STA137 Project II

Junyao Lu

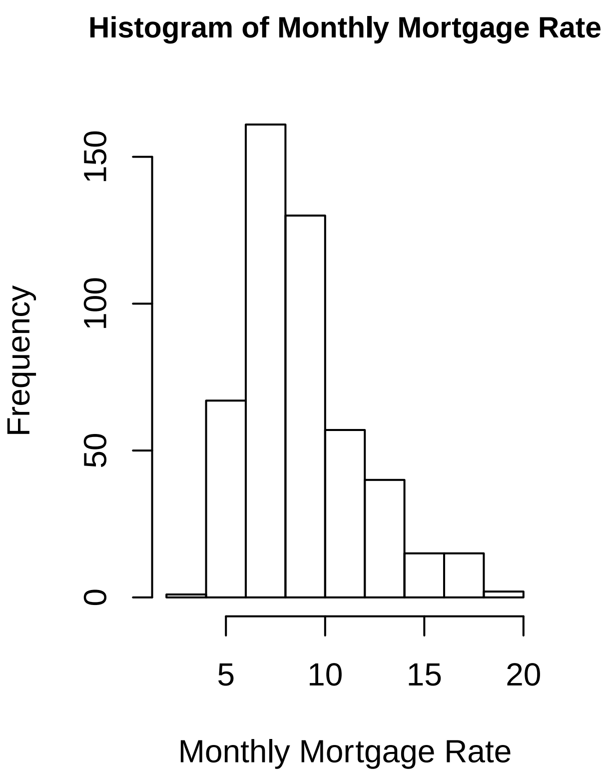
12/02/2020

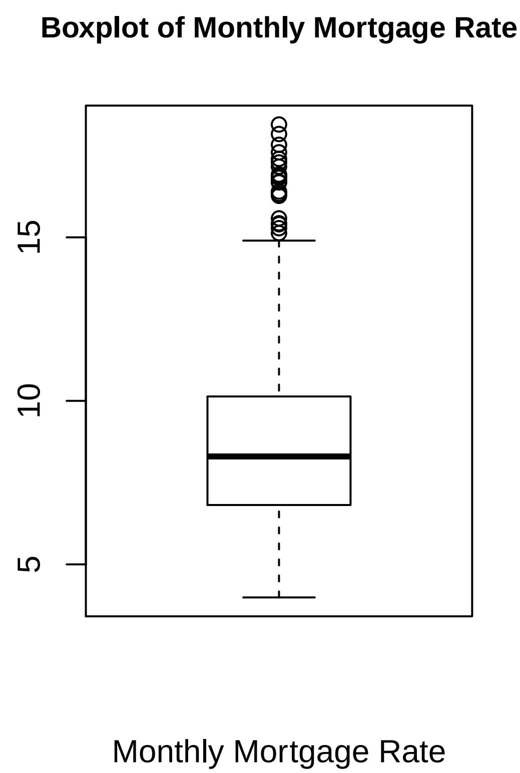
Introduction

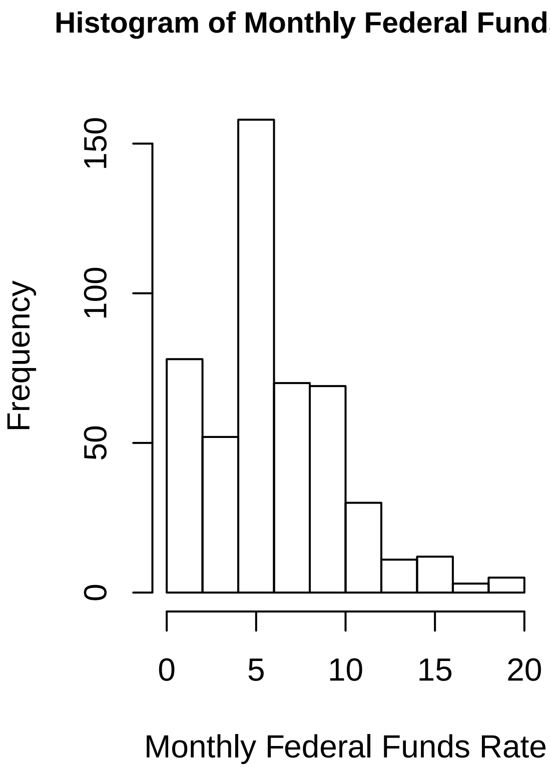
A mortgage is known as a type of loan that people can borrow money from banks and financial institutions to buy or refinance a home or piece of property. The mortgage rate is the interest rate of the mortgage, which can be fixed or fluctuate depending on the agreements. In order to understand the past mortgage rates, predict future mortgage rates, and make relative policy suggestions, the dataset of the US monthly 30-year conventional mortgage rates from April 1971 to November 2011 is obtained from Federal Reserve Economic Data to develop a precise statistical model. We are also interested in investigating the relationship between the mortgage rates and the Federal Funds rates. There are in total of 488 observations in the dataset collected from 1971 to 2011. There are 5 variables in the dataset, which are year, month, day, morg known as the monthly mortgage rate, and ffr known as the monthly federal funds rate. There is one additional date variable constructed based on the values of variables year, month, and day, which is useful for the time series analysis. In the following report, we are going to examine the stationarity of the data, transform the data if it is not stationary, build ARIMA models, perform model checking and selection, and fit a time series model for the mortgage rate and the lag-1 federal funds.

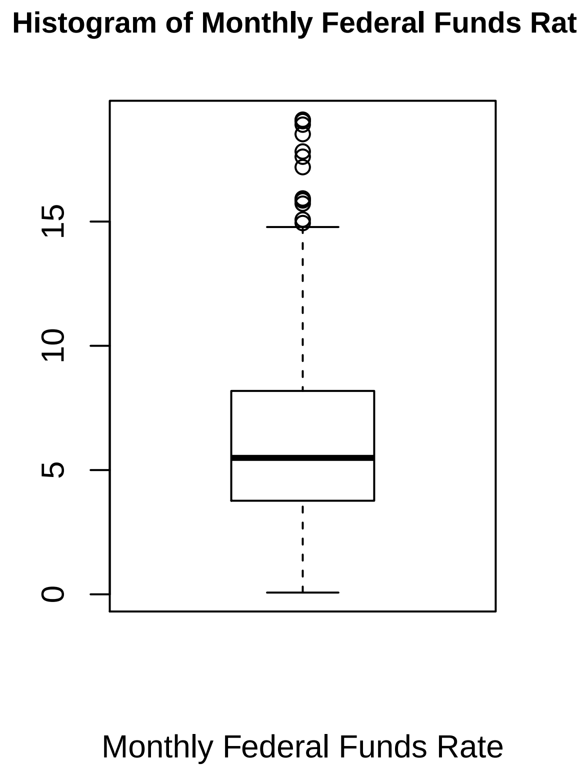
Material and Methods

This dataset is time-series data because each observation is collected at successive time periods, which is the first day of each month from April 1971 to November 2011. For the variable morg, the minimum monthly mortgage rate is 3.990, the maximum monthly mortgage rate is 18.450, and the mean is 8.800. From the histogram, we can see that the monthly mortgage rate is right-skewed, which means that there are a lot of observations that are larger than the mode. The boxplot reveals the same information that there are a lot of outliers beyond the upper whisker. For the variable ffr, the minimum monthly federal funds rate is 0.070, the maximum monthly federal funds rate is 19.100, and the mean is 5.995. The histogram of the monthly federal funds rate also shows a right skewness, and the boxplot points out a lot of outliers beyond the upper whisker.

