FA542 - Homework #3

I pledge my honor that I have abided by the Stevens Honor System.

Sid Bhatia

2023-10-17

Problem 1

Consider the weekly yields of Moody's Aaa and Baa seasoned bonds from January 5, 1962, to April 10, 2009 (*w-aaa.txt* and *w-Baa.txt* files).

The data are obtained from the Federal Reserve Bank of St Louis. Weekly yields are averages of daily yields.

```
# Load libraries for skewness, kurtosis, and plotting.
library(moments)
library(fBasics)
```

```
##
## Attaching package: 'fBasics'

## The following objects are masked from 'package:moments':
##
##     kurtosis, skewness
```

```
# Establish the directory for data.
data_directory <- "C:/Users/sbhatia2/My Drive/University/Academics/Semester V/FA542 - Time Series with A

# Load in each dataset.
moody_Aaa <- read.table(paste(data_directory, 'w-aaa.txt', sep=""))
moody_Baa <- read.table(paste(data_directory, 'w-Baa.txt', sep=""))

head(moody_Aaa)
```

```
##     V1 V2 V3   V4
## 1 1962  1  5 4.43
## 2 1962  1 12 4.42
## 3 1962  1 19 4.42
## 4 1962  1 26 4.41
## 5 1962  2  2 4.42
## 6 1962  2  9 4.42
```

```
head(moody_Baa)
```

```
##     V1 V2 V3   V4
## 1 1962  1  5 5.11
## 2 1962  1 12 5.09
## 3 1962  1 19 5.08
## 4 1962  1 26 5.08
## 5 1962  2  2 5.07
## 6 1962  2  9 5.08
```

```r
# Combine V1, V2, and V3 columns into a single date column.
moody_Aaa$Date <- as.Date(paste(moody_Aaa$V1, moody_Aaa$V2, moody_Aaa$V3, sep = "-"), format = "%Y-%m-%
moody_Baa$Date <- as.Date(paste(moody_Baa$V1, moody_Baa$V2, moody_Baa$V3, sep = "-"), format = "%Y-%m-%

# Keep only 'Date' and rename 'V4' to 'Aaa Bond Yield' and 'Baa Bond Yield' respectively.
moody_Aaa <- moody_Aaa[, c("Date", "V4")]
moody_Baa <- moody_Baa[, c("Date", "V4")]

colnames(moody_Aaa)[colnames(moody_Aaa) == "V4"] <- "Aaa Bond Yield"
colnames(moody_Baa)[colnames(moody_Baa) == "V4"] <- "Baa Bond Yield"

head(moody_Aaa)
```

```
##         Date Aaa Bond Yield
## 1 1962-01-05           4.43
## 2 1962-01-12           4.42
## 3 1962-01-19           4.42
## 4 1962-01-26           4.41
## 5 1962-02-02           4.42
## 6 1962-02-09           4.42
```

```r
head(moody_Baa)
```

```
##         Date Baa Bond Yield
## 1 1962-01-05           5.11
## 2 1962-01-12           5.09
## 3 1962-01-19           5.08
## 4 1962-01-26           5.08
## 5 1962-02-02           5.07
## 6 1962-02-09           5.08
```

a. Obtain the summary statistics (sample mean, standard deviation, skewness, excess kurtosis, minimum, and maximum) of the two yield series.

Are the bond yields skewed?

Do they have heavy tails? Answer the questions using 5% signifficance level.

```r
# Create function that calculates the sample mean, standard deviation, skewness, excess kurtosis, minim
compute_statistics <- function(returns)
{
  mean_val <- mean(returns)
  sd_val <- sd(returns)
  skewness_val <- skewness(returns)
  excess_kurtosis_val <- kurtosis(returns)
  min_val <- min(returns)
  max_val <- max(returns)

  # Create a list to hold each sample statistic.
  result <- list(
    mean = mean_val,
    standard_deviation = sd_val,
    skewness = skewness_val,
    excess_kurtosis = excess_kurtosis_val,
    minimum = min_val,
    maximum = max_val
```

```r
  )

  return(result)
}

# Compute statistics for both bond yields.
compute_statistics(moody_Aaa$`Aaa Bond Yield`)
```

```
## $mean
## [1] 7.830109
##
## $standard_deviation
## [1] 2.418744
##
## $skewness
## [1] 0.857092
## attr(,"method")
## [1] "moment"
##
## $excess_kurtosis
## [1] 0.5786054
## attr(,"method")
## [1] "excess"
##
## $minimum
## [1] 4.19
##
## $maximum
## [1] 15.85
```

```r
compute_statistics(moody_Baa$`Baa Bond Yield`)
```

```
## $mean
## [1] 8.847122
##
## $standard_deviation
## [1] 2.717073
##
## $skewness
## [1] 0.9297785
## attr(,"method")
## [1] "moment"
##
## $excess_kurtosis
## [1] 0.760896
## attr(,"method")
## [1] "excess"
##
## $minimum
## [1] 4.78
##
## $maximum
## [1] 17.29
```

We are testing the following:

$$H_0 : S(r) = 0$$

$$H_a : S(r) \neq 0$$

Where $S(r)$ is the skewness of the bond yields.

