ISYE 6402 Homework 2 Q1 Solutions

Background

In this problem, we will study fluctuations in currency exchange rate over time.

File `EURUSDCurrency.csv` download contains the daily exchange rate of USD/EUR from January 1999 through December 31st 2020. We will aggregate the data on a weekly basis, by taking the average rate within each week. The time series of interest is the weekly currency exchange. We will analyze this time series and its first order difference.

Instructions on reading the data

To read the data in `R`, save the file in your working directory (make sure you have changed the directory if different from the R working directory) and read the data using the `R` function `read.csv()`

```
fpath <- "EURUSDCurrency.csv"
df <- read.csv(fpath, head = TRUE)
```

Here we upload the libraries needed the this data analysis:

```
library(mgcv)
library(lubridate)
library(dplyr)
```

To prepare the data, run the following code snippet. First, aggregate by week:

```
df$date <- as.Date(df$Date, format='%m/%d/%Y')
df$week <- floor_date(df$date, "week")
df$eur <- df$EUR

df <- df[, c("week", "eur")]
```

We now form the weekly aggregated time series to use for data exploration! Please note that we will analyze the weekly aggregated data not the original (daily) data.

```
agg <- aggregate(x = df$eur, by = list(df$week), FUN = mean)
colnames(agg) <- c("week", "eur")

price <- ts(agg$eur, start = 1999, freq = 52)
```

Please use the `price` series to code and answer the following questions.

Question 1a: Exploratory Data Analysis

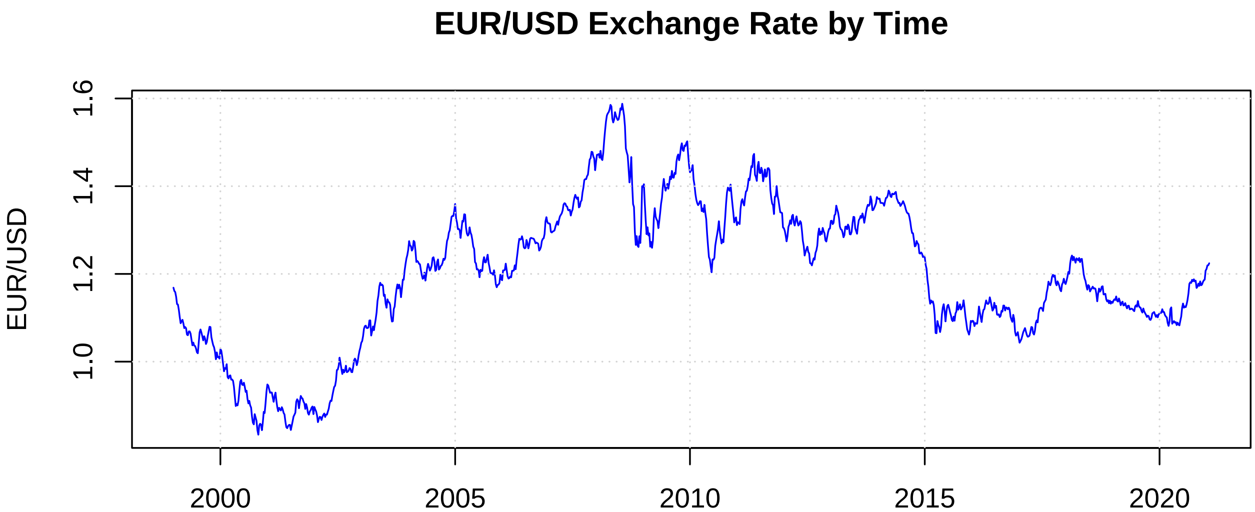
Before exploring the data, can you infer the data features from what you know about the USD-EUR currency exchange? Next plot the Time Series and ACF plots of the weekly data. Comment on the main features, and identify what (if any) assumptions of stationarity are violated.

Which type of model do you think will fit the data better: the trend or seasonality fitting model? Provide details for your response.

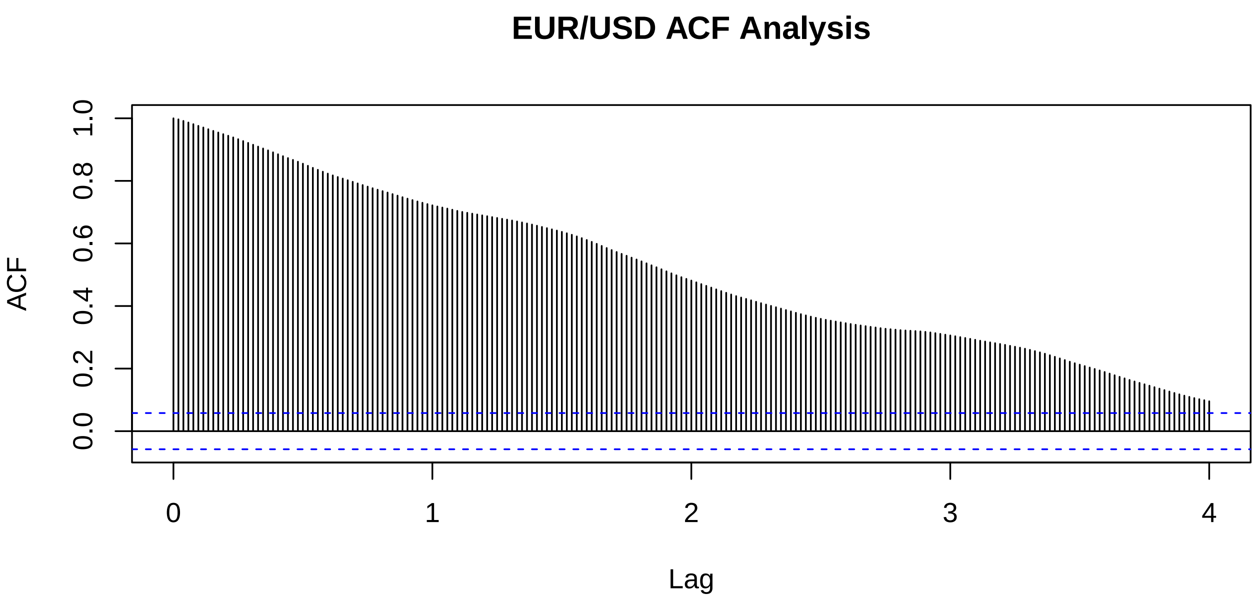
Response: General Insights on the USD-EUR Currency Rate

