ANN DESPAIN
5075
Computer Project Completed in R

This report will examine the exchange rate of Euros to US Dollars during the period of April 30, 2015 through November 30, 2015.

1: Import and Smooth the data
Here, I imported the data and renamed the second column, "x" from its given title, "Spot exchange rate, Euro into US $." I also reformatted the date data.

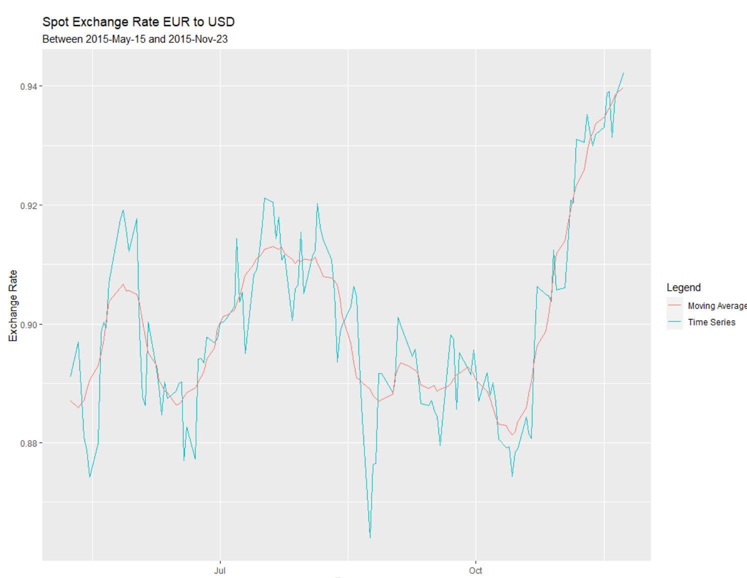
```
BankData <-read.csv(file.choose(),header=T)
names(BankData)[2] <- "x"
BankData$date2 <- dmy(BankData$date)
```

To smooth the data, I used a rolling mean function, as suggested. To compensate for the first five and last five entries that do not have 11 observations to sum, I created a new data field, BankData2 for $\tilde{x}_i$ that did not include those entries.

```
x_tilde <- rollmean(BankData$x , k=11, fill=0)
BankData$x_tilde <- x_tilde
BankData2 <- BankData[c(6:145),c(2:4)]
```

Here, we see the original data and the smoothed data together.

```
ggplot(boe_spot2) +
geom_line(aes(date2,x,color="Time Series")) +
geom_line(aes(date2,x_tilde,
             color="Moving Average")) +

 labs(title = "Spot Exchange Rate EUR to USD",
 subtitle = "Between 2015-May-15 and 2015-Nov-23",
       y = "Exchange Rate",
       color = "Legend",
 x = "Date")
```



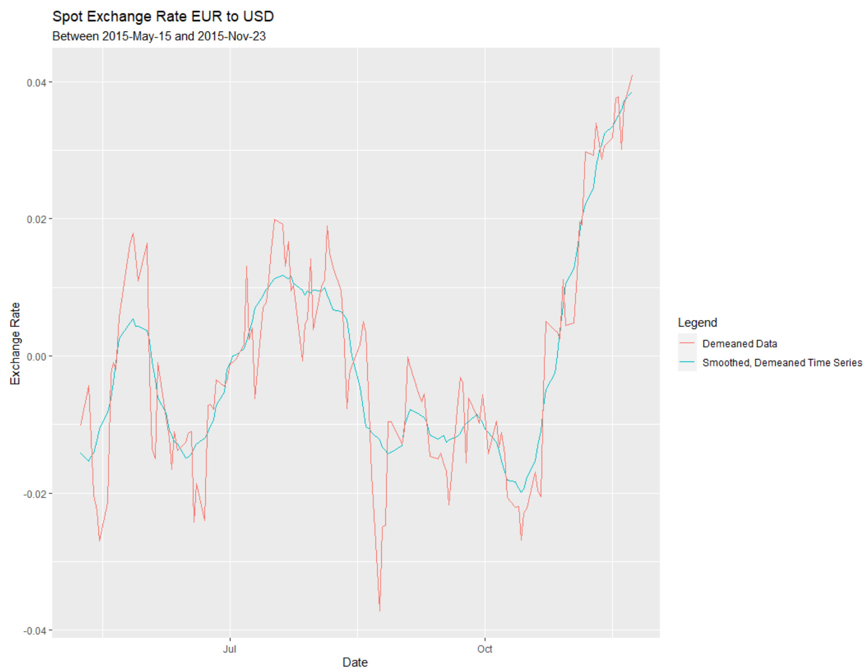
The smoothed moving average shows the general path of the original data without the extremes. We see that there is an increasing trend with the exception of three sudden larger drops in rates in June, August, and October. We see that the data is generally centered around .90, and bounded quite well ± 0.02.

