenomenon, this project intends to compare some Deep Learning methods

with literature suggested econometric methods. In this project, Box-Jenkins' ARIMA methodology will be used as a baseline model (acquired from the time-series forecasting literature), while MLP and CNN techniques will be getting employed for Deep Learning models. Lastly, I have chosen GBP/USD (British Sterling / US Dollar) rate to represent a financial time-series to this project.

As a note, I should add this as well. Machine Learning methodologies offer Recurrent Neural Networks (RNN) and Long-Short Term Memory (LSTM) approaches for sequence to squence data analysis as well. Yet, with the given time constraints, I had to exclude these methods. I am hoping to further analyse them as a continuation study to this project in the near future.

III. Problem Statement and Difficulties

I have mentioned this above already, yet it would be a good idea to reiterate some of it and emphasize a few different points. Exchange Rates are very important components of economy. Not only for macroeconomics, as being one of the major indicator, it is also very important in the micro level for the constituents of the economic activities such as international traders or even for local farmers due to the fact that it can be a cost factor to their businesses such as oil prices or logistic costs. Empowering correct expectation to the actors of economy, forecasting exchange rates plays a significant role. AS the dictator of the monetary policies, central banks are the main employers of these forecasting models. While technology keeps developing, probing and testing for alternative methods and exploring new possibilities is a good excercise for the betterment of the current methods.

That being said, I have to admit that I had a great deal of hardship to define this problem statement. The main reason to that was the discrepancy among the time constraint, my desires and expectation from this project and the requirements of the practicum. My understanding is that a business related project that can contribute to a solution regarding a general real world problem or a spesific problem of a business was what was expected from me. Yet, my desire was working on an academic or research topic matter rather than what was expected from me. At this point, I made a mistake and started to work on developing models without clarifying my problem statement. I guess it was around 6th week when I realized my mistake because I was creating models, testing hyperparameters, producing results etc, however, none of them were a material to relate something. With the support of my instructor, I managed to narrow down my focus and eventually came up with the above statement.

IV. Milestones of the Project

1. Data Collection

In order to conduct my project, I had to collect some data for my project. I started with the primary data: GBP/USD exchange rate. I was looking for the daily exchange rate data for this project. My first, collection of series was from "https://www.investing.com/". The data was coming with the **Date, Price, Open, Hight, Low, Volume and Change(%)** structure. Later on my project, I have discarded this data and collect a new dataset from FED, including the other **22 exchange rate parities** (which I used for the multivariate analysis models) along with the **GBP/USD parity**.

```
Forex (daily -  AUD, EUR, NZD, GBP, BRL, CAD, CNY, DKK, HKD, INR, JPY, MYR,
MXN, NOK, ZAR, SGD, KRW, LKR, SEK, CHF, TWD, THB, VEB.), from
https://www.federalreserve.gov/, from 2000-01-03 to 2020-08-21.
```

My second data source was Federal Reserve Bank of St.Louis. I have collected **libor (interest) rates** and **normalized GDP** data (both for US and UK, data downloaded separately) from here. Normalized GDP (monthly), from https://fred.stlouisfed.org/ from 2000-01-01 to 2020-05-01.

```
Libor Rates (daily), from https://fred.stlouisfed.org/, from 2001-01-02 to
2020-09-18.
```

Lastly, I acquired current account to GDP data from OECD's website. Current Account to GDP (quarterly), from https://stats.oecd.org/, from Q1-2000 to Q1-2020.

All my data was collected as `CSV` files. Excel is used to delete some useless rows and columns, when I have downloaded the data first time, but all of the data cleaning and processing is done on python in `Data Preparation.ipynb` notebook. I preferred and used Juypter notebooks at every step of my project, due to the step-by-step functionality and clean cell looking.

At this phase, I organized my data in a way that I can use during my analysis. First, I organized my exchange rate data. This one needed the most cleaning among all of them and the data were provided daily.

Second, I cleaned and integrated the GDP data to Forex (stands for Foreign exchange rate) data. GDP data was monthly, thus I had to fill the missing data by interpolating the monthly data to the daily data of the Forex.

Third, the Current Account to GDP data, similarly, I needed to fill the missing data by interpolating quarterly data to Forex' daily data.

Fourth, Libor Rates were provided daily, but they were not numerical.

After cleaning and merging all these data, I saved them to a new CSV file that I named `Merged` and for further analysis, I pulled the data from this file.