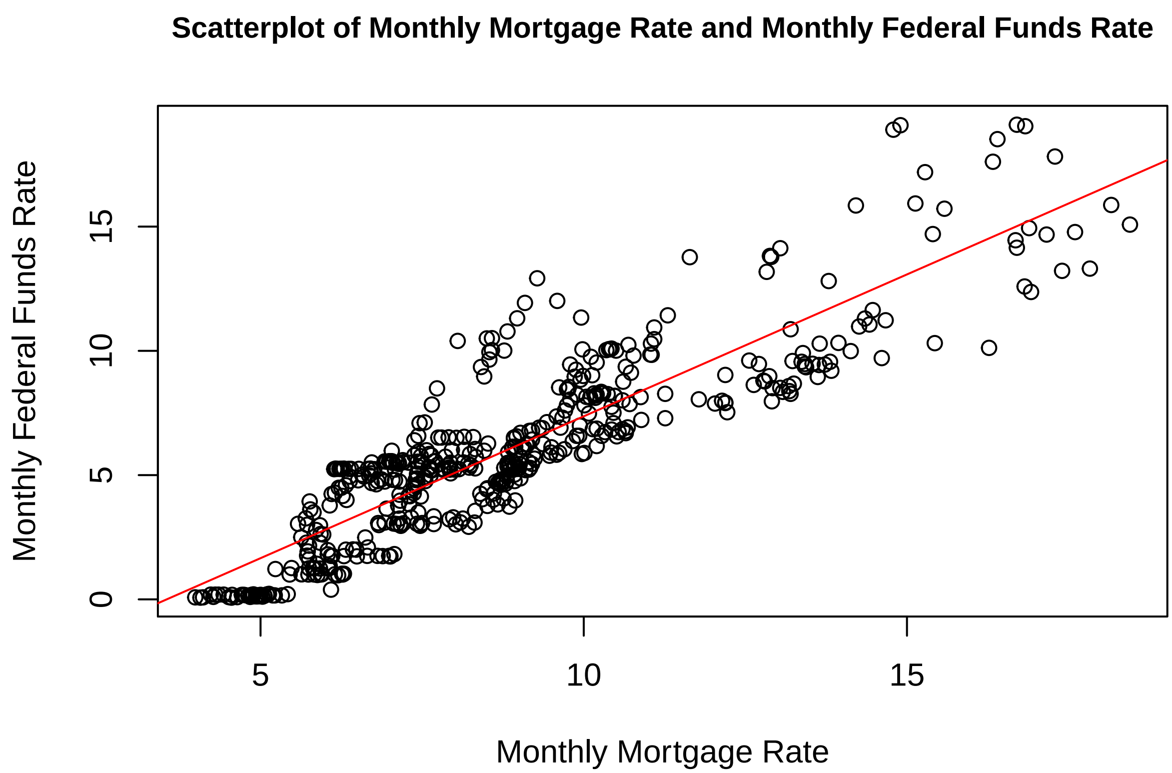




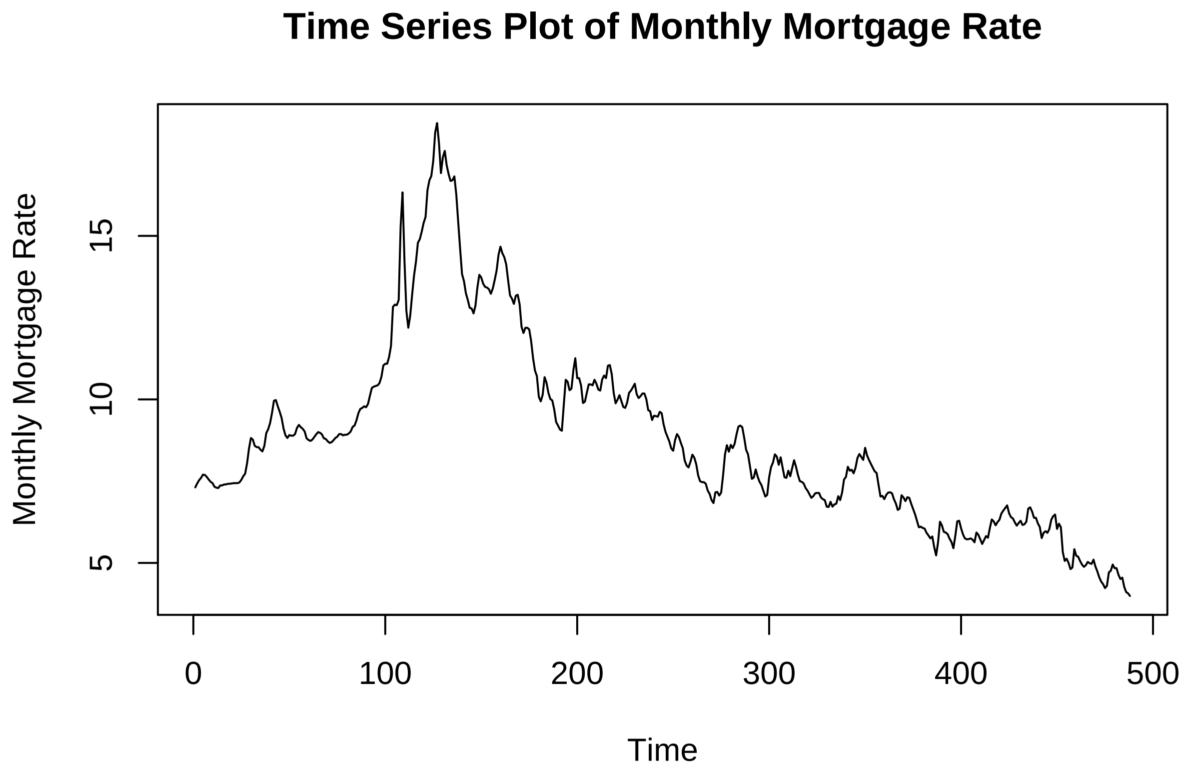


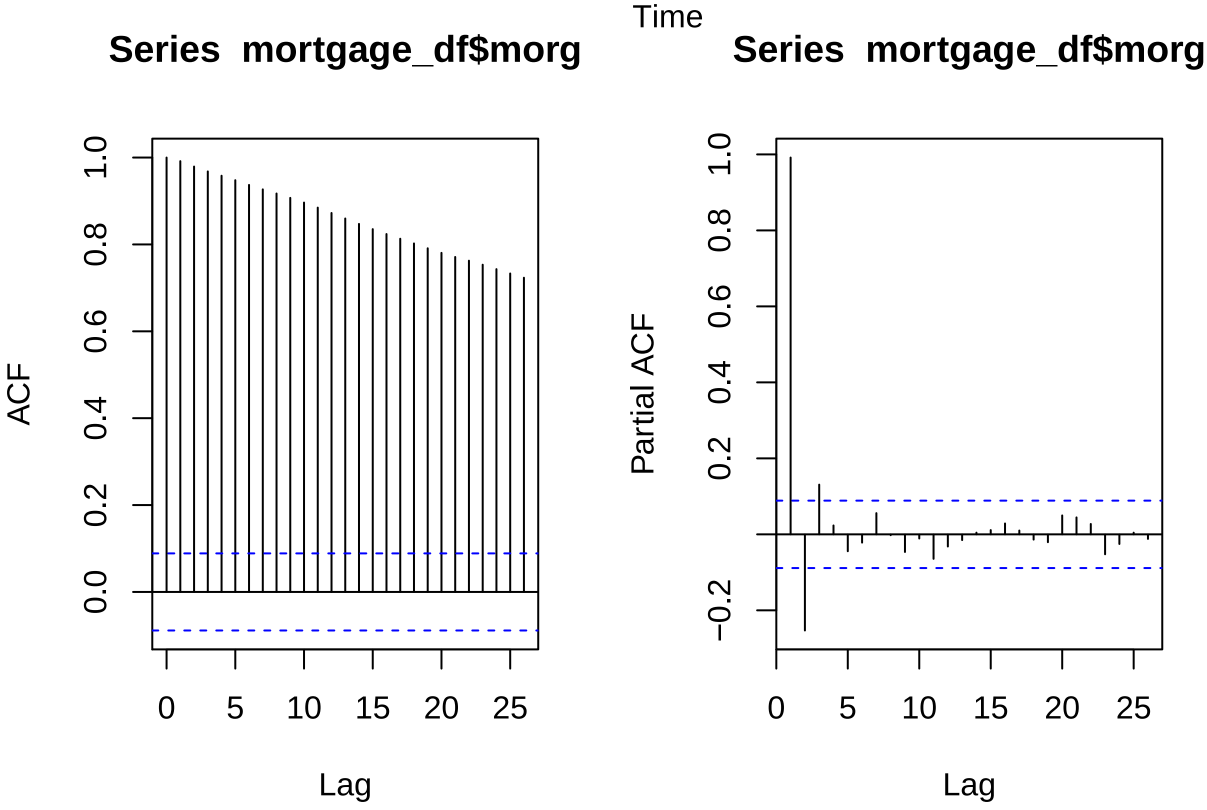


The scatter plot of these two variables shows that there is a positive linear relationship between the monthly mortgage interest rate and the monthly federal funds rate. As the monthly mortgage interest rate increases, the monthly federal funds rate also tends to increase.



Then, from the time series plot of the monthly mortgage interest rate, it can be seen that there is a clear trend in the data, and there is no constant mean and variance. So, the data is not stationary. The ACF ordinates are large and decay slowly, and the PACF cuts off after lag 1. These observations also confirm that the data is not stationary.

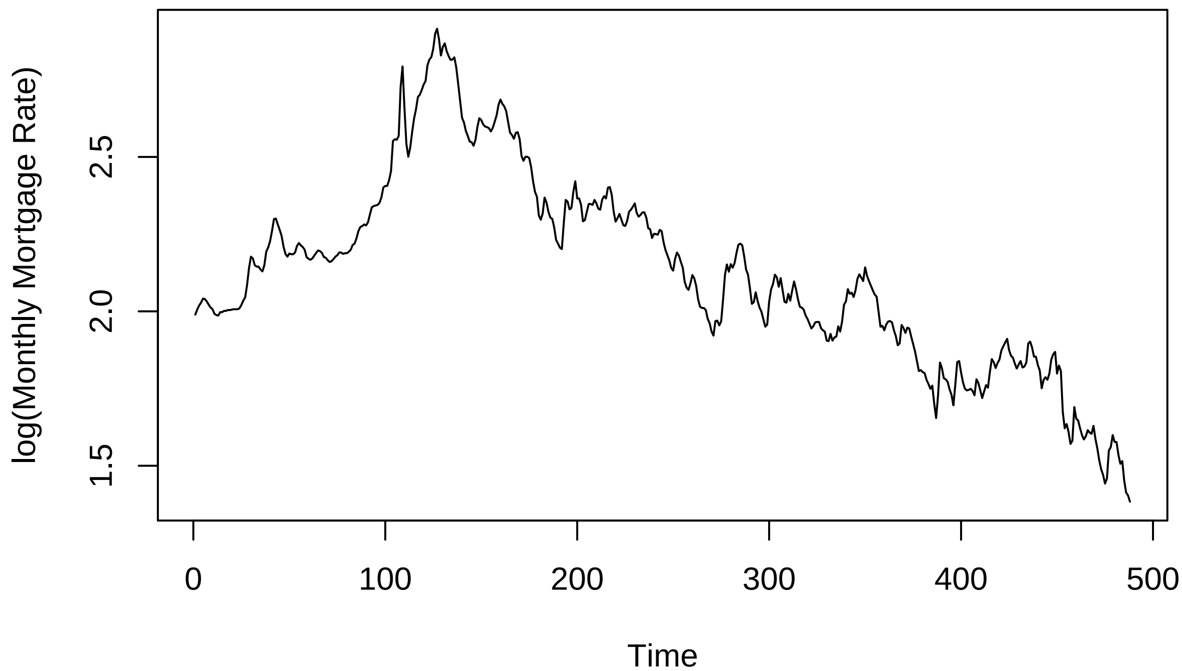




In order to fix the problem of nonstationarity, we can apply transformations to the original data, such as log transformation, log differencing, and differencing. Next, we are going to apply each of these transformations to the data to see if they can make the transformed series stationary.