The t-statistic for sample skewness is the following:

$$t = \frac{\hat{S}(r)}{\sqrt{6/n}}$$

```
# Construct function to test whether the population skewness of yields is equal to 0 or not.
skewness_test <- function(yields)
{
  t_skewness <- skewness(yields) / sqrt(6 / length(yields))

  p_skewness <- 2 * (1 - pnorm(abs(t_skewness)))

  return(p_skewness)
}

# Conduct the skewness test for both yields.
skewness_test(moody_Aaa$`Aaa Bond Yield`)
```

```
## [1] 0
## attr(,"method")
## [1] "moment"
```

```
skewness_test(moody_Baa$`Baa Bond Yield`)
```

```
## [1] 0
## attr(,"method")
## [1] "moment"
```

Since both yields resulted in a p-value less than 0.05, we reject the null hypothesis that the (population) skewness of the bond yields are 0, and we accept the alternative hypothesis that bond yields are skewed.

We are testing the following:

$$H_0 : K = 3$$
$$H_a : K \neq 3$$

where K denotes the kurtosis of the returns (excess kurtosis = 0).

The t-statistic for kurtosis of the yields is the following:

$$t = \frac{\hat{K}(r) - 3}{\sqrt{24/n}}$$

```
# Construct function to test whether the population kurtosis of returns is equal to 3 or not (excess ku
kurtosis_test <- function(yields)
{
  # Since the kurtosis function already computes excess log returns, we do not have to subtract by thre
```

```r
  t_kurtosis <- (kurtosis(yields)) / sqrt(24 / length(yields))

  p_kurtosis <- 2 * (1 - pnorm(abs(t_kurtosis)))

  return(p_kurtosis)
}

# Conduct the kurtosis test for both yields.
kurtosis_test(moody_Aaa$`Aaa Bond Yield`)
```

```
## [1] 4.457306e-09
## attr(,"method")
## [1] "excess"
```

```r
kurtosis_test(moody_Baa$`Baa Bond Yield`)
```

```
## [1] 1.221245e-14
## attr(,"method")
## [1] "excess"
```

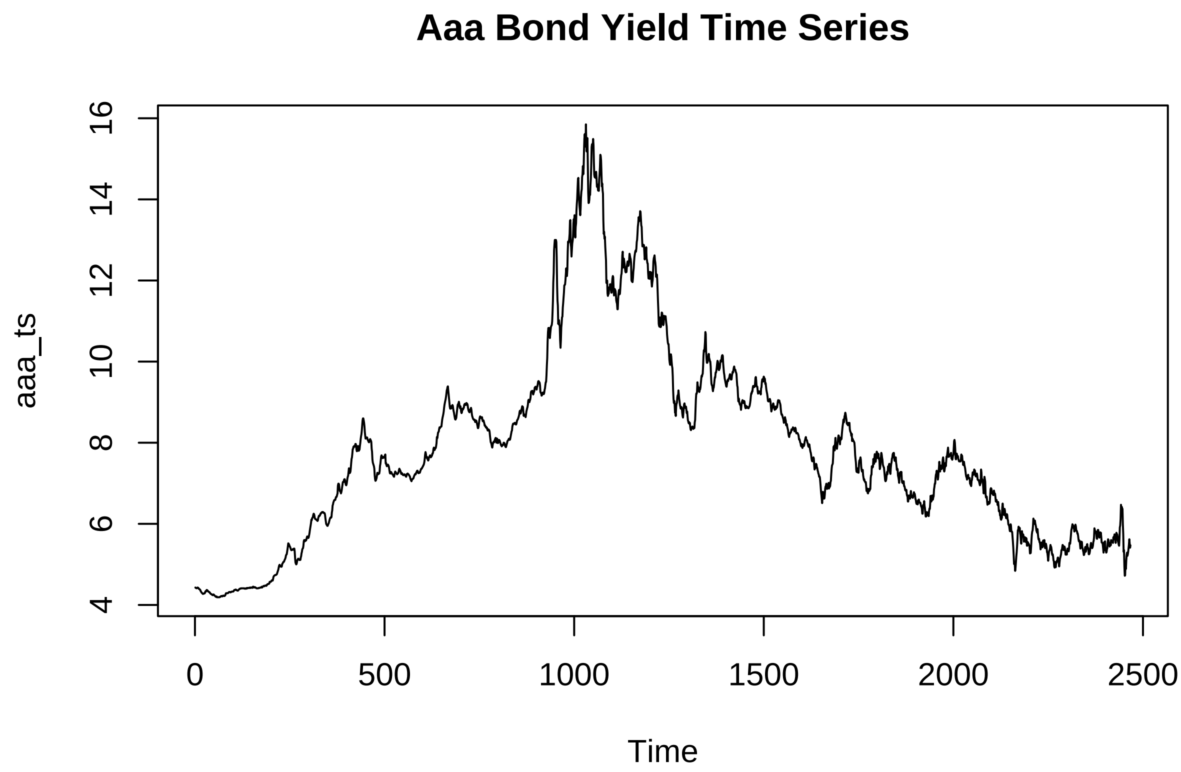
Since both yields resulted in a p-value less than 0.05, we reject the null hypothesis that the (population) kurtosis of the bond yields are 3, and we accept the alternative hypothesis that the kurtosis of the bond yields are not 3, implying heavy tails.