2. Data Preparation

a. Foreign Exchange Rates

I started processing my set with the forex data. Parsing the dates and assigning them to the idex column, gave me the index values that are datetime object. However, the rest of the data, forex values, were read as object, whereas they were supposed to be numerical. The reason for that was missing values for holidays that are marked with **"ND"** in the dataset. So I needed to clean them first and then convert my dataset values to numerical. learning curve: droping the rows with **"ND"** values and converting the objects to numeric values.

b. GDP

I had GDP data separately for UK and US from the same source. I read the data from `CSV` files as *series* and then concatanated them to a dataframe. Merging it with the previous Forex dataframe was the next. learning curve: reading data from the file with `squeeze=True` argument, merging dataframes in accordance with the needs.

c. Current Account to GDP

I needed to get rid of the first row because I mistakenly got one extra quarter data from the source. The data was quarterly and the date information was given in **"Qx-YYYY"** format. I needed to search for some time to find a way to convert it to datetime series, but I figured it out. The rest of the data was OK. After the conversion, I merged it to the previous dataframe. learning curve: converting the quarterly data information to datatime series.

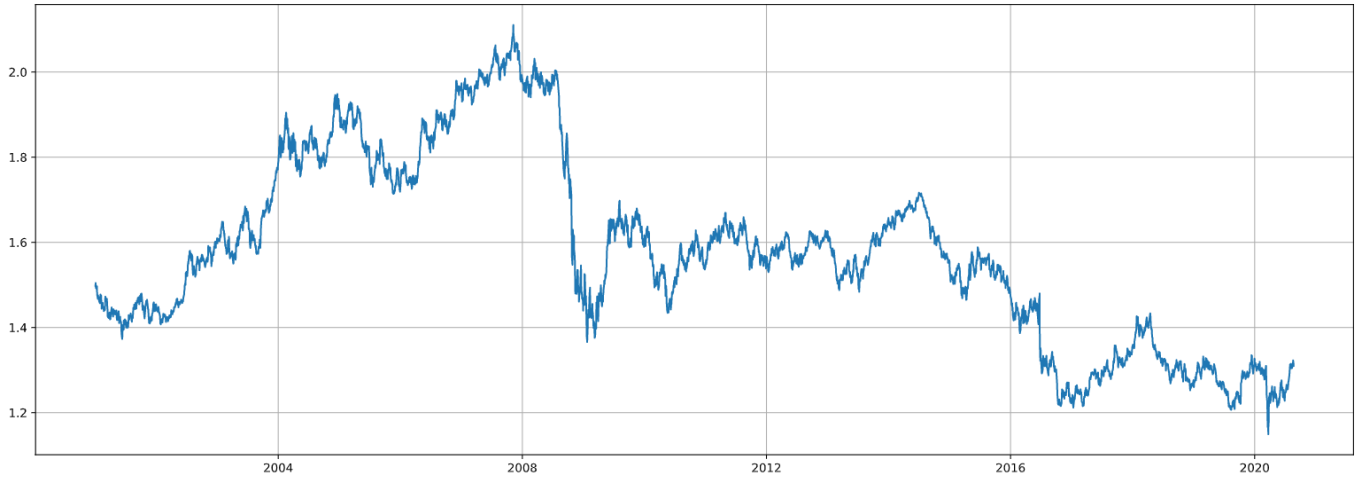
d. Libor Rates (interest RAtes for GBP and USD)

I acquired the data in two separate `CSV` file. Read the data as *series*, parsed the dates and assigned them to the index and then concatanated the *series*. There were missing data in the dataframe given in **"."**. The values were not numeric. Converted them into proper format, and merged with the previous dataframe. Due to merging data with different time frames (daily, monthly and quarterly), there were missing values. Using interpolation method, I filled them. The reason to do that was the characteristics of the financial data. For instance, a monthly GDP information is given during the daily changes of exchange rates or libor rates. Market gives reaction, whenever a new GDP information gets into the market, yet during that one month period all the valuation are made in accordance to the previous GDP data. Even after interpolation, there were still missing data and these were from the Libor interest rate data (for the entire year of 2000). I dropped these rows with `NaN` values. Thus, my ready to process data starts from the beginning of 2001.