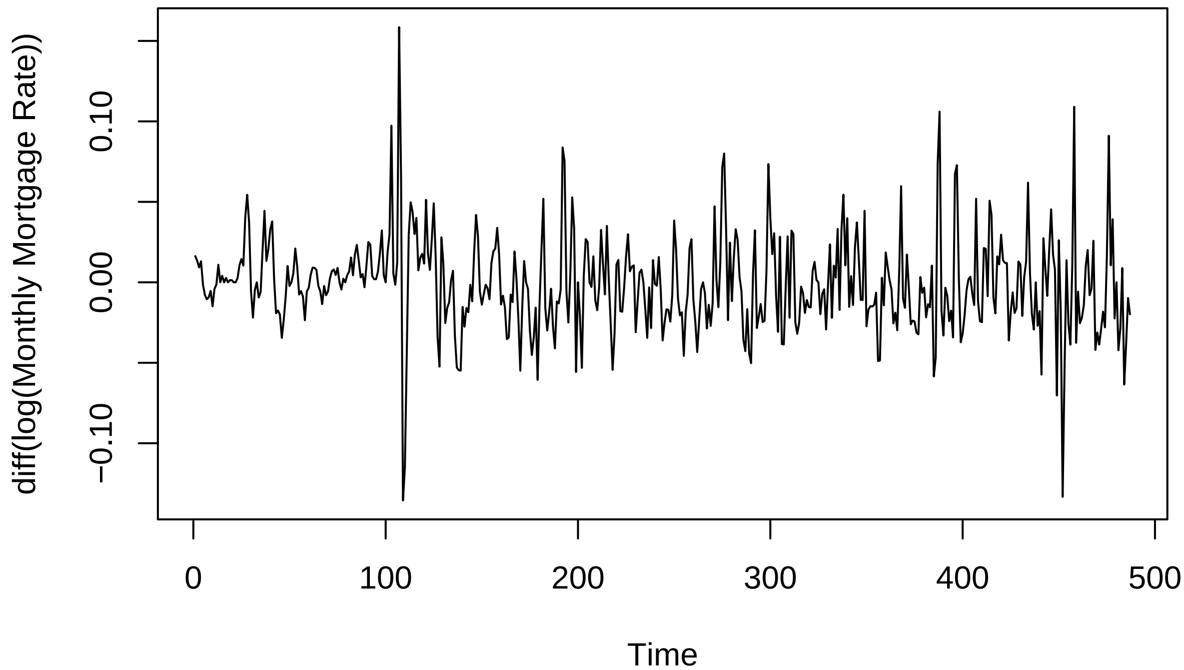
Log Transformation

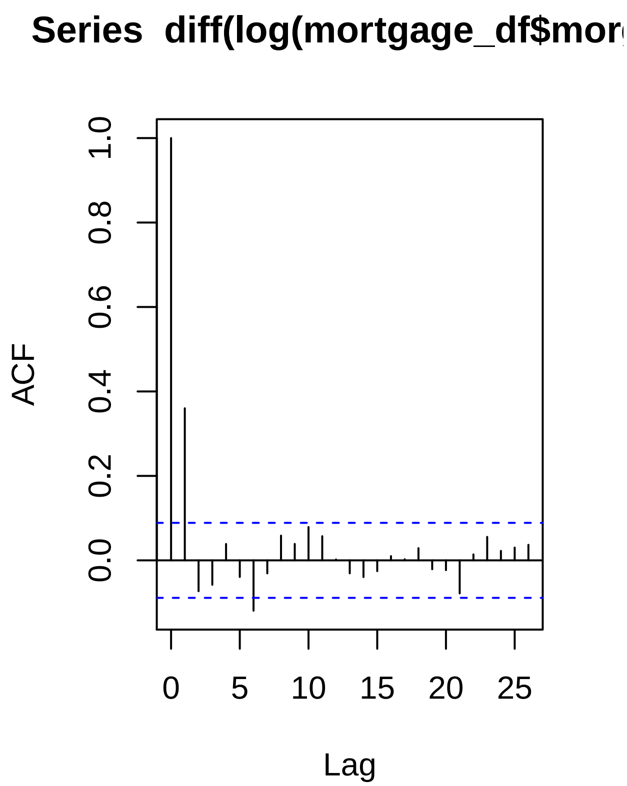
After applying log transformation to the data, we can see that there is still a obvious trend in the plot and there is no constant mean and variance. The problem of nonstationarity is not fix by using log transformation. So, we will not continue using this method.

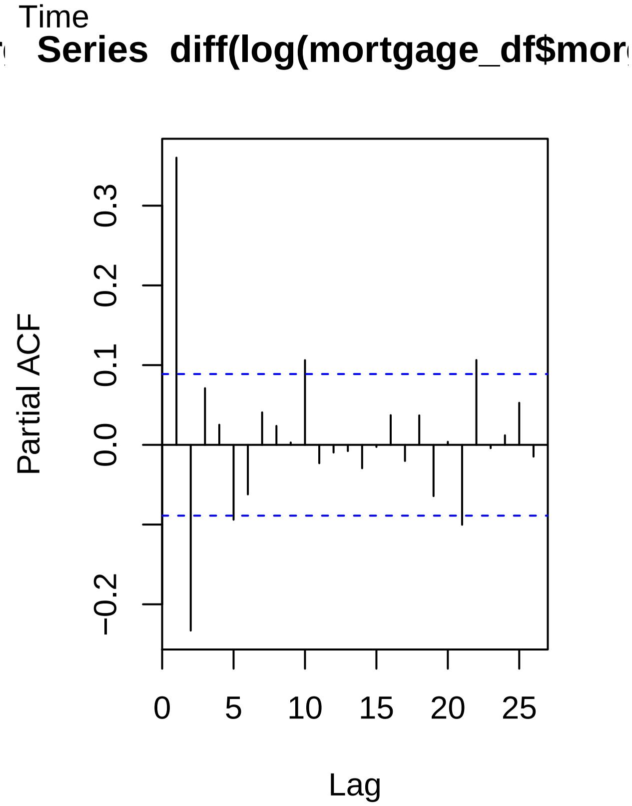


Log Differencing

Then, we apply log differencing to the original data to see if this method helps. From the new time series plot, we observe that there is no significant trend and the data is roughly stationary. The ACF and PACF plots of the log difference series suggest a MA(2) model and a ARMA(2, 2) model.





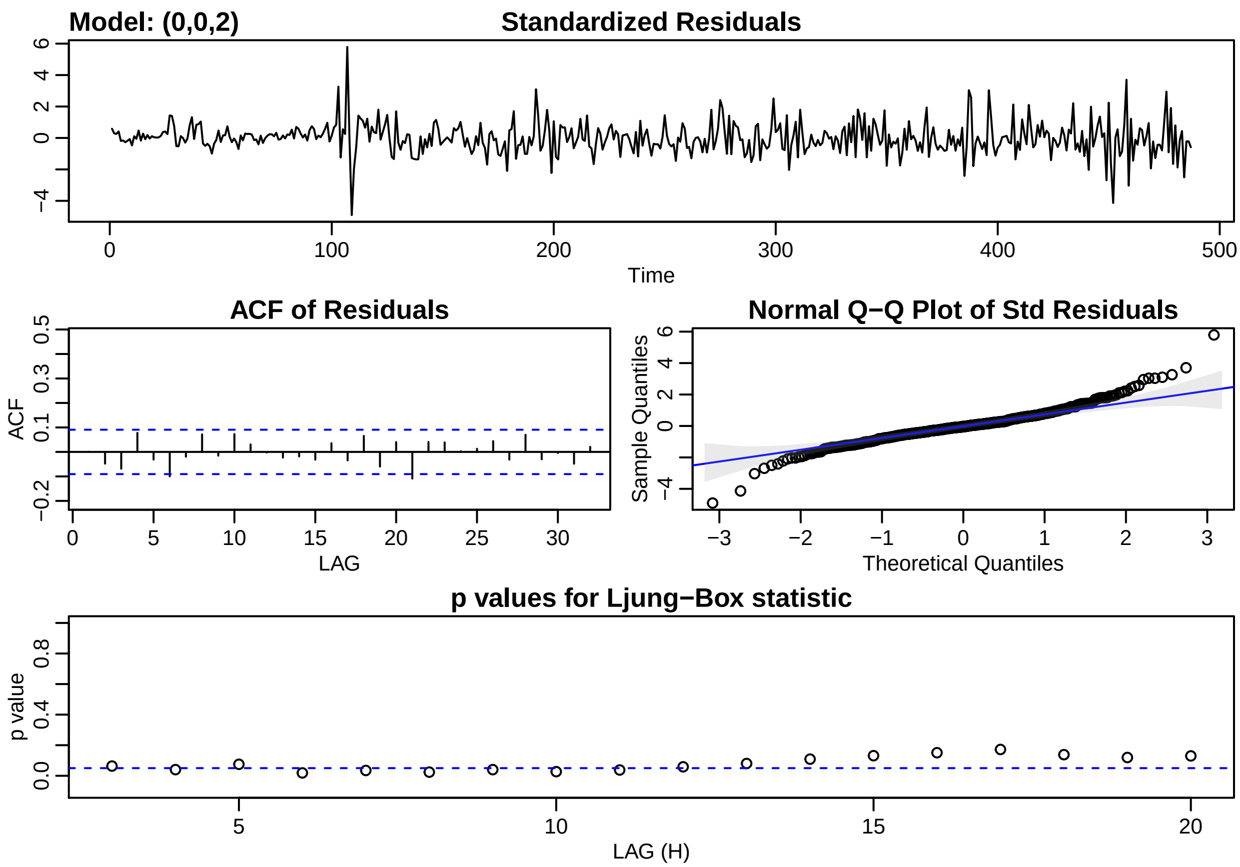


Then, we examine both models to see if they fit the data well.