b. Build a time series model for the Aaa series.

```r
# Load in libraries for time series analysis.
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
# Retrieve the Aaa bond yields.
aaa_bond_yield <- moody_Aaa$`Aaa Bond Yield`

# Convert Aaa bond yields to a time series object.
aaa_ts <- ts(aaa_bond_yield)

# Plot the time series.
plot(aaa_ts, main = "Aaa Bond Yield Time Series")
```




```r
# Perform the ADF test to check for stationarity.
adf_test_result <- adf.test(aaa_ts)
adf_test_result
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  aaa_ts
## Dickey-Fuller = -1.5577, Lag order = 13, p-value = 0.7656
## alternative hypothesis: stationary
```

Since the p-value for the Augmented Dickey-Fuller (ADF) Test is greater than 0.05, we fail to reject the null hypothesis that the time series data is non-stationary. As such, we will have to difference the data and test for stationarity.

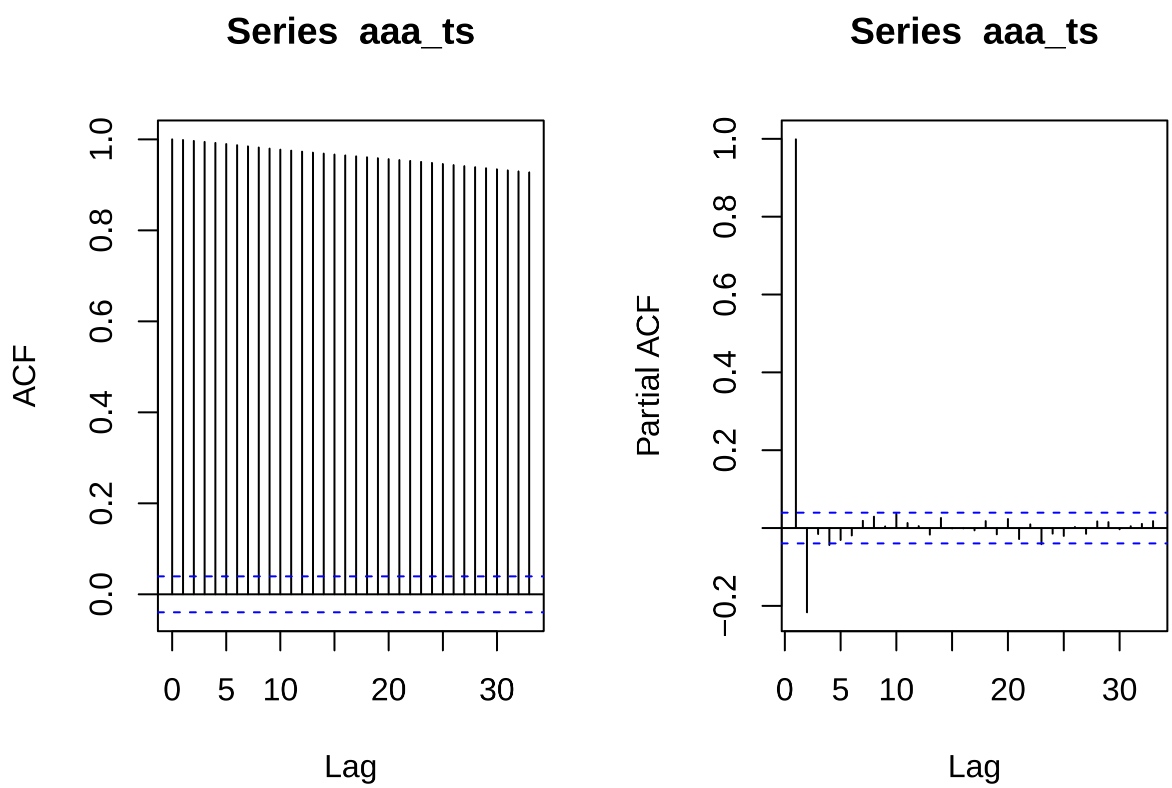
```r
# Perform differencing.
differenced_aaa_ts <- diff(aaa_ts)

adf.test(differenced_aaa_ts)
```

```
## Warning in adf.test(differenced_aaa_ts): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  differenced_aaa_ts
## Dickey-Fuller = -14.461, Lag order = 13, p-value = 0.01
## alternative hypothesis: stationary
```