The time series of a currency rate would generally follow a trend, depending on the trade policies and international relations of the two countries. For example, the currency rate of some countries in the last 100 years has been varying significantly compared to others. Recall the currency rate debate around the USD vs Yuan (Chinese currency), for example. The Indian currency vs USD has weakened considerably in the past 100 years and this would clearly follow a downward trend. For the EUR-USD currency exchange, the EURO economy vs the US economy has changed patterns; while overall we expect an upward trend, it would not be very steep. There also would be periods of trend reversal.

```
ts.plot(price, col = "blue", xlab = "", ylab = "EUR/USD",
        main = "EUR/USD Exchange Rate by Time")
grid()
```




```
acf(price, lag.max = 52 * 4, xlab = "Lag", ylab = "ACF", main = "EUR/USD ACF Analysis")
```



Response: General Insights from the Graphical Analysis

From th time series plot, we see that the variance varies significantly within the time window. The trend also has some variation in certain time periods. We can say that the variability depends on time for this time series. From the ACF plot, we can see that the autocorrelation is significant but slowly decreasing for all lag periods.

From the two plots, we can clearly say that the trend is clearly present, but no seasonality is observed. Hence, a trend fitting model would be a better fit than a seasonality model for this time series.

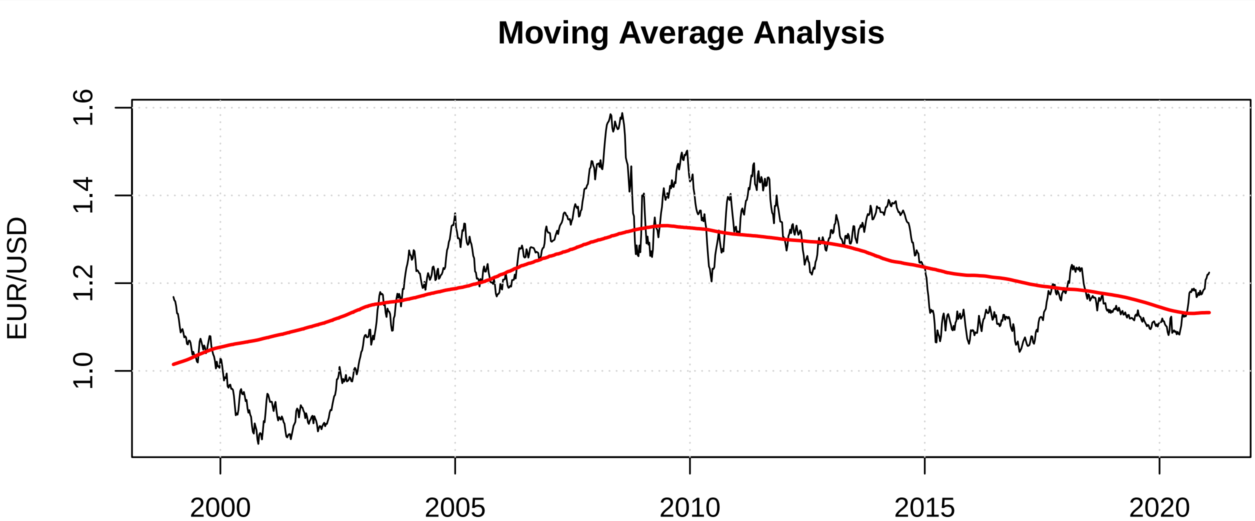
Question 1b: Trend Estimation

Fit the following trend estimation models:

- Moving Average
- Parametric Quadratic Polynomial
- Local Polynomial
- Splines Smoothing

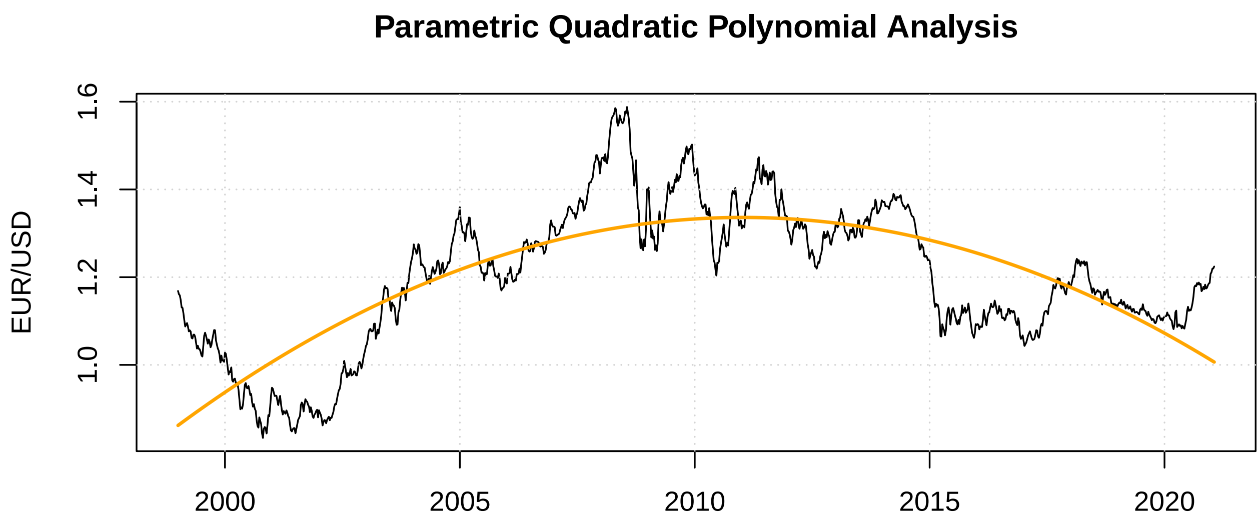
Overlay the fitted values on the original time series. Plot the residuals with respect to time for each model. Plot the ACF of the residuals for each model also. Comment on the four models fit and on the appropriateness of the stationarity assumption of the residuals.

```
# convert X axis to 0-1 scale
points <- 1:length(price)
points <- (points - min(points)) / max(points)

# 1. Fit a moving average model
mav.model <- ksmooth(points, price, kernel = "box")
mav.fit <- ts(mav.model$y, start = 1999, frequency = 52)

ts.plot(price, xlab = "", ylab = "EUR/USD", main = "Moving Average Analysis")
grid()
lines(mav.fit, lwd = 2, col = "red")
```