2: Compute the Sample Mean and Demean Data

```
> x_bar
[1] 0.9012507
```

We find this is consistent with where the data is centered around 0.90 in the image above. Now, we define $\hat{x}_i = x_i - \bar{x}_n$. Essentially, we are subtracting the mean from our data so that it will be centered around zero. Here is the potted series of the demeaned data, $\hat{x}_i$:

```
x_bar <- mean(BankData$x)
x_hat <- (BankData$x-x_bar)
x_tilde2 <- rollmean(x_hat , k=11, fill=0)

BankData3 <- data.frame("date" <- BankData$date2,
     "x_hat" <- x_hat,
     "x_tilde2" <- x_tilde2)

names(BankData3)[3] <- "x_tilde2"
names(BankData3)[1] <- "date"
names(BankData3)[2] <- "x_hat"

BankData3 <- BankData3[c(6:145),c(1:3)]

ggplot(BankData3) +

 geom_line(aes(date,x_tilde2,color="Demeaned Data")) +
  geom_line(aes(date,x_hat,
         color="Smoothed, Demeaned Time Series")) +

labs(title = "Demeaned Spot Exchange Rate EUR to USD",
    subtitle = "Between 2015-May-15 and 2015-Nov-23",
        y = "Exchange Rate",
  color = "Legend",
  x = "Date")
```



We see that the data is now centered around zero.

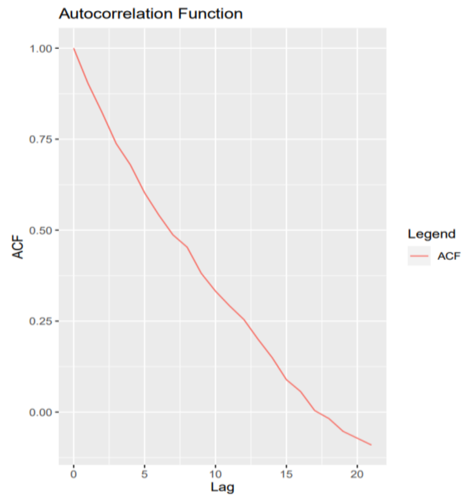
3: Plot ACF and PACF

We now use the demeaned data to plot the Auto Correlation Function (ACF) and the Partial Auto Correlation Function (PACF).

```
ACF <- acf(x_hat,plot=TRUE)
PACF <- pacf(x_hat,plot=TRUE)

ACF_df <- data.frame("acf_lag" <- ACF$lag,
                "acf" <- ACF$acf)
names(ACF_df)[1] <- "lag"
names(ACF_df)[2] <- "acf"

PACF_df <- data.frame("pacf_lag" <- PACF$lag,
                "pacf" <- PACF$acf)

names(PACF_df)[1] <- "lag"
names(PACF_df)[2] <- "pacf"
```

The ACF describes how well the present value of the series is related with its past values.

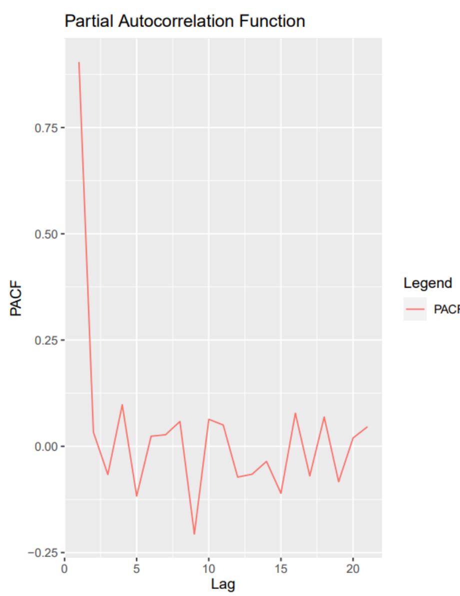
```
ggplot(ACF_df) +
 geom_line(aes(lag,acf,color="ACF")) +

 labs(title = "Autocorrelation Function",
    y = "ACF",
       color = "Legend",
       x = "Lag")
```



We see that with zero lag, the value is one, as expected, with a linear decrease afterward. This would suggest that the present values are more relevant than the past values. Note that Lag 1 = 0.904.

Taking a look at the PACF:

```
ggplot(PACF_df) +
geom_line(aes(lag,pacf,color="PACF")) +

 labs(title = "Partial Autocorrelation Function",
       y = "PACF",
      color = "Legend",
      x = "Lag")
```



The PACF does not consider the data that lies within our error bands, so instead of finding correlations of the present with lags like the ACF, it finds correlation of the residuals. Therefore, if there is any hidden information in the residual data which can be modeled by the next lag, we might get a good correlation and we would keep that next lag as a feature while modeling. As we have a high initial value (PACF lag 1 = 0.9041, the same as the lag 1 value for ACF) followed by a drastic drop, we would assume that the relevant information is found not with past residuals where the values of the PACF are close to zero, but with the more current information. This would suggest that an AR(1) would be a good fit.

