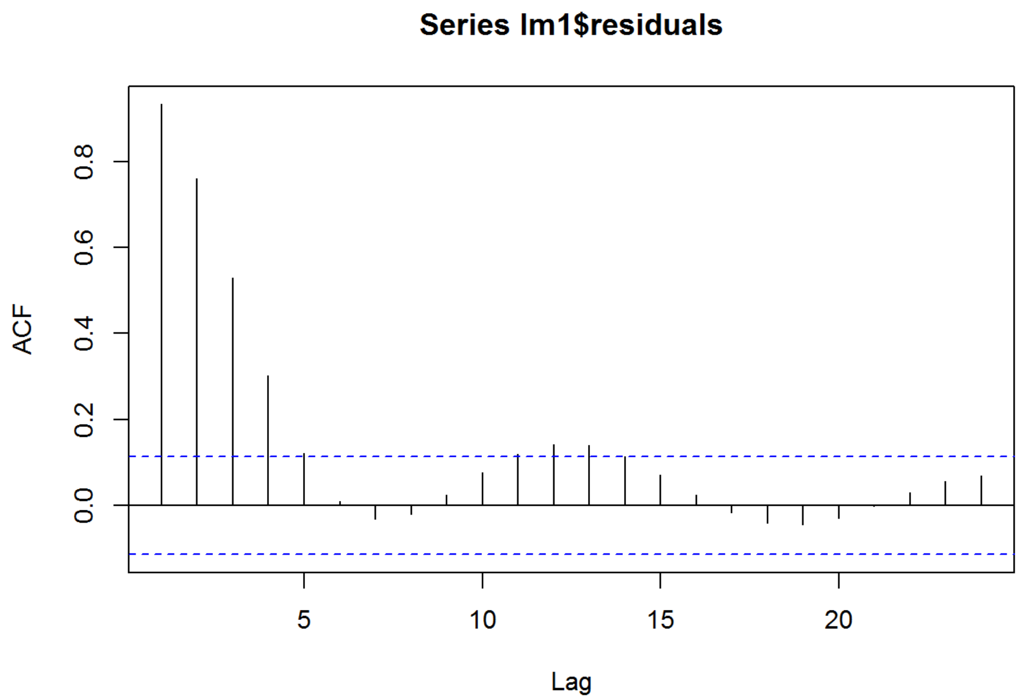
HW 6

Adham Suliman

November 11, 2018

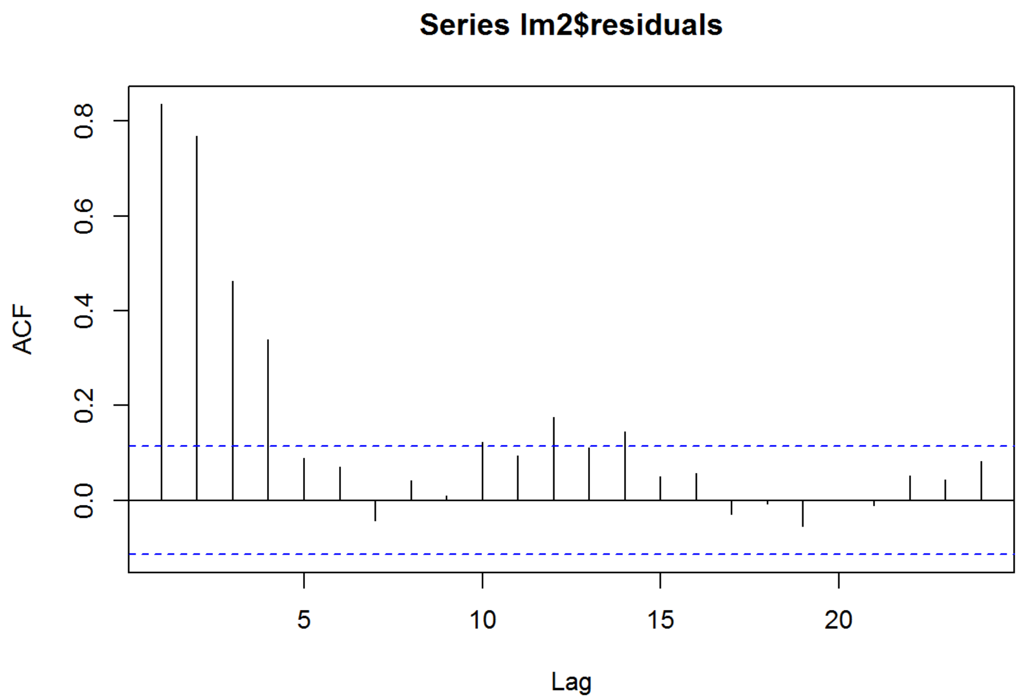
1.Use linear regression model - plot the ACF - what can you conclude ? ACF of residuals shows autocorrelation up to lag 5.

```
lm1 <- lm(df[,2]~df[,1])
acf(lm1$residuals)
```