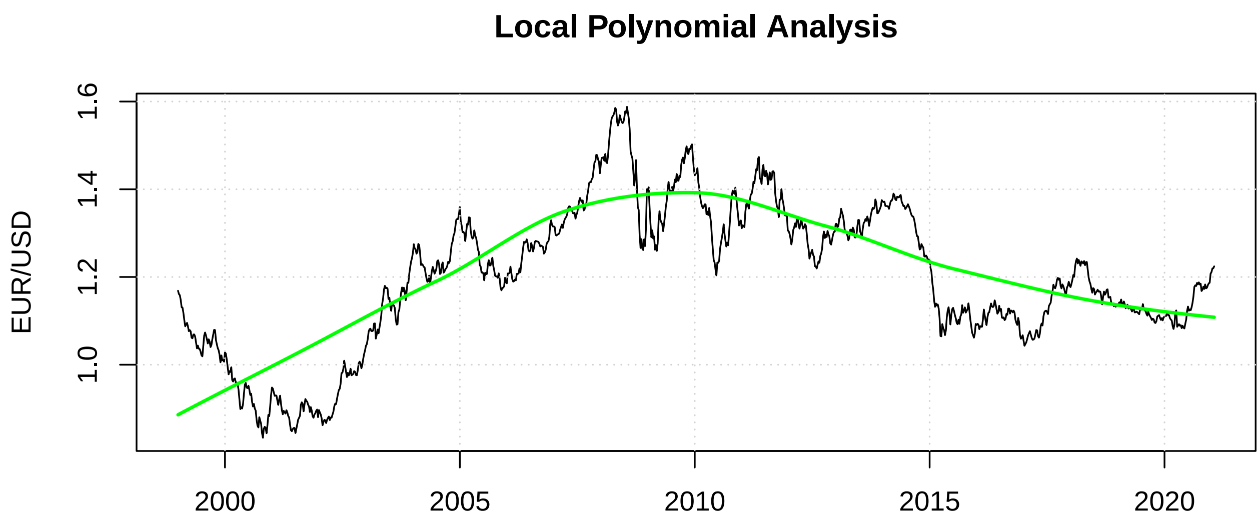



```
# 2. Fit a parametric quadratic polynomial model
x1 <- points
x2 <- points ^ 2
para.model <- lm(price ~ x1 + x2)
para.fit <- ts(fitted(para.model), start = 1999, frequency = 52)
```

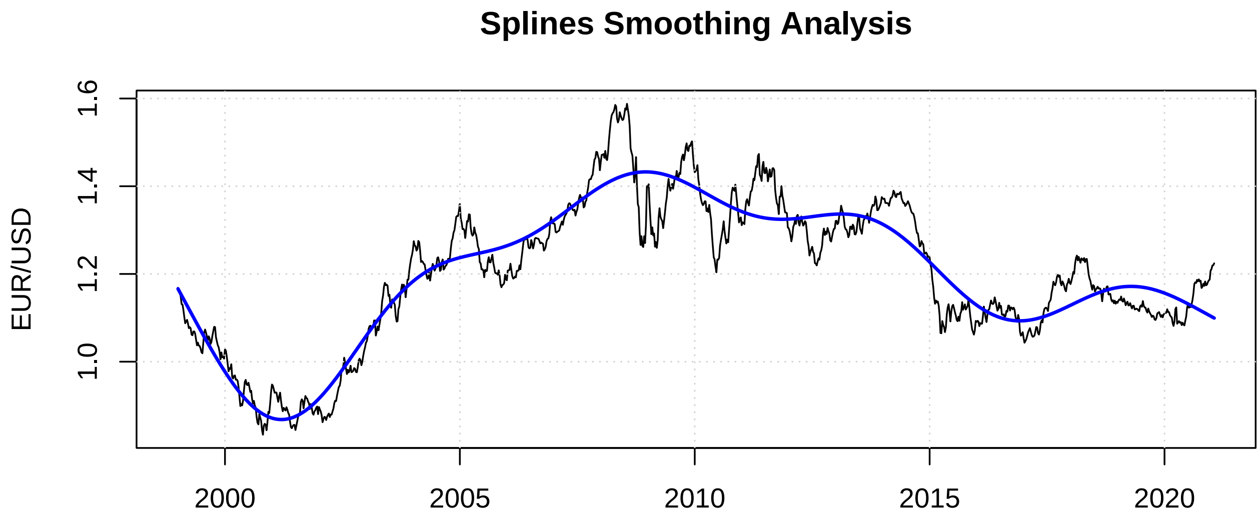
```
ts.plot(price, xlab = "", ylab = "EUR/USD", main = "Parametric Quadratic Polynomial Analysis")
grid()
lines(para.fit, lwd = 2,col = "orange")
```