Fitting MA(2) to the log difference series

```
## initial  value -3.514889
## iter   2 value -3.605247
## iter   3 value -3.608431
## iter   4 value -3.609573
```

```
## iter   5 value -3.609741
## iter   6 value -3.609742
## iter   6 value -3.609742
## iter   6 value -3.609742
## final  value -3.609742
## converged
## initial  value -3.609552
## iter   2 value -3.609553
## iter   3 value -3.609553
## iter   3 value -3.609553
## iter   3 value -3.609553
## final  value -3.609553
## converged
```



Estimated parameters of MA(2)

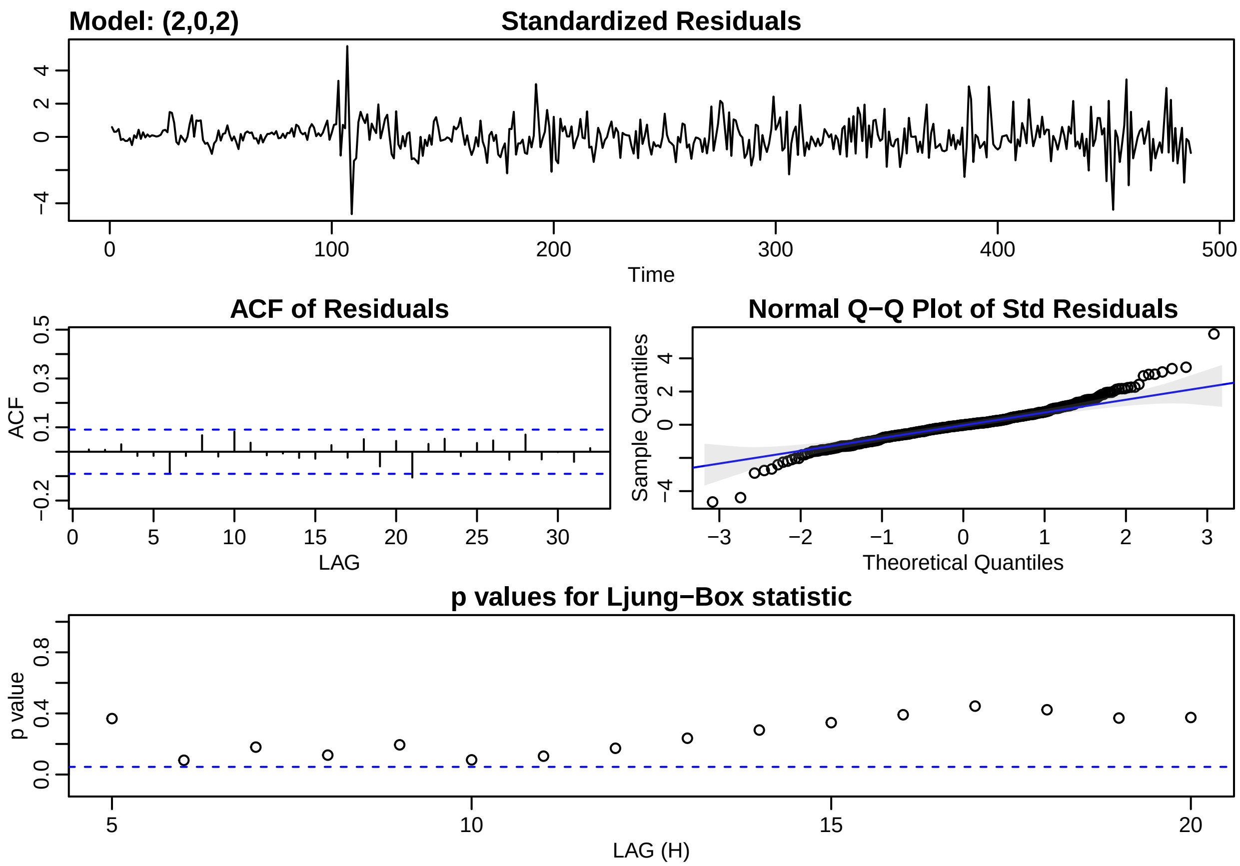
```
##        Estimate     SE t.value p.value
## ma1      0.4566 0.0484  9.4263  0.0000
## ma2      0.0008 0.0481  0.0166  0.9867
## xmean   -0.0012 0.0018 -0.6959  0.4868
```

Fitting ARMA(2, 2) to the log difference series

```
fit_arma = astsa::sarima(diff(log(mortgage_df$morg)), 2, 0, 2)
```

```
## initial  value -3.513436
## iter   2 value -3.573468
## iter   3 value -3.606906
## iter   4 value -3.609356
## iter   5 value -3.611974
## iter   6 value -3.612228
## iter   7 value -3.612610
## iter   8 value -3.612891
## iter   9 value -3.612946
## iter  10 value -3.613002
## iter  11 value -3.613152
## iter  12 value -3.613486
## iter  13 value -3.613728
## iter  14 value -3.613899
## iter  15 value -3.613961
## iter  16 value -3.614118
## iter  17 value -3.614346
## iter  18 value -3.614447
## iter  19 value -3.614581
## iter  20 value -3.614866
## iter  21 value -3.615141
## iter  22 value -3.615394
## iter  23 value -3.615454
## iter  24 value -3.615454
## iter  25 value -3.615454
## iter  26 value -3.615455
## iter  27 value -3.615455
## iter  28 value -3.615455
## iter  29 value -3.615455
## iter  30 value -3.615455
## iter  31 value -3.615455
## iter  32 value -3.615455
## iter  33 value -3.615455
## iter  33 value -3.615455
## iter  33 value -3.615455
## final  value -3.615455
## converged
## initial  value -3.617051
## iter   2 value -3.617051
## iter   3 value -3.617052
## iter   4 value -3.617052
## iter   5 value -3.617053
## iter   6 value -3.617054
## iter   7 value -3.617054
## iter   8 value -3.617054
## iter   9 value -3.617054
## iter  10 value -3.617055
## iter  11 value -3.617055
## iter  11 value -3.617055
```

```
## iter  11 value -3.617055
## final  value -3.617055
## converged
```



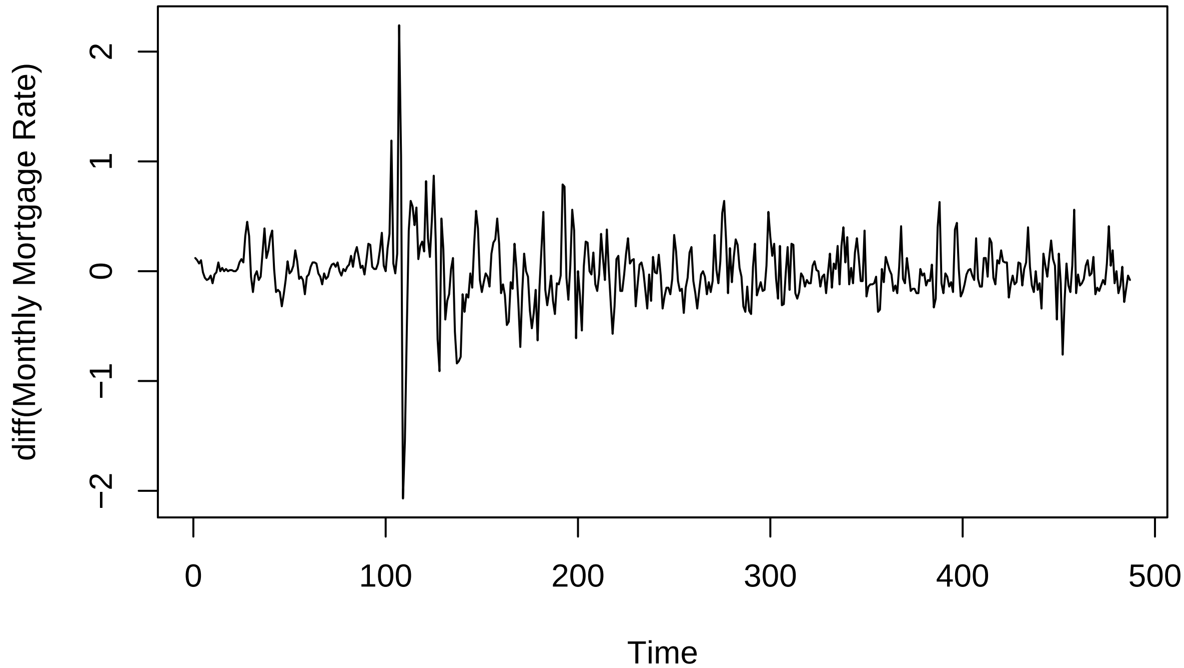
Estimated parameters of ARMA(2, 2)