Since the p-value of the ADF test is less than 0.05, we can reject the null hypothesis that the difference time series data is non-stationary and accept the alternative hypothesis that the data is stationary at the 5 significance level.

```
# Import the `forecast` library for ARIMA model.
library(forecast)

# Create a layout with one row and two columns.
par(mfrow = c(1, 2))

# Plot ACF and PACF for Aaa bond yields.
acf(aaa_ts)
pacf(aaa_ts)
```




```
# Create a function to perform grid search to find the best ARIMA based on AIC.
grid_search <- function(ts)
{
  best_model <- NULL
  best_aic <- Inf
  best_order <- c(0, 0, 0)

  for (p in 0:5)
  {
    for (d in 1:1)
    {
      for (q in 0:5)
      {
```

```r
        model <- arima(ts, order = c(p, d, q))
        aic <- AIC(model)

        if (aic < best_aic)
        {
          best_model <- model
          best_aic <- aic
          best_order <- c(p, d, q)
        }
      }
    }
  }

  # Create a result data frame with the best AIC and formatted best order.
  result_df <- data.frame(Best_AIC = best_aic, Best_Order = paste0("(", paste(best_order, collapse = ",
  return(result_df)
}

grid_search(aaa_ts)
```

```
## Warning in arima(ts, order = c(p, d, q)): possible convergence problem: optim
## gave code = 1
```

```
##    Best_AIC Best_Order
## 1 -4918.369    (4,1,5)
```

```r
# Compare model with `auto.arima`.
auto.arima(aaa_ts)
```

```
## Series: aaa_ts
## ARIMA(3,1,5)
##
## Coefficients:
##          ar1      ar2     ar3     ma1     ma2      ma3      ma4     ma5
##       0.2077  -0.7928  0.4810  0.1713  0.8112  -0.1025  -0.0699  0.0866
## s.e.  0.0993   0.0382  0.0958  0.0997  0.0725   0.1186   0.0433  0.0227
##
## sigma^2 = 0.007939:  log likelihood = 2467.37
## AIC=-4916.74   AICc=-4916.66   BIC=-4864.44
```

As such, the best model for the Aaa bond yield data is ARIMA($p = 4, d = 1, q = 5$) since the model had a lower AIC than ARIMA($p = 3, d = 1, q = 5$).