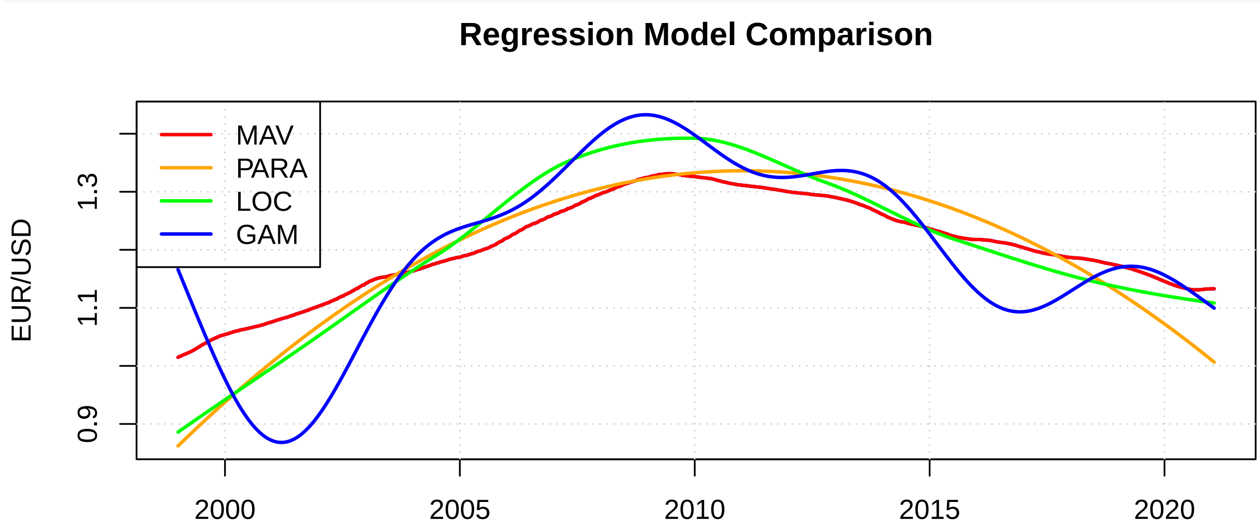

```
# 3. Fit a local polynomial model
loc.model <- loess(price ~ points)
loc.fit <- ts(fitted(loc.model), start = 1999, frequency = 52)

ts.plot(price, xlab = "", ylab = "EUR/USD", main = "Local Polynomial Analysis")
grid()
lines(loc.fit, lwd = 2,col = "green")
```




```
# 4. Fit a splines smoothing model
gam.model <- gam(price ~ s(points))
gam.fit <- ts(fitted(gam.model), start = 1999, frequency = 52)

ts.plot(price, xlab = "", ylab = "EUR/USD", main = "Splines Smoothing Analysis")
grid()
lines(gam.fit, lwd = 2,col = "blue")
```



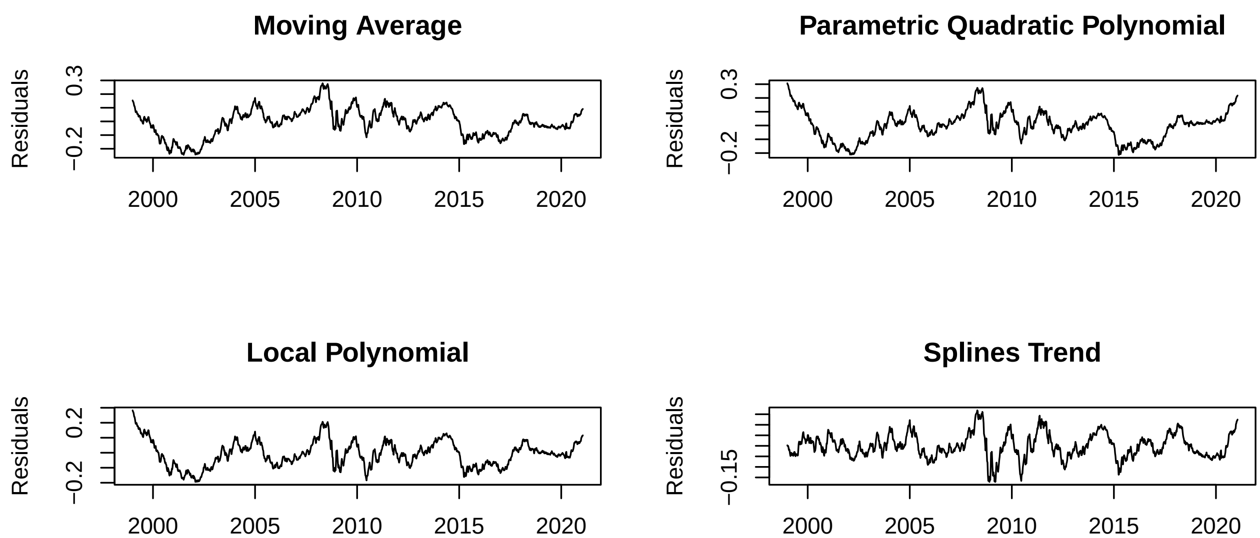

```r
# 5. Compare all estimated trends

vals <- c(mav.fit, para.fit, loc.fit, gam.fit)
ylim <- c(min(vals), max(vals))

ts.plot(mav.fit, lwd = 2, col = "blue", ylim = ylim,
        xlab = "", ylab = "EUR/USD",
        main = "Regression Model Comparison")
grid()
lines(mav.fit, lwd = 2, col = "red")
lines(para.fit, lwd = 2, col = "orange")
lines(loc.fit, lwd = 2, col = "green")
lines(gam.fit, lwd = 2, col = "blue")
legend("topleft", legend = c("MAV", "PARA", "LOC", "GAM"),
       col = c("red", "orange", "green", "blue"), lwd = 2)
```