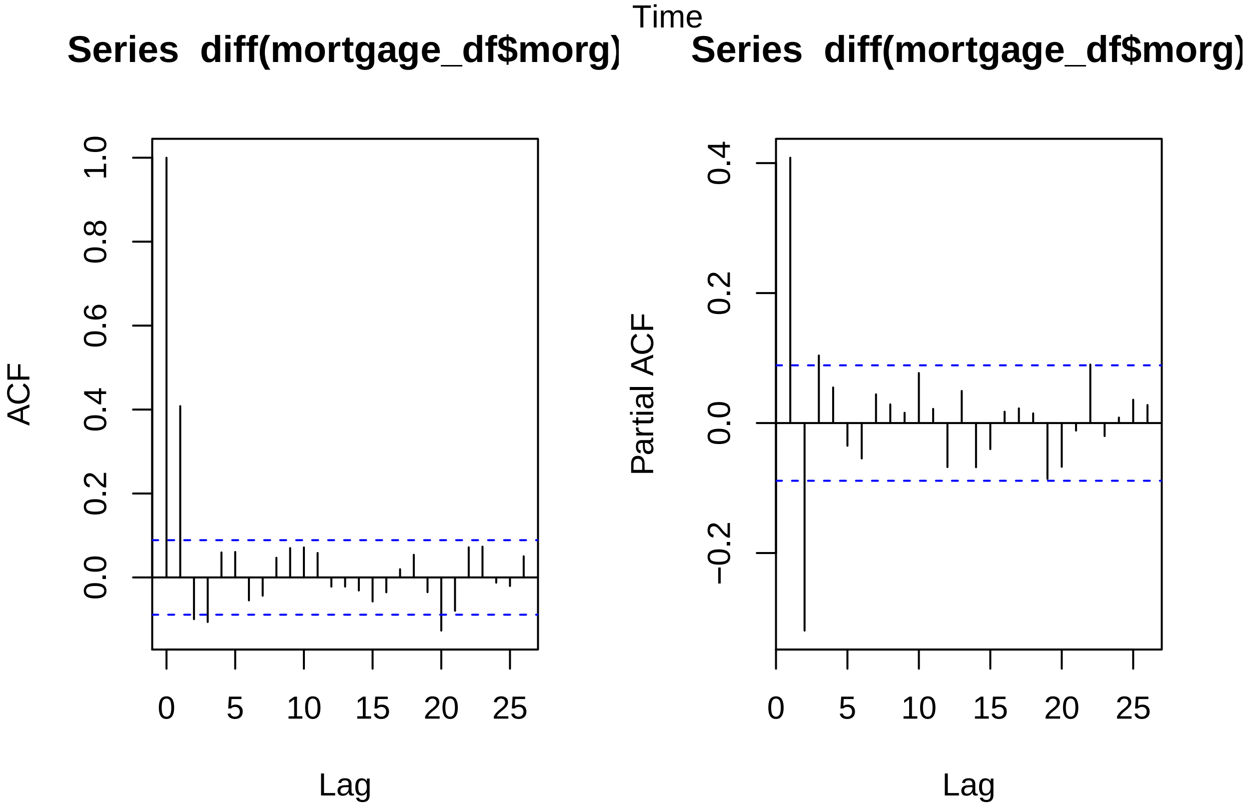
```
##        Estimate     SE t.value p.value
## ar1     -0.1598 0.1838 -0.8693  0.3851
## ar2     -0.3093 0.1149 -2.6920  0.0073
## ma1      0.6118 0.1799  3.4003  0.0007
## ma2      0.3345 0.1401  2.3871  0.0174
## xmean   -0.0013 0.0016 -0.7855  0.4326
```

From the results above, there is no apparent trend or pattern in both plots of the standardized residuals. For both models, the ACF shows no apparent significant dependence structure, as the ordinates are within the blue bounds. For the normal Q-Q plots, most of the points lie on the blue line. There are some deviations on both tails, probably due to the existence of outliers. But the normality assumption seems to be appropriate with the exception of outliers for both models. The only difference is that all the p-values for Ljung-Box statistics are above the blue dotted line for ARMA(2, 2), but some of the p-values for MA(2) are on or below the blue dotted line. Hence, ARMA(2, 2) for the log difference series is a better model for the data. From the estimated parameters for the model ARMA(2, 2), we can see that some of the p values are very large, which indicates that some of the estimated parameters are not statistically significant. So, this model may not be the best model to fit the data. We might need to consider other models.

Differencing

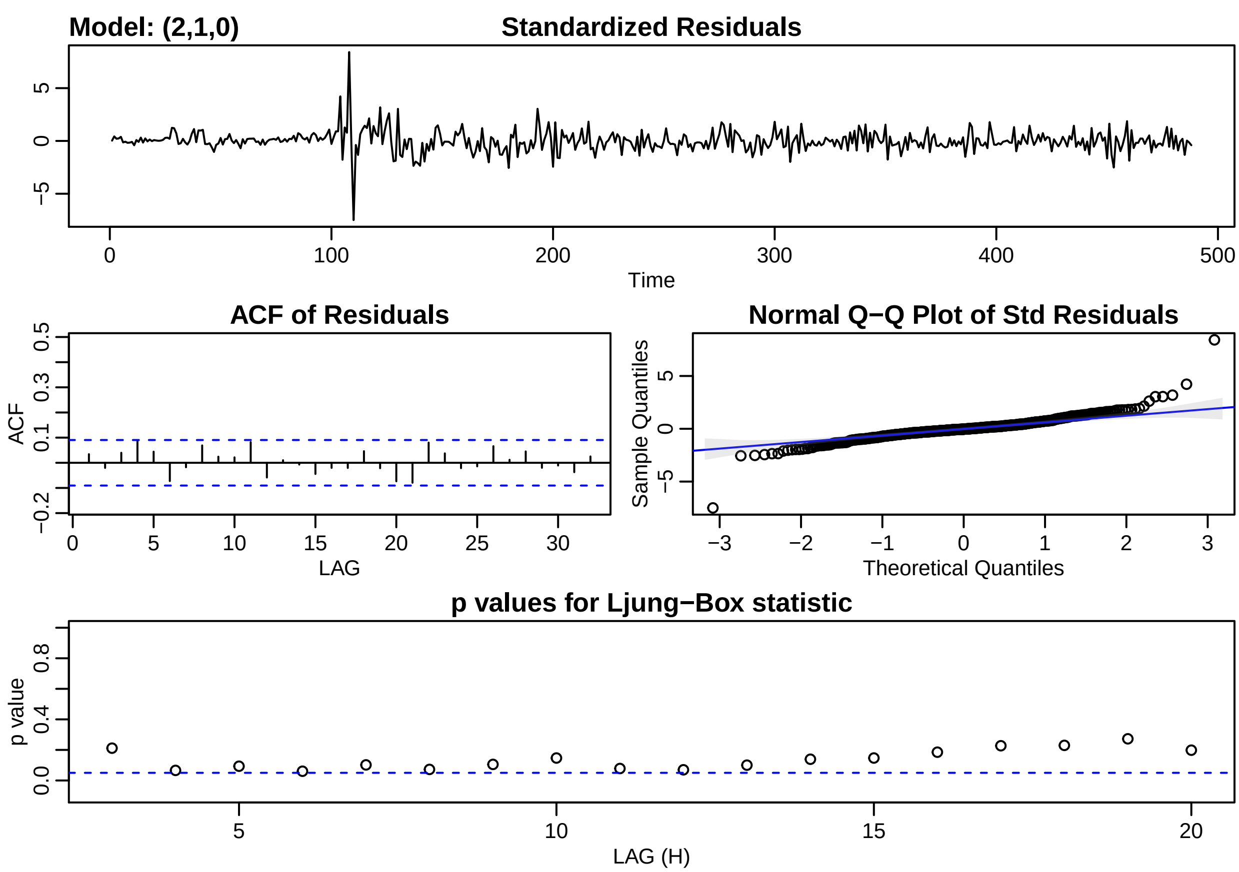
Differencing is another transformation that can be used. By using the first-order difference, the data becomes roughly stationary in the new graph, and there is no obvious trend. The ACF and PACF plots suggest a ARIMA(2, 1, 0) model and a ARIMA(2, 1, 2) model to the original series.