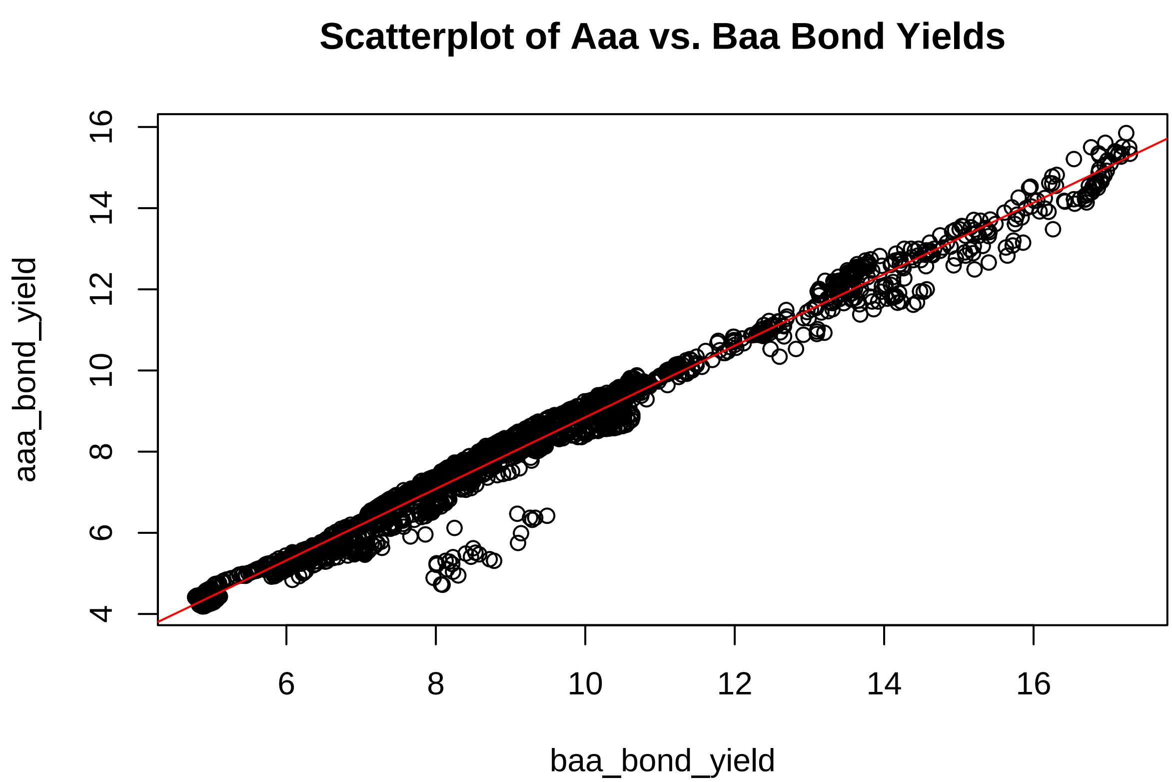
c. What is the relationship between the Aaa and Baa series?

To answer this question, build a time series model using yields of Aaa bonds as the dependent variable and yields of Baa bonds as independent variable.

```r
# Retrieve Baa bond yields.
baa_bond_yield <- moody_Baa$`Baa Bond Yield`

# Fit a linear regression model where Aaa bonds is the dependent variable and Baa is the independent va
model <- lm(aaa_bond_yield ~ baa_bond_yield)

# Get summary of the regression model.
summary(model)
```

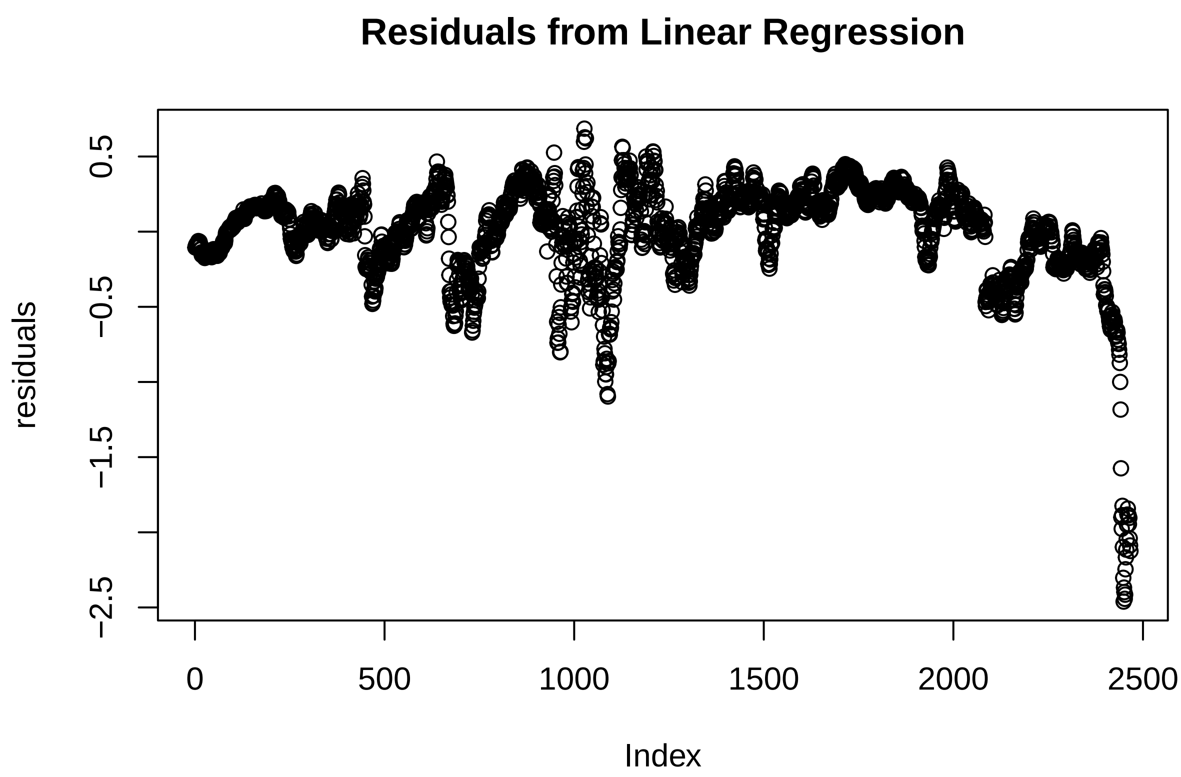
```
##
## Call:
## lm(formula = aaa_bond_yield ~ baa_bond_yield)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.46094 -0.14265  0.06717  0.20420  0.68515
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.030569   0.023025   1.328    0.184
## baa_bond_yield 0.881591   0.002488 354.350   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3357 on 2465 degrees of freedom
## Multiple R-squared:  0.9807, Adjusted R-squared:  0.9807
## F-statistic: 1.256e+05 on 1 and 2465 DF,  p-value: < 2.2e-16
```

```r
# Create a scatterplot.
plot(baa_bond_yield, aaa_bond_yield, main = "Scatterplot of Aaa vs. Baa Bond Yields")

# Add the regression line to the plot.
abline(model, col = "red")
```




```r
# Retrieve the residuals from the linear regression model.
residuals <- residuals(model)
```

```r
# Visualize the residuals.
plot(residuals, main = "Residuals from Linear Regression")
```




```r
# Check stationarity of residuals.
adf.test(residuals)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  residuals
## Dickey-Fuller = -1.7271, Lag order = 13, p-value = 0.6939
## alternative hypothesis: stationary
```

```r
# After differencing the residuals, we have stationarity.
adf.test(diff(residuals))
```

```
## Warning in adf.test(diff(residuals)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(residuals)
## Dickey-Fuller = -16.811, Lag order = 13, p-value = 0.01
## alternative hypothesis: stationary
```

```r
# Create ARIMA model for residual using `auto.arima`.
auto_residuals_model <- auto.arima(residuals)
```