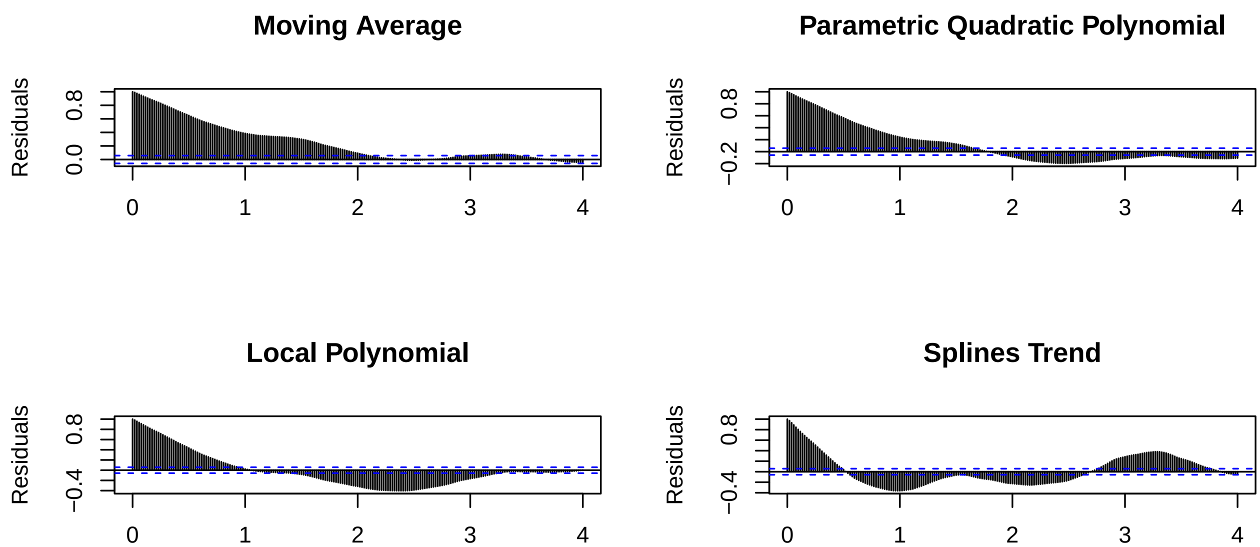
Response: Comparison of the fitted trend models:

Moving average fit is quite wiggly, hence it doesn't provide a good 'feel' for the fit. Splines smoothing shows much more complexity, although it seems to capture the overall trend. While the parametric polynomial model captues the inverted 'U' trend, the local polynomial model adds a little more pattern from the actual time series. (Please note that these observations may vary from one student to another.)

```r
# Residual and Residual ACF plots of the residuals from the fitted models
diff.mav <- price - mav.fit
diff.para <- price - para.fit
diff.loc <- price - loc.fit
diff.gam <- price - gam.fit

par(mfrow = c(2, 2))
ts.plot(diff.mav, xlab = "", ylab = "Residuals", main = "Moving Average")
ts.plot(diff.para, xlab = "", ylab = "Residuals", main = "Parametric Quadratic Polynomial")
ts.plot(diff.loc, xlab = "", ylab = "Residuals", main = "Local Polynomial")
ts.plot(diff.gam, xlab = "", ylab = "Residuals", main = "Splines Trend")
```




```r
par(mfrow = c(2, 2))
acf(diff.mav, lag.max = 52 * 4, xlab = "", ylab = "Residuals", main = "Moving Average")
acf(diff.para, lag.max = 52 * 4, xlab = "", ylab = "Residuals",
    main = "Parametric Quadratic Polynomial")
acf(diff.loc, lag.max = 52 * 4, xlab = "", ylab = "Residuals", main = "Local Polynomial")
acf(diff.gam, lag.max = 52 * 4, xlab = "", ylab = "Residuals", main = "Splines Trend")
```



Response: Appropriateness of the trend model for stationarity

The residuals from the trend models are provided above. They show clear non-stationarity, suggesting that trend removal alone using any of the three models is not sufficient for accounting for non stationary variations in the time series.

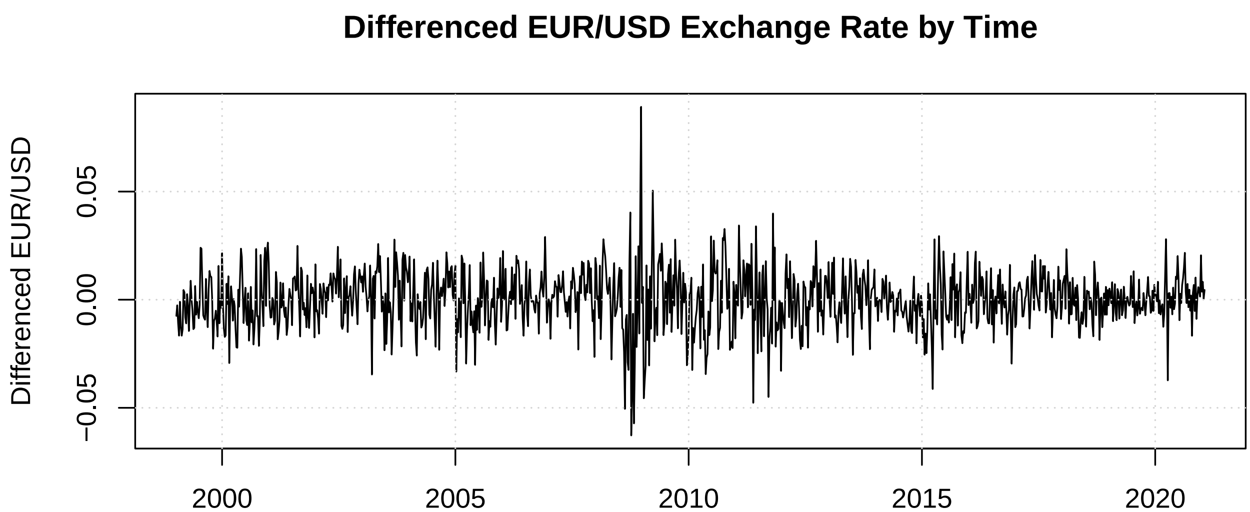
The ACFs of the residuals also supports this observation of non stationarity. In the ACF of the splines fit there is some cyclicity observed but that does not seem to be pointing to seasonality. Same goes with the parametric and local polynomial model residuals (albeit lesser cyclicity than the splines fit) and all residual process ACFs show a trend with the decreasing magnitudes.

Question 1c: Differenced Data Modeling

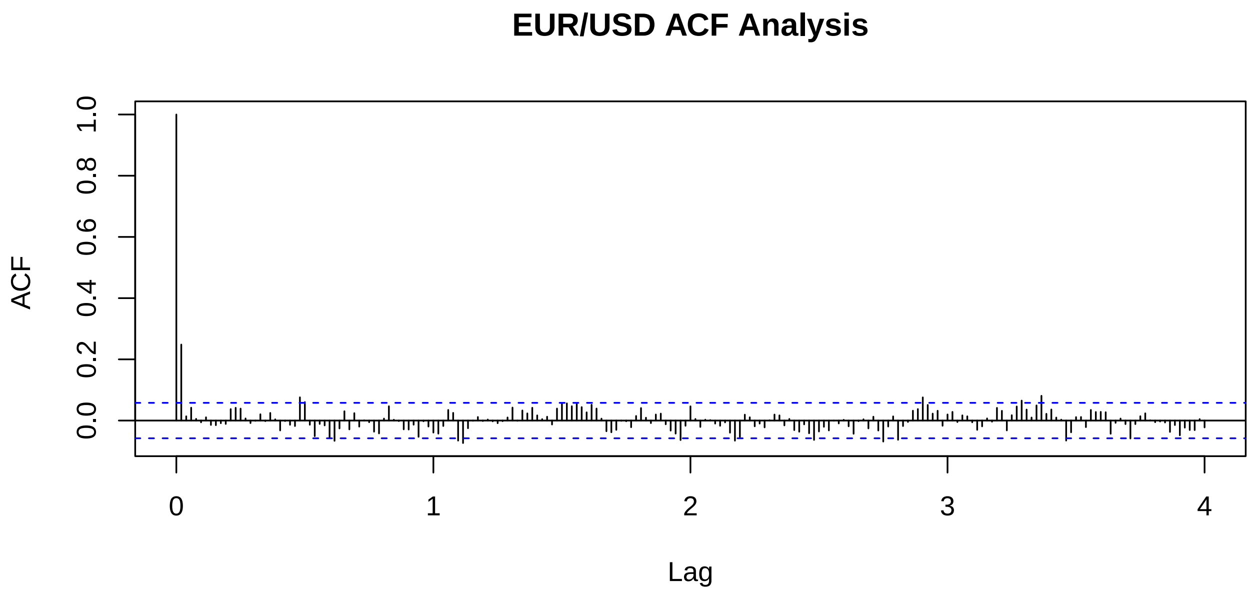
Now plot the difference time series and its ACF plot. Apply the four trend models in Question 1b to the differenced time series. What can you conclude about the difference data in terms of stationarity? Which model would you recommend to apply (trend removal via fitting trend vs differencing) such that to obtain a stationary process?

Hint: When TS data are differenced, the resulting dataset will have an NA in the first data element due to the differencing.

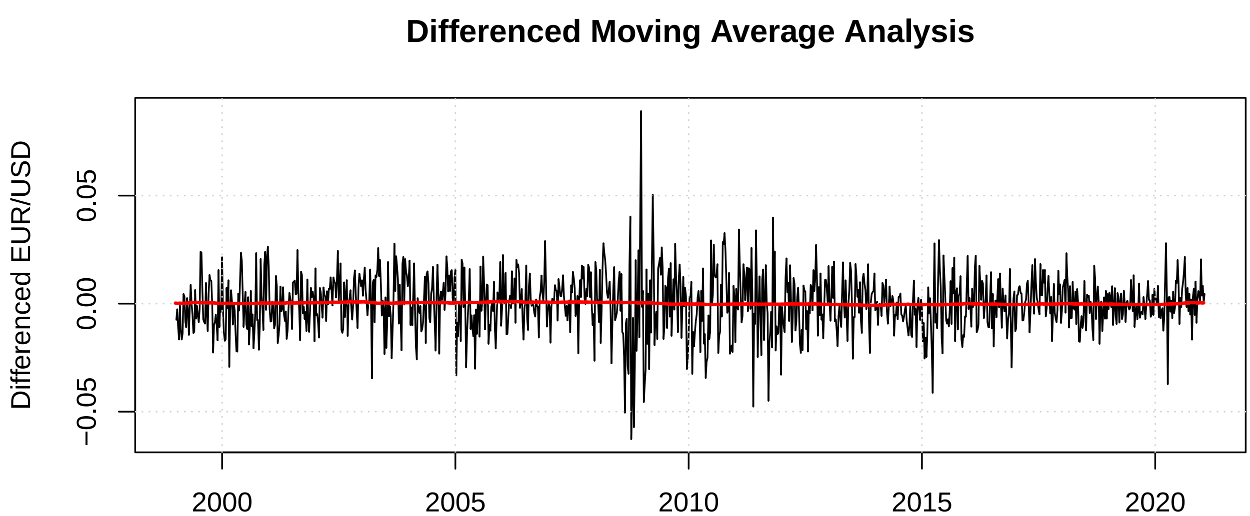
```
ts.plot(diff(price), col = "black", xlab = "", ylab = "Differenced EUR/USD",
        main = "Differenced EUR/USD Exchange Rate by Time")
grid()
```




```
acf(diff(price), lag.max = 52 * 4, xlab = "Lag", ylab = "ACF ", main = "EUR/USD ACF Analysis")
```

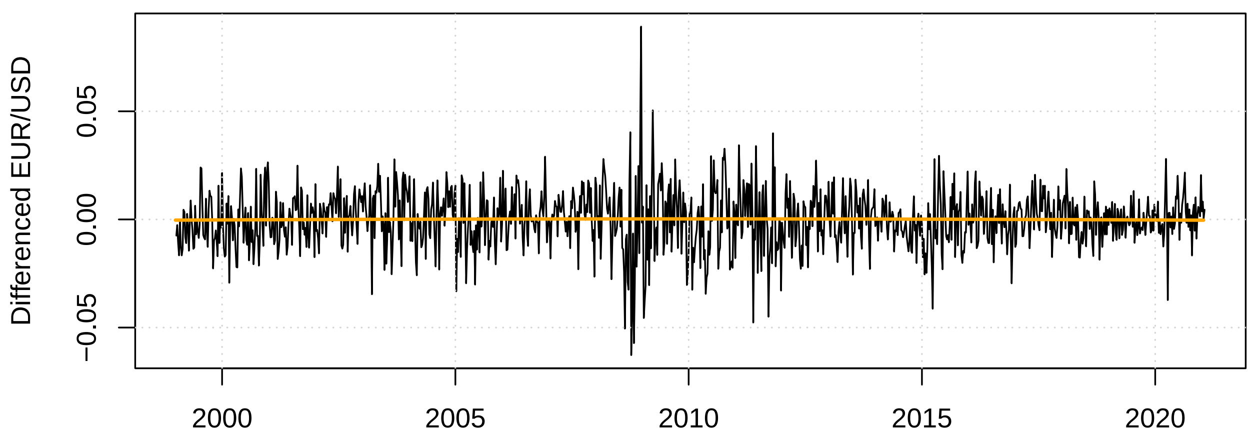



```r
# 1. Fit a moving average model
mav.model <- ksmooth(points[-1], diff(price), kernel = "box")
mav.fit <- ts(mav.model$y, start = 1999, frequency = 52)

ts.plot(diff(price), xlab = "", ylab = "Differenced EUR/USD",
        main = "Differenced Moving Average Analysis")
grid()
lines(mav.fit, lwd = 2, col = "red")
```



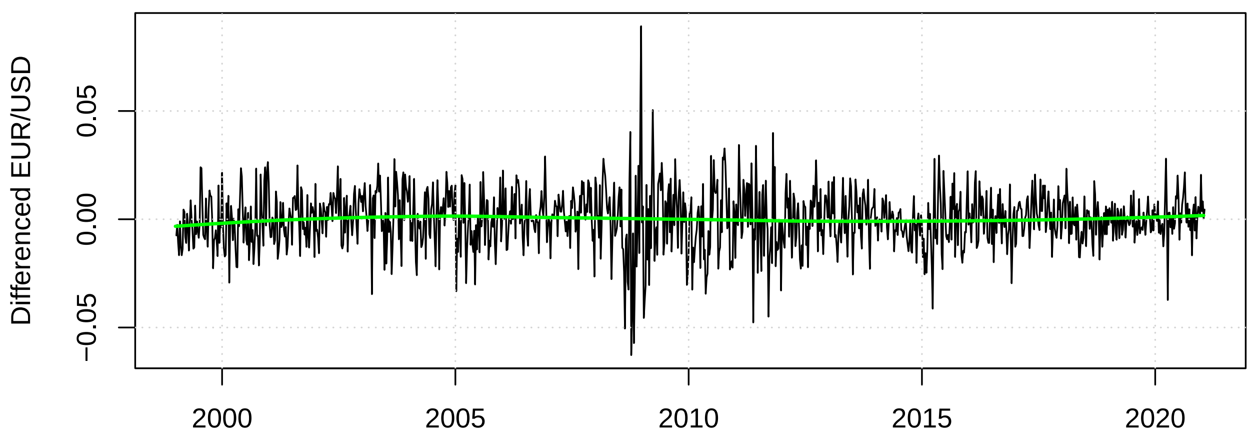

```r
# 2. Fit a parametric quadratic polynomial model
x1 <- points[-1]
x2 <- points[-1] ^ 2
para.model <- lm(diff(price) ~ x1 + x2)
para.fit <- ts(fitted(para.model), start = 1999, frequency = 52)

ts.plot(diff(price), xlab = "", ylab = "Differenced EUR/USD",
        main = "Differenced Parametric Quadratic Polynomial Analysis")
grid()
lines(para.fit, lwd = 2,col = "orange")
```