Eventually, I had 29 different data (28 of them to be predictor variable) and 4920 observation for each of them. I saved my pre-processed data to a new `CSV` file: **"Merged.csv"**

3. Base Model (ARIMA)

My intention was creating a base model for comparison purposes, first, using the econometric ARIMA process. Thus, I read the data and extract the GBP series for univariate analysis.



EXTRA LINE:

FONT SIZE:

This is my text number1
This is my text number2
This is my text number3
This is my text number4

ALINTI:

> "Predicting the age of abalone from physical measurements. The age of abalone is determined by cutting the shell through the cone, staining it, and counting the number of rings through a microscope -- a boring and time-consuming task. Other measurements, which are easier to obtain, are used to predict the age. Further information, such as weather patterns and location (hence food availability) may be required to solve the problem." (Waugh, 1995)

HUCRE:

```
Attribute Information:

Given is the attribute name, attribute type, the measurement unit and a
brief description of each attribute.

Name / Data Type / Measurement Unit / Description
----------------------------
1. Sex / nominal / -- / M, F, and I (infant)
2. Length / continuous / mm / Longest shell measurement
3. Diameter / continuous / mm / perpendicular to length
4. Height / continuous / mm / with meat in shell
5. Whole weight / continuous / grams / whole abalone
6. Shucked weight / continuous / grams / weight of meat
7. Viscera weight / continuous / grams / gut weight (after bleeding)
8. Shell weight / continuous / grams / after being dried
9. Rings / integer / -- / +1.5 gives the age in years
```

Abalone is a shellfish and this dataset has above mentioned attributes of these shellfishes. The purpose of this study is predicting the age of Abalone shellfishes. The generic way of finding the age of an Abalone

shellfish is time-consuming. Their age is determined like a tree by counting the rings formed in the shell and this is not an easy task. I will try to use some machine learning methods to see if it is possible to find the age of an Abalone shellfish only by depending on the attributes given above. In the given data set, age representing data attribute is given with the name of `Rings`. There are 28 levels of this attribute. To calculate the age of an Abalone, the ring level of it should be summed with 1.5, because that is usually takes that long to form the first ring by an Abalone shellfish. I will be using this attribute to create my labels (response variable). Also, as I am suggested for this case study, I will be grouping ages in three category: `"Young", "Adult" and "Old"`. My age classification will be as fallows:

Young (age < 9), Adult (9<=age<=12), Old (age >12)

```
Objective: to predict the age in years of abalone shells (rings) using
physical measurements such as length diameter, whole weight, etc.
```

II. Data Preparation

a. Libraries

I first start with the libraries that I used in this case study. I give them right at the top, however, I also give them (commented out) in the chunks where they needed very first time. And to be more spesific, `e1071` package is used for SVM modelling, `class` package is used for kNN modelling, `caret` package is used to demonstrate confusinMatrix and `nnet`package is used to model NN structures.

```
library(e1071)
library(caret)
library(nnet)
library(class)
library(klaR)
```

b. Reading the Data

First thing that I have done was downloading the data from the source and reading it to RStudio. Before doing that, I have downloaded the csv file in my computer and looked at it, hence I thought it would be better to read the data as stated below with *no header*, yet I did not touch the class features of the attributes.

```
link <- "https://archive.ics.uci.edu/ml/machine-learning-
databases/abalone/abalone.data"
abalone.raw <- read.csv(link, header = F)
```

c. First Look

When I look my data after reading it, I notice that there are 9 attributes as it is explained by the source of the data. When I checked my data first thing caught to my eyes was that the data had no headers. Luckily, the source had provided that information in their "names" file and I set them as they are given in the file. I also notice that we have 4177 observations for each attribute. When I checked for the missing data, there is also none, as informed in the documantation.

```
names(abalone.raw) <- c("Sex", "Length", "Diameter", "Height", "Whole",
                        "Shucked", "Viscera", "Shell", "Rings")

str(abalone.raw)
```

d. Pre-Processing

So after naming my columns, next, I started to pre-processing step. My data has `Rings` attribute in it, though there is no `Age` information given in the data set. Relying on the information given about the data, I have created `Age` attribute myself and made it a part of the dataframe. But before doing that, I have created a copy of the raw data which I was planning to process on a bit.

```
abalone <- abalone.raw

abalone$Age <- abalone$Rings+1.5
```

After creating `Age` attribute by adding **1.5** to `Rings` attribute, I have aggregated the attribute content into three factor group: ***"Young","Adult","Old"***.