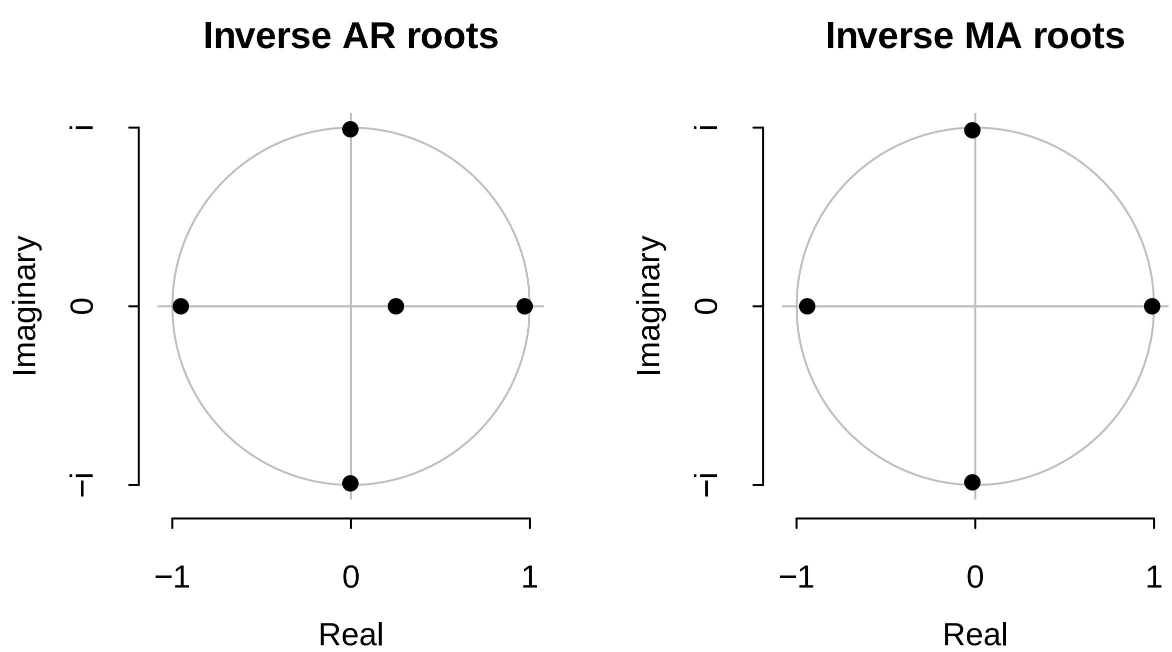
```r
# Use grid search function to find ARIMA model with lowest AIC.
grid_search(residuals)
```

```
## Warning in arima(ts, order = c(p, d, q)): possible convergence problem: optim
## gave code = 1

##    Best_AIC Best_Order
## 1 -7617.324    (5,1,4)
```

```r
best_residuals_model <- arima(residuals, order = c(5, 1, 4))
```

```
## Warning in arima(residuals, order = c(5, 1, 4)): possible convergence problem:
## optim gave code = 1
```

```r
# Diagnostic plots for the time series model.
plot(best_residuals_model)
```