Differenced Parametric Quadratic Polynomial Analysis

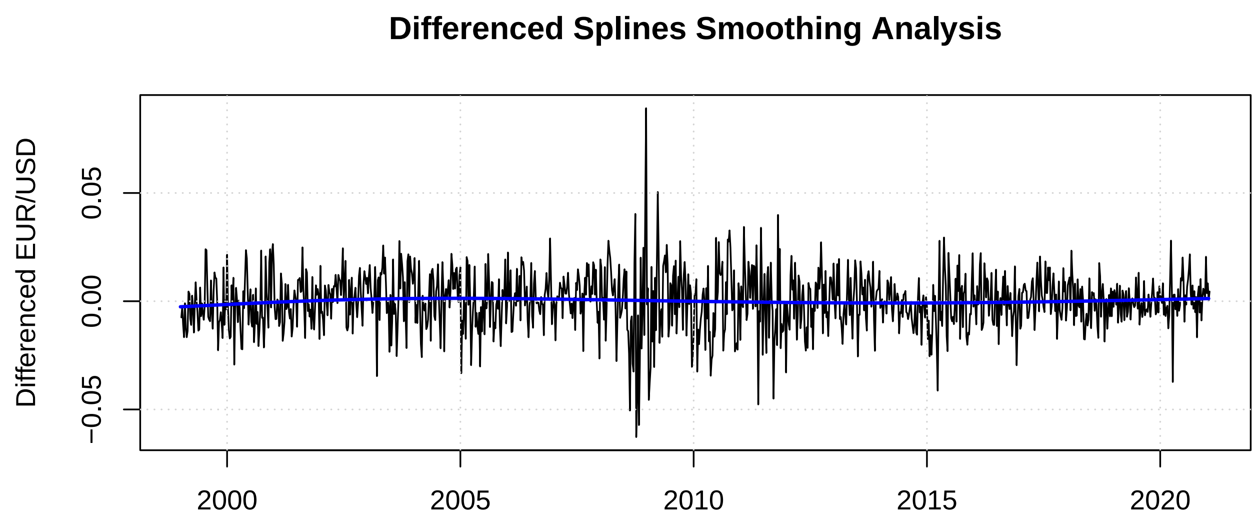



```r
# 3. Fit a local polynomial model
loc.model <- loess(diff(price) ~ points[-1])
loc.fit <- ts(fitted(loc.model), start = 1999, frequency = 52)

ts.plot(diff(price), xlab = "", ylab = "Differenced EUR/USD",
        main = "Differenced Local Polynomial Analysis")
grid()
lines(loc.fit, lwd = 2,col = "green")
```

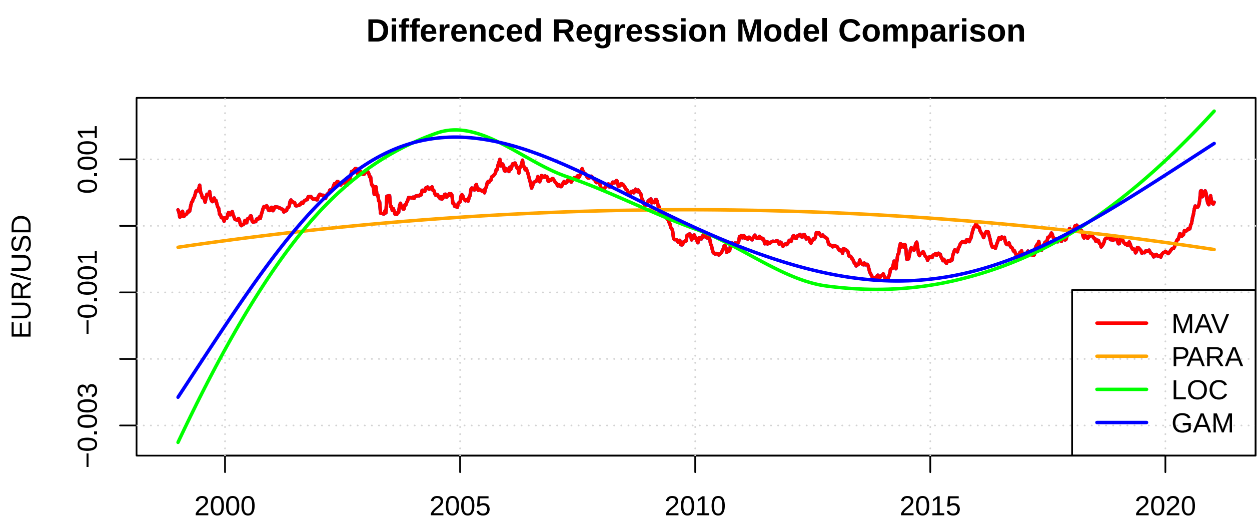
Differenced Local Polynomial Analysis




```r
# 4. Fit a splines smoothing model
gam.model <- gam(diff(price) ~ s(points[-1]))
gam.fit <- ts(fitted(gam.model), start = 1999, frequency = 52)

ts.plot(diff(price), xlab = "", ylab = "Differenced EUR/USD",
        main = "Differenced Splines Smoothing Analysis")
grid()
lines(gam.fit, lwd = 2,col = "blue")
```




```r
# 5. Compare all estimated trends

vals <- c(mav.fit, para.fit, loc.fit, gam.fit)
ylim <- c(min(vals), max(vals))

ts.plot(mav.fit, lwd = 2, col = "blue", ylim = ylim,
        xlab = "", ylab = "EUR/USD",
        main = "Differenced Regression Model Comparison")
grid()
lines(mav.fit, lwd = 2, col = "red")
lines(para.fit, lwd = 2, col = "orange")
lines(loc.fit, lwd = 2, col = "green")
lines(gam.fit, lwd = 2, col = "blue")
legend("bottomright", legend = c("MAV", "PARA", "LOC", "GAM"),
       col = c("red", "orange", "green", "blue"), lwd = 2)
```



Response: Comments about the stationarity of the difference data:

The time series plots seem to clearly show the appropriateness of fit of the models and the indication of the stationarity.

The fitted line showing the moving average trend seems to have the least variability as compared to the constant line. The parametric quadratic model also has little variability but not as much as the splines model which has higher deviation in trend, and local polynomial model which has the highest deviation as shown in the combined graph. The moving average trend model, however, has many ‘kinks’ that capture the minor movements that might not be of use in determining the trend!

From this analysis, we can confirm the property of stationarity hence using the difference data is more appropriate approach to removing the trend such that the time series becomes stationary. I would recommend the moving average model because it has the least variability. (Note: any reasonable recommendation is acceptable.)