4: Fit an AR(1) model to $\hat{x}_i$ and compute residuals and s^2 as defined by the project.

To fit an AR(1), we use an ARMA (p,q) model in R, recalling that we use the ACF and PACF pair to help identify the orders p and q. The ACF tails off for AR(p) and cuts off after a certain lag MA(q) while the PACF cuts off the AR(p) after lag p and tails off an MA (q).

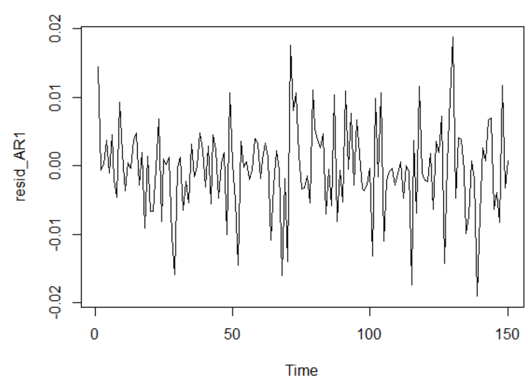
$$\hat{\epsilon}_i = \hat{x}_i - \hat{\varphi}\hat{x}_{i-1}$$

$$s_1^2 = \frac{1}{n-2}\sum_{i=2}^{n} \hat{\epsilon}_i^2$$

```
AR1_fit <- arima(x_hat,order = c(1,0,0))
AR1_fit
```

```
Call:
arima(x = x_hat, order = c(1, 0, 0))

Coefficients:
         ar1  intercept
      0.9407     0.0032
s.e.  0.0294     0.0083

sigma^2 estimated as 4.321e-05:
```

```
resid_AR1 <- AR1_fit$resid
plot(resid_AR1)
```

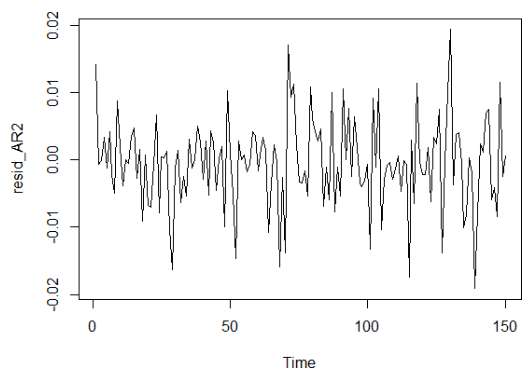


Similarly, we fit an AR(2) model to $\hat{x}_i$ and compute s^2.

```
AR2_fit <- arima(x_hat,order = c(2,0,0))
AR2_fit
```

```
Call:
arima(x = x_hat, order = c(2, 0, 0))

Coefficients:
         ar1     ar2  intercept
      0.8904  0.0548     0.0035
s.e.  0.0812  0.0827     0.0089

sigma^2 estimated as 4.308e-05:
```

```
resid_AR2 <- AR2_fit$resid
plot(resid_AR2)
```

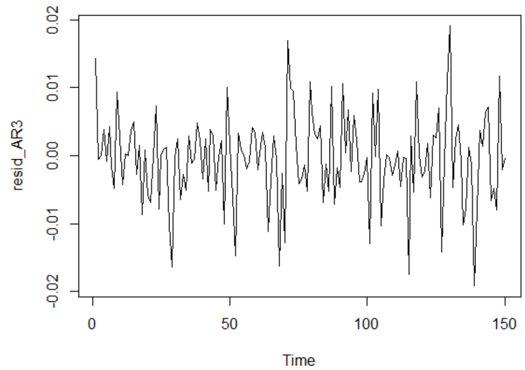


Finally, we fit an AR(3) model to $\hat{x}_i$ and compute s^2.

```
AR3_fit <- arima(x_hat,order = c(3,0,0))
AR3_fit
```

```
Call:
arima(x = x_hat, order = c(3, 0, 0))

Coefficients:
         ar1     ar2      ar3  intercept
      0.8951  0.1219  -0.0779     0.0032
s.e.  0.0811  0.1088   0.0824     0.0082

sigma^2 estimated as 4.283e-05:
```

```
resid_AR3 <- AR3_fit$resid
plot(resid_AR3)
```



There is not any detectable visible difference in the graphs of the residual data. The error bands for all three seem to be $\pm$ 0.01. Analyzing the coefficients we received by fitting the data to the AR(1), AR(2), and AR(3) models, along with what we know from the ACF and PACF, the AR(1) model appears to be the best fit. With coefficients from the AR(1) of 0.9407, 0.8901, 0.8954 from the three models respectively, we see these values are very close to our ACF and PACF lag 1 values of .904; and are also very near our true mean of 0.9012.

```
Note:

R libraries required to run the code:
require(zoo)
require(tidyverse)
require(hrbrthemes)
require(lubridate)
```