Fitting ARIMA(2, 1, 0)

```
## initial  value -1.216814
## iter   2 value -1.334494
## iter   3 value -1.355518
## iter   4 value -1.361740
## iter   5 value -1.361758
## iter   6 value -1.361759
## iter   7 value -1.361759
## iter   8 value -1.361759
## iter   8 value -1.361759
## final  value -1.361759
## converged
## initial  value -1.363175
## iter   2 value -1.363176
## iter   3 value -1.363177
## iter   4 value -1.363177
## iter   5 value -1.363177
## iter   5 value -1.363177
## iter   5 value -1.363177
## final  value -1.363177
## converged
```

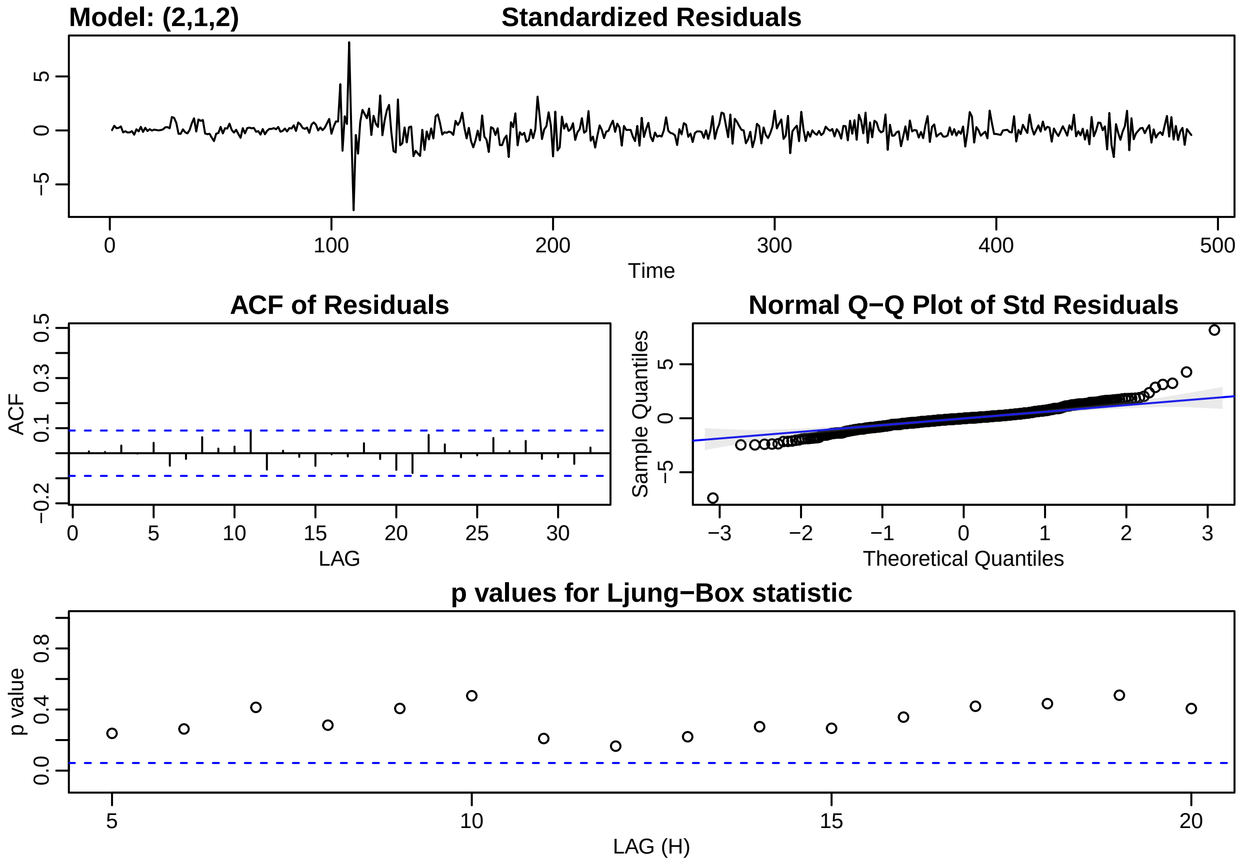


Estimated parameters of ARIMA(2, 1, 0)

```
##           Estimate     SE t.value p.value
## ar1         0.5378 0.0429 12.5361   0.000
## ar2        -0.3182 0.0429 -7.4251   0.000
## constant  -0.0068 0.0149 -0.4610   0.645
```

Fitting ARIMA(2, 1, 2)

```
## initial  value -1.216814
## iter   2 value -1.323210
## iter   3 value -1.354730
## iter   4 value -1.360775
## iter   5 value -1.364259
## iter   6 value -1.365165
## iter   7 value -1.366168
## iter   8 value -1.366699
## iter   9 value -1.366804
## iter  10 value -1.367050
## iter  11 value -1.367448
## iter  12 value -1.367909
## iter  13 value -1.368028
## iter  14 value -1.368135
## iter  15 value -1.368182
## iter  16 value -1.368240
## iter  17 value -1.368257
## iter  18 value -1.368263
## iter  19 value -1.368274
## iter  20 value -1.368290
## iter  21 value -1.368305
## iter  22 value -1.368311
## iter  23 value -1.368311
## iter  23 value -1.368311
## final  value -1.368311
## converged
## initial  value -1.369793
## iter   2 value -1.369794
## iter   3 value -1.369795
## iter   4 value -1.369795
## iter   5 value -1.369795
## iter   6 value -1.369795
## iter   6 value -1.369795
## iter   6 value -1.369795
## final  value -1.369795
## converged
```



Estimated parameters of ARIMA(2, 1, 2)

```
##          Estimate     SE t.value p.value
## ar1        0.0860 0.1666  0.5162  0.6059
## ar2       -0.3260 0.0867 -3.7621  0.0002
## ma1        0.4736 0.1657  2.8575  0.0045
## ma2        0.2428 0.1240  1.9589  0.0507
## constant  -0.0068 0.0159 -0.4290  0.6681
```

From the results above, we conclude that there is no apparent trend or pattern in the plots of standardized residuals. For both models, the ACF shows no apparent significant dependence structure, as the ordinates are within the blue bounds. For the normal Q-Q plots, most of the points lie on the blue line. There are some deviations on both tails, probably due to the existence of outliers. But the normality assumption seems to be appropriate with the exception of outliers for both models. All the p-values for Ljung-Box statistics are above the blue dotted line, which indicates that the processes match the white noise process. Thus, both models fit well to the data and estimated parameters are significant.

In order to find the best fit model, we are going to perform model selection by coparing AIC, AICc, and BIC.

ARIMA(2, 1, 0)

AIC:

```
## [1] 0.1279502
```

AICc:

```
## [1] 0.1280522
```

BIC:

```
## [1] 0.1623507
```

ARIMA(2, 1, 2)

AIC:

```
## [1] 0.1229282
```

AICc:

```
## [1] 0.1231843
```

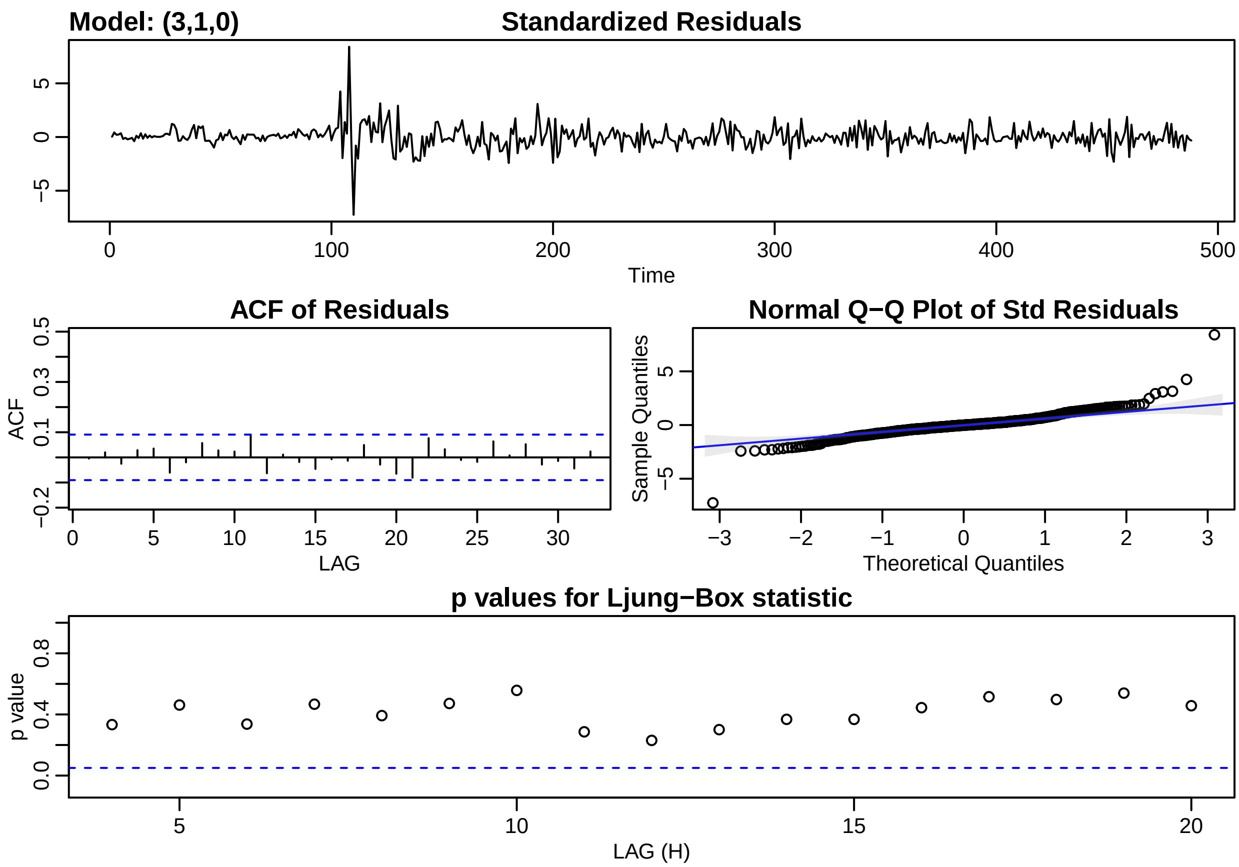
BIC:

```
## [1] 0.174529
```

From the above results, we notice that both AIC and AICc are smaller for ARIMA(2, 1, 2), which indicates that ARIMA(2, 1, 2) is a better fit. The BIC criterion is smaller for ARIMA(2, 1, 0). And the statistics are close to each other for both models, so they are both good fits. However, we also notice that some of the estimated parameters for ARIMA(2, 1, 2) are not significant. In this case, ARIMA(2, 1, 0) might be a better choice. As a final check, we might consider to overfit the model to see if the results change significantly. We find out that ARIMA(3, 1, 0) fits the data even better. Also, the AIC, AICc, and BIC are smaller for ARIMA(3, 1, 0) compared to others. Therefore, we think ARIMA(3, 1, 0) is the best fit model.