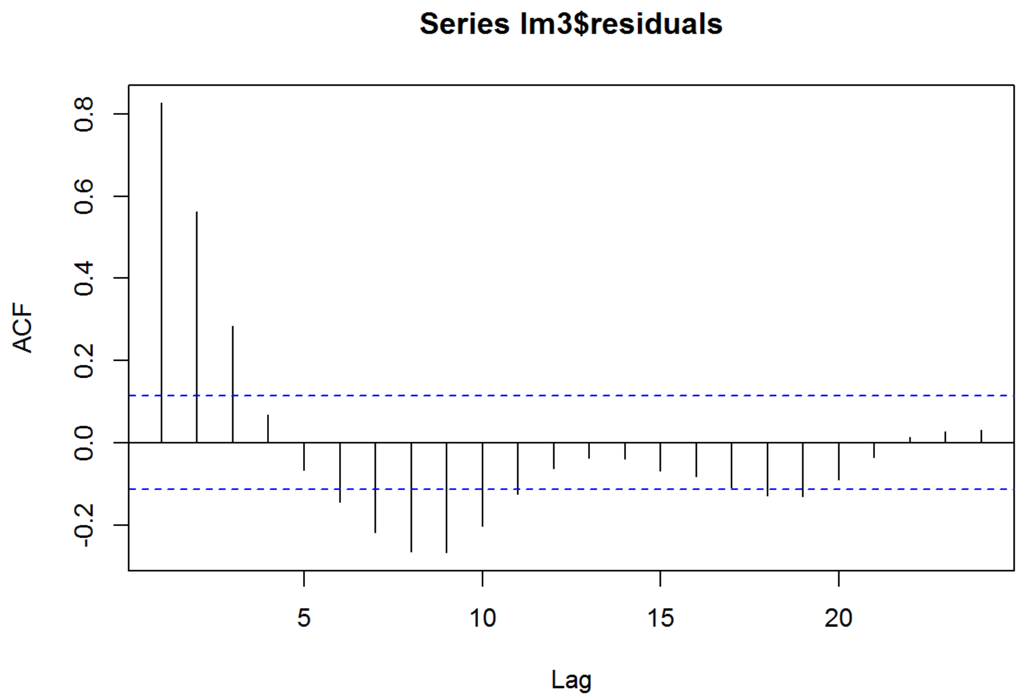
2.Use ARIMA (0,0,1) model for the residuals. Adjust the Input gas rate and Output CO2 % with the MA coefficient. Combine with the linear regression model. Plot the residuals.

```
lm2 <- Arima(df[,2], order = c(0,0,1), xreg=df[,1])
acf(lm2$residuals)
```