As such, after creating a time series models for the residuals of the linear regression model, we find that the best model is a ARIMA$(p = 5, d = 1, q = 4)$.

According to the linear regression model, there seems to be a strong positive linear relationship between Aaa bond yields and Baa bond yields as the adjuusted $R^2 = 0.9807$ with a F-statistic being very large and corresponding p-value less than 0.01.

Problem 2

This problem is concerned with the dynamic relationship between the spot and futures prices of the S&P 500 index.

The data file *sp5may.txt* has three columns: log (futures price), log (spot price), and cost-of-carry ($\times 100$). The data were obtained from the Chicago Mercantile Exchange for the S&P 500 stock index in May 1993 and its June futures contract.

The time interval is 1 minute (intraday). Several authors used the data to study index futures arbitrage.

Here we focus on the first two columns. Let f_t and s_t be the log prices of futures and spot, respectively.

Consider $y_t = f_t - f_{t-1}$ and $x_t = s_t - s_{t-1}$.

Build a regression model with time series errors between y_t and x_t with y_t being the dependent variable.

```r
# Load in S&P500 data.
SP_500 <- read.table(paste(data_directory, 'sp5may.txt', sep=""), header = T)

head(SP_500)
```

```
##   lnfuture  lnspot     cost
## 1  6.08382 6.08618 -0.16501
## 2  6.08404 6.08623 -0.16501
## 3  6.08473 6.08630 -0.16501
## 4  6.08450 6.08630 -0.16501
## 5  6.08450 6.08623 -0.16501
## 6  6.08439 6.08625 -0.16501
```

```r
# Calculate log returns for spot and futures.
SP_500_futures_log_returns <- diff(SP_500$lnfuture)
SP_500_spot_log_returns <- diff(SP_500$lnspot)

head(SP_500_futures_log_returns)
```

```
## [1]  0.00022  0.00069 -0.00023  0.00000 -0.00011 -0.00035
```

```r
head(SP_500_spot_log_returns)
```

```
## [1]  5e-05  7e-05  0e+00 -7e-05  2e-05  3e-05
```

```r
# Build the linear regression model.
model <- lm(SP_500_futures_log_returns ~ SP_500_spot_log_returns, data = SP_500)

# Print the summary of the linear regression results.
summary(model)
```

```
##
## Call:
## lm(formula = SP_500_futures_log_returns ~ SP_500_spot_log_returns,
##     data = SP_500)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0038484 -0.0001568 -0.0000014  0.0001612  0.0026256
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             1.354e-06  3.509e-06   0.386      0.7
## SP_500_spot_log_returns 6.212e-01  1.754e-02  35.420   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.0002948 on 7058 degrees of freedom
## Multiple R-squared:  0.1509, Adjusted R-squared:  0.1508
## F-statistic:  1255 on 1 and 7058 DF,  p-value: < 2.2e-16
```

```r
# Fit an ARIMA model for the residuals of the model.
auto.arima((residuals(model)))
```

```
## Series: (residuals(model))
## ARIMA(1,0,1) with zero mean
##
## Coefficients:
##          ar1      ma1
##       0.8179  -0.9196
## s.e.  0.0152   0.0102
##
## sigma^2 = 8.429e-08:  log likelihood = 47483.26
## AIC=-94960.51   AICc=-94960.51   BIC=-94939.93
```

As such, the best model for the residuals or time series errors is $\text{ARMA}(p = 1, q = 1)$.

Problem 3

Consider the daily CDS spreads of JP Morgan from July 20, 2004 to September 19, 2014 (*d-cdsJPM.txt* file).

The period includes the financial crisis of 2008 so that the CDS spread vary substantially. The data are in the file *d-cdsJPM.txt* (column 2).

Since the spreads are small, we consider the time series $x_t = 100 \times (\text{spread})$.