$$x_t = 0.5710x_{t-1} + 0.5710x_{t-2} + 0.1037x_{t-3} + w_t$$

```
## initial  value -1.215851
## iter   2 value -1.329738
## iter   3 value -1.358749
## iter   4 value -1.365203
## iter   5 value -1.366111
## iter   6 value -1.366241
## iter   7 value -1.366246
## iter   8 value -1.366246
## iter   9 value -1.366246
## iter  10 value -1.366246
## iter  10 value -1.366247
## final  value -1.366247
## converged
## initial  value -1.368603
## iter   2 value -1.368604
## iter   3 value -1.368604
## iter   4 value -1.368605
## iter   5 value -1.368605
## iter   5 value -1.368605
## iter   5 value -1.368605
## final  value -1.368605
## converged
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = constant, transform.pars = trans, fixed = fixed,
##     optim.control = list(trace = trc, REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ar2     ar3  constant
##        0.571  -0.3742  0.1037   -0.0068
## s.e.   0.045   0.0490  0.0450    0.0165
##
## sigma^2 estimated as 0.06469:  log likelihood = -24.51,  aic = 59.02
##
## $degrees_of_freedom
## [1] 483
##
## $ttable
##          Estimate     SE t.value p.value
## ar1        0.5710 0.0450 12.6783  0.0000
## ar2       -0.3742 0.0490 -7.6293  0.0000
## ar3        0.1037 0.0450  2.3061  0.0215
## constant  -0.0068 0.0165 -0.4157  0.6778
##
## $AIC
## [1] 0.1212004
```

```
##
## $AICc
## [1] 0.1213708
##
## $BIC
## [1] 0.1642011
```

```
##          Estimate     SE t.value p.value
## ar1        0.5710 0.0450 12.6783  0.0000
## ar2       -0.3742 0.0490 -7.6293  0.0000
## ar3        0.1037 0.0450  2.3061  0.0215
## constant  -0.0068 0.0165 -0.4157  0.6778
```

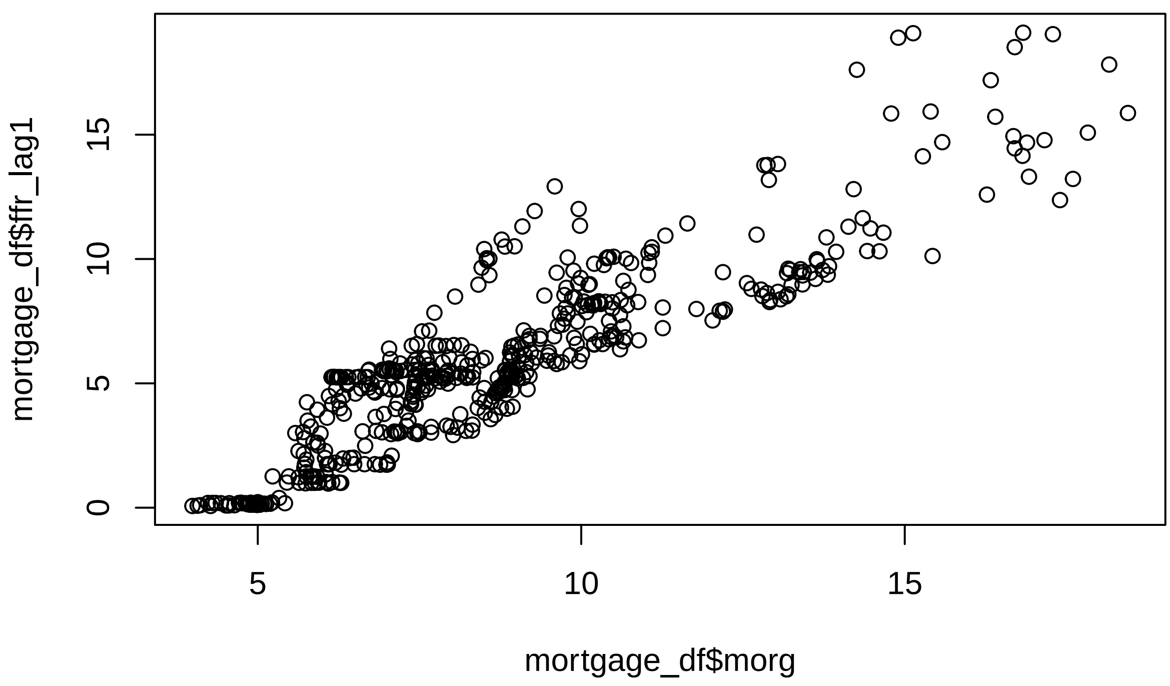
```
## [1] 0.1212004
```

```
## [1] 0.1213708
```

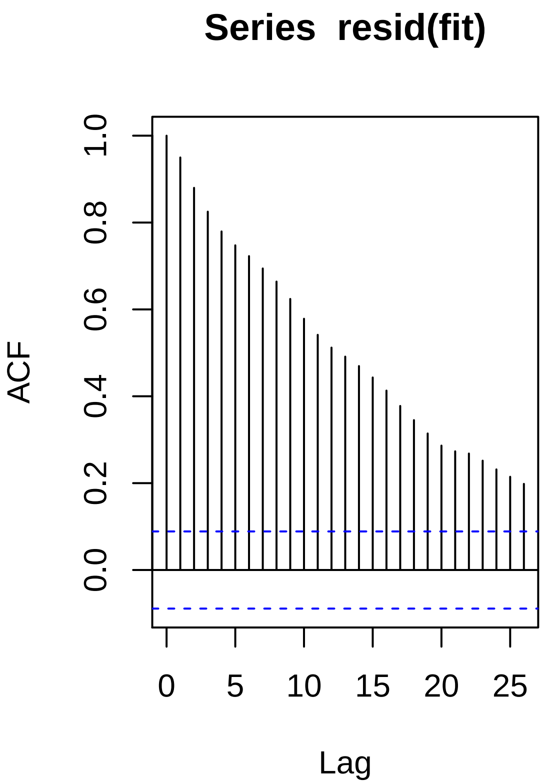
```
## [1] 0.1642011
```

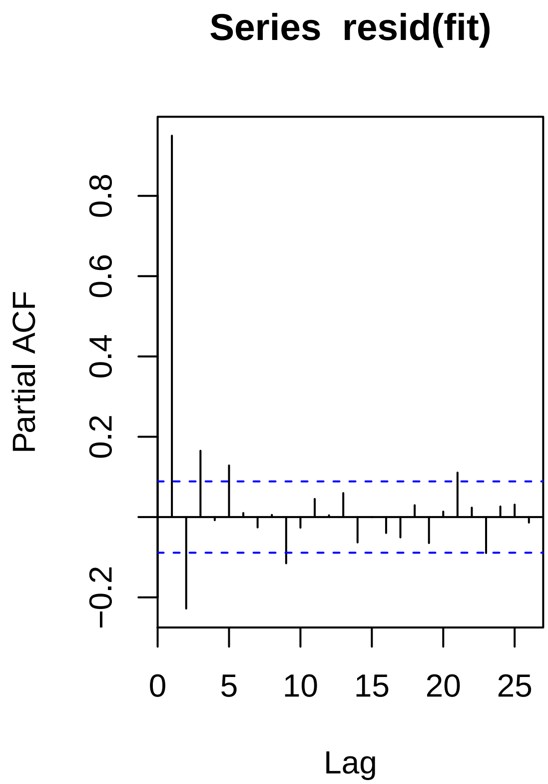
Next, we are going to find a time series model for the mortgage rate using the lag-1 federal funds rate as an explanatory variable. From the plot, we can see that there is a positive linear relationship between the mortgage rate and the lag-1 federal funds. This suggests that the model to consider could be

$$M_t = \beta_0 + \beta_1 F_{t-1} + x_t$$



Then, based on the result, the plots of ACF and PACF are consistent with those of a AR(1) process.