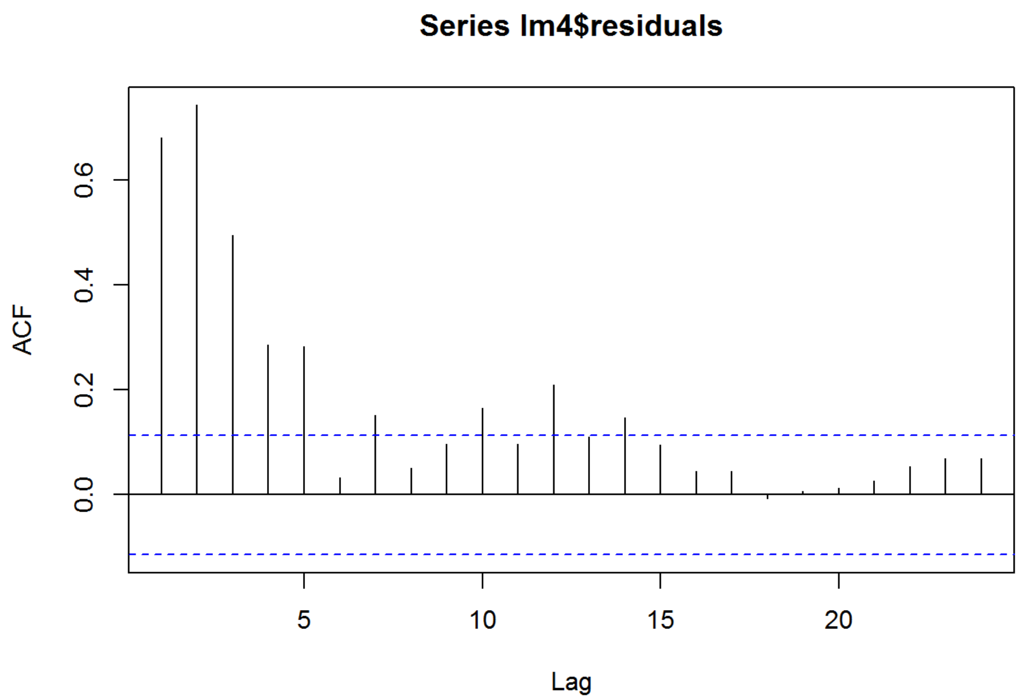
3.Use ARIMA (1,0,0) model for the residuals. Adjust the Input gas rate and Output CO2 % with the AR coefficient. Combine with the linear regression model. Plot the residuals.

```
lm3 <- Arima(df[,2], order = c(1,0,0), xreg=df[,1])
acf(lm3$residuals)
```



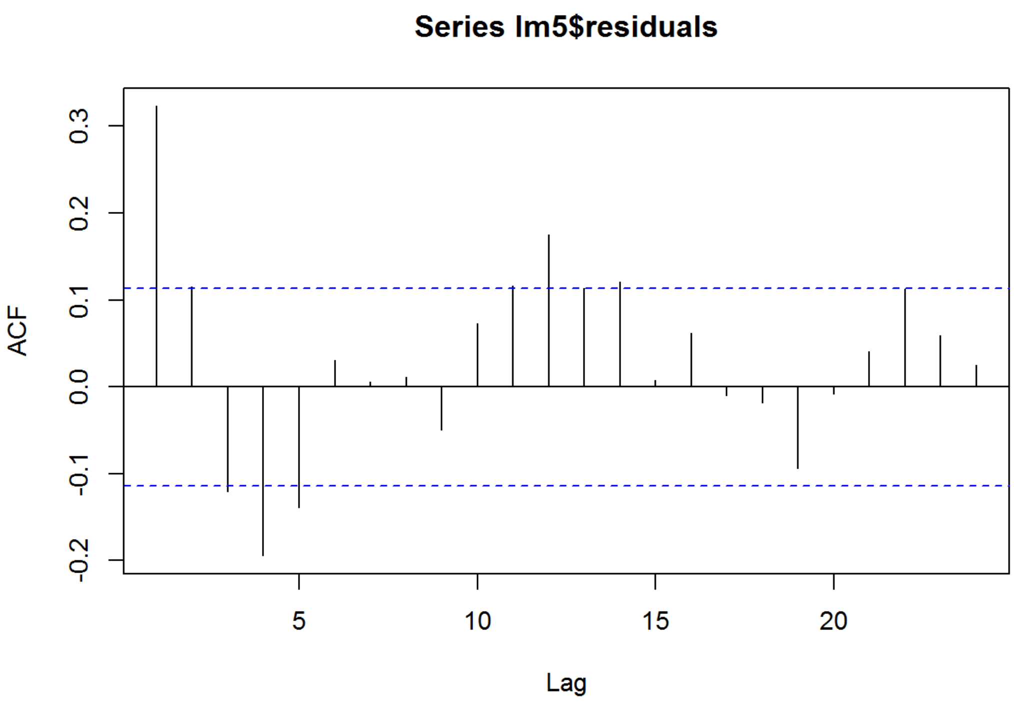
4.Use ARIMA (0,0,2) model for the residuals. Adjust the Input gas rate and Output CO2 % with the MA coefficient. Combine with the linear regression model. Plot the residuals.

```
lm4 <- Arima(df[,2], order = c(0,0,2), xreg=df[,1])
acf(lm4$residuals)
```