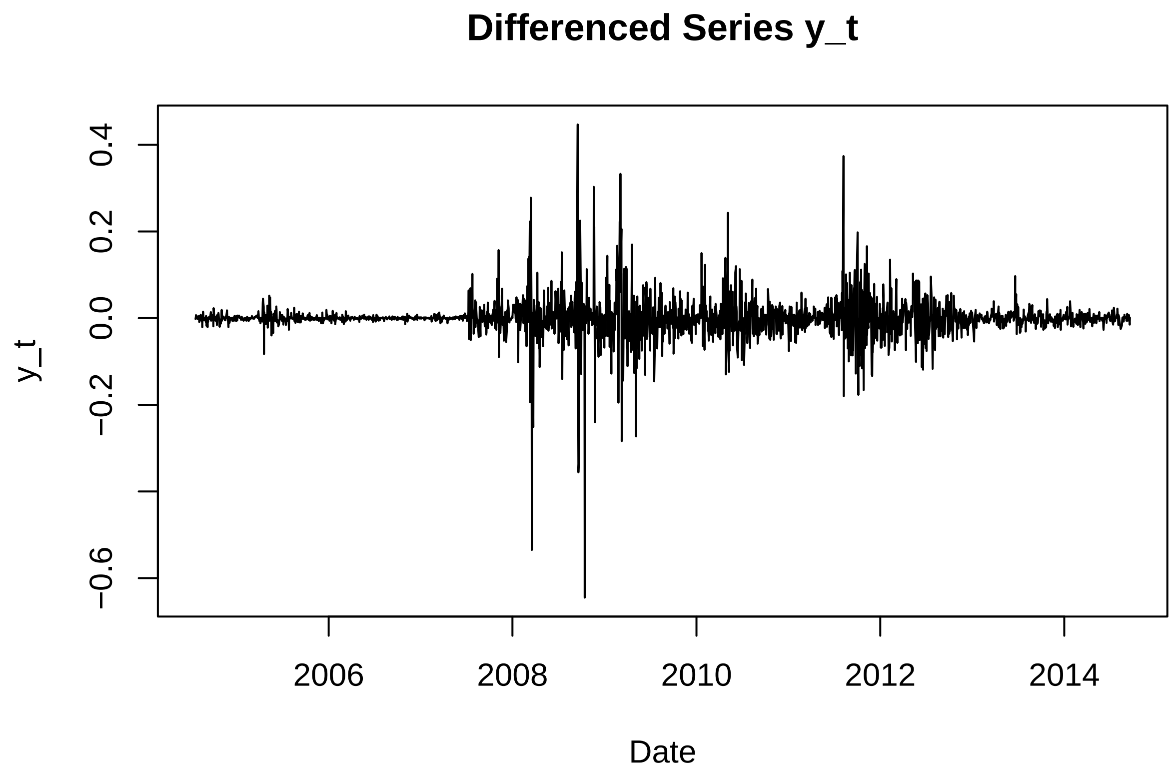
In addition, sample ACF of x_t shows strong serial dependence. Therefore, we analyze the differenced series $y_t = (1 - B)x_t$.

Build a time series model for y_t. Write down the fitted model.

```r
# Load JPM CDS Spreads.
JPM_CDS <- read.table(paste(data_directory, 'd-cdsJPM.txt', sep=""), col.names = c("Date", "JPM_CDS_Spre

# Convert the date column to "YYYYMMDD" format.
JPM_CDS$Date <- as.Date(as.character(JPM_CDS$Date), format = "%Y%m%d")

head(JPM_CDS)
```

```
##         Date JPM_CDS_Spreads
## 1 2004-07-20         0.00235
## 2 2004-07-21         0.00233
## 3 2004-07-22         0.00235
## 4 2004-07-23         0.00242
## 5 2004-07-26         0.00242
## 6 2004-07-27         0.00243
```

```r
# Calculate x_t.
x_t <- 100 * JPM_CDS$JPM_CDS_Spreads

# Calculate the differenced series y_t.
y_t <- c(NA, diff(x_t))

# Plot the differenced series y_t.
plot(JPM_CDS$Date, y_t, type = "l", xlab = "Date", ylab = "y_t", main = "Differenced Series y_t")
```




```r
auto.arima(y_t)
```

```
## Series: y_t
## ARIMA(1,0,3) with zero mean
##
## Coefficients:
##          ar1      ma1      ma2      ma3
##       0.5795  -0.4277  -0.0861  -0.1249
## s.e.  0.0855   0.0866   0.0229   0.0216
##
## sigma^2 = 0.001892:  log likelihood = 4503.17
## AIC=-8996.35   AICc=-8996.32   BIC=-8966.99
```

As such, the best model for the difference series y_t is ARMA$(p = 1, q = 3)$.