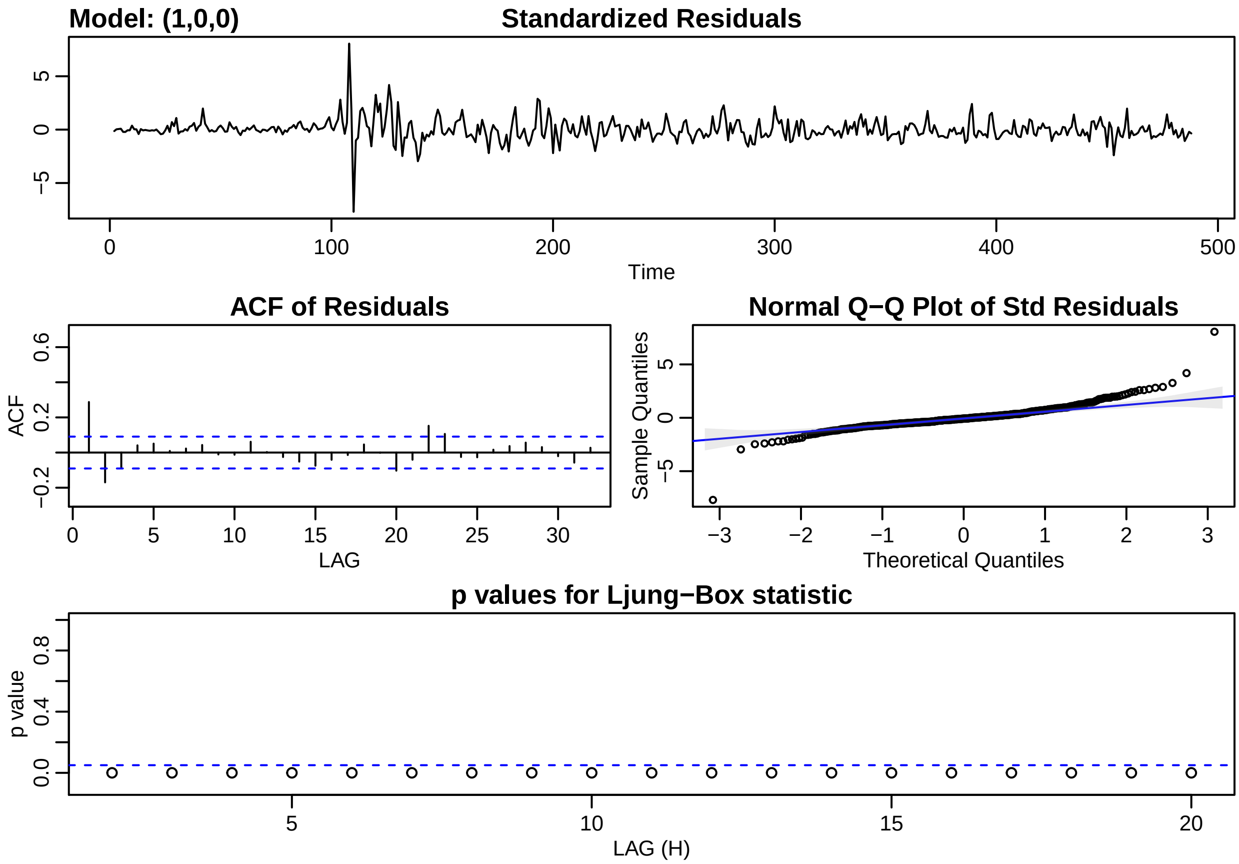





```
## initial  value 0.261434
## iter   2 value -0.715033
## iter   3 value -0.939016
## iter   4 value -0.969965
## iter   5 value -1.253429
## iter   6 value -1.277195
## iter   7 value -1.286742
## iter   8 value -1.286765
## iter   9 value -1.286770
## iter  10 value -1.286920
## iter  11 value -1.286950
## iter  12 value -1.287450
## iter  13 value -1.288540
## iter  14 value -1.288737
## iter  15 value -1.289909
## iter  16 value -1.290133
## iter  17 value -1.290912
## iter  18 value -1.290913
## iter  19 value -1.290939
## iter  20 value -1.290948
## iter  21 value -1.290952
## iter  22 value -1.290952
## iter  23 value -1.290953
## iter  24 value -1.290981
## iter  25 value -1.290982
## iter  26 value -1.290985
## iter  27 value -1.290988
## iter  28 value -1.290993
## iter  29 value -1.290997
```

```
## iter  30 value -1.290999
## iter  31 value -1.291013
## iter  32 value -1.291014
## iter  33 value -1.291036
## iter  34 value -1.291042
## iter  35 value -1.291045
## iter  36 value -1.291045
## iter  37 value -1.291046
## iter  38 value -1.291063
## iter  39 value -1.291065
## iter  40 value -1.291074
## iter  41 value -1.291079
## iter  42 value -1.291082
## iter  43 value -1.291084
## iter  44 value -1.291085
## iter  45 value -1.291097
## iter  46 value -1.291099
## iter  47 value -1.291116
## iter  48 value -1.291119
## iter  49 value -1.291120
## iter  50 value -1.291120
## iter  51 value -1.291120
## iter  52 value -1.291129
## iter  53 value -1.291130
## iter  54 value -1.291141
## iter  55 value -1.291146
## iter  56 value -1.291149
## iter  57 value -1.291150
## iter  58 value -1.291151
## iter  59 value -1.291164
## iter  60 value -1.291166
## iter  61 value -1.291177
## iter  62 value -1.291178
## iter  63 value -1.291178
## iter  64 value -1.291179
## iter  65 value -1.291179
## iter  66 value -1.291180
## iter  67 value -1.291181
## iter  68 value -1.291185
## iter  69 value -1.291193
## iter  70 value -1.291210
## iter  71 value -1.291232
## iter  72 value -1.291235
## iter  73 value -1.291251
## iter  74 value -1.291251
## iter  75 value -1.291264
## iter  76 value -1.291265
## iter  77 value -1.291266
## iter  78 value -1.291266
## iter  79 value -1.291266
## iter  80 value -1.291270
## iter  81 value -1.291271
## iter  82 value -1.291276
## iter  83 value -1.291278
```

```
## iter  84 value -1.291279
## iter  85 value -1.291280
## iter  86 value -1.291280
## iter  87 value -1.291285
## iter  88 value -1.291286
## iter  89 value -1.291289
## iter  90 value -1.291290
## iter  91 value -1.291290
## iter  91 value -1.291290
## iter  91 value -1.291290
## final  value -1.291290
## converged
```



Results

$$x_t = 0.5710x_{t-1} + 0.5710x_{t-2} + 0.1037x_{t-3} + w_t$$

$$M_t = 4.62099 + 0.69606F_{t-1} + x_t$$

$$x_t = 0.9931x_{t-1} + w_t$$

where {wt} ~ WN(0, 0.07492)

Appendix

```
mortgage_df = read.delim("mortgage.txt", sep = " ", header = TRUE)
mortgage_df$date = as.Date(ISOdate(year = mortgage_df$year,
                                   month = mortgage_df$month,
                                   day = mortgage_df$day))
dimension = dim(mortgage_df)
min = mortgage_df[1,1:2]
max = mortgage_df[488,1:2]
par(mfrow=c(1, 2))
hist(mortgage_df$morg, cex.main=0.9,
     xlab = "Monthly Mortgage Rate",
     main = "Histogram of Monthly Mortgage Rate")
boxplot(mortgage_df$morg, cex.main=0.9,
        xlab = "Monthly Mortgage Rate",
        main = "Boxplot of Monthly Mortgage Rate")
par(mfrow=c(1, 2))
hist(mortgage_df$ffr, cex.main=0.9,
     xlab = "Monthly Federal Funds Rate",
     main = "Histogram of Monthly Federal Funds Rate")
boxplot(mortgage_df$ffr, cex.main=0.9,
        xlab = "Monthly Federal Funds Rate",
        main = "Histogram of Monthly Federal Funds Rate")
par(mfrow=c(1, 1))
plot(mortgage_df$morg, mortgage_df$ffr, cex.main=0.9,
     xlab = "Monthly Mortgage Rate",
     ylab = "Monthly Federal Funds Rate",
     main = "Scatterplot of Monthly Mortgage Rate and Monthly Federal Funds Rate")
abline(lm(mortgage_df$ffr~mortgage_df$morg), col = "red")
par(mfrow=c(1, 1))
ts.plot(mortgage_df$morg, type = "l",
        ylab = "Monthly Mortgage Rate",
        main = "Time Series Plot of Monthly Mortgage Rate")
par(mfrow=c(1, 2))
acf(mortgage_df$morg)
pacf(mortgage_df$morg)
par(mfrow=c(1, 1))
ts.plot(log(mortgage_df$morg), type = "l", ylab = "log(Monthly Mortgage Rate)")
par(mfrow=c(1, 1))
ts.plot(diff(log(mortgage_df$morg)), type = "l",
        ylab = "diff(log(Monthly Mortgage Rate))")
par(mfrow=c(1, 2))
acf(diff(log(mortgage_df$morg)))
pacf(diff(log(mortgage_df$morg)))
fit_ma = astsa::sarima(diff(log(mortgage_df$morg)), 0, 0, 2)
fit_ma$ttable
fit_arma = astsa::sarima(diff(log(mortgage_df$morg)), 2, 0, 2)
fit_arma$ttable
par(mfrow=c(1, 1))
ts.plot(diff(mortgage_df$morg), type = "l",
        ylab = "diff(Monthly Mortgage Rate)")
par(mfrow=c(1, 2))
acf(diff(mortgage_df$morg))
```

```r
pacf(diff(mortgage_df$morg))
fit_arima210 = astsa::sarima(mortgage_df$morg, 2, 1, 0)
fit_arima210$ttable
fit_arima212 = astsa::sarima(mortgage_df$morg, 2, 1, 2)
fit_arima212$ttable
fit_arima210$AIC
fit_arima210$AICc
fit_arima210$BIC
fit_arima212$AIC
fit_arima212$AICc
fit_arima212$BIC
fit_arima310 = astsa::sarima(mortgage_df$morg, 3, 1, 0)
fit_arima310
fit_arima310$ttable
fit_arima310$AIC
fit_arima310$AICc
fit_arima310$BIC
mortgage_df$ffr_lag1 = Hmisc::Lag(mortgage_df$ffr, 1)
plot(mortgage_df$morg, mortgage_df$ffr_lag1)
fit = lm(mortgage_df$morg ~ mortgage_df$ffr_lag1)
par(mfrow=c(1, 2))
acf(resid(fit))
pacf(resid(fit))
x = astsa::sarima(mortgage_df$morg, 1, 0, 0, xreg = mortgage_df$ffr_lag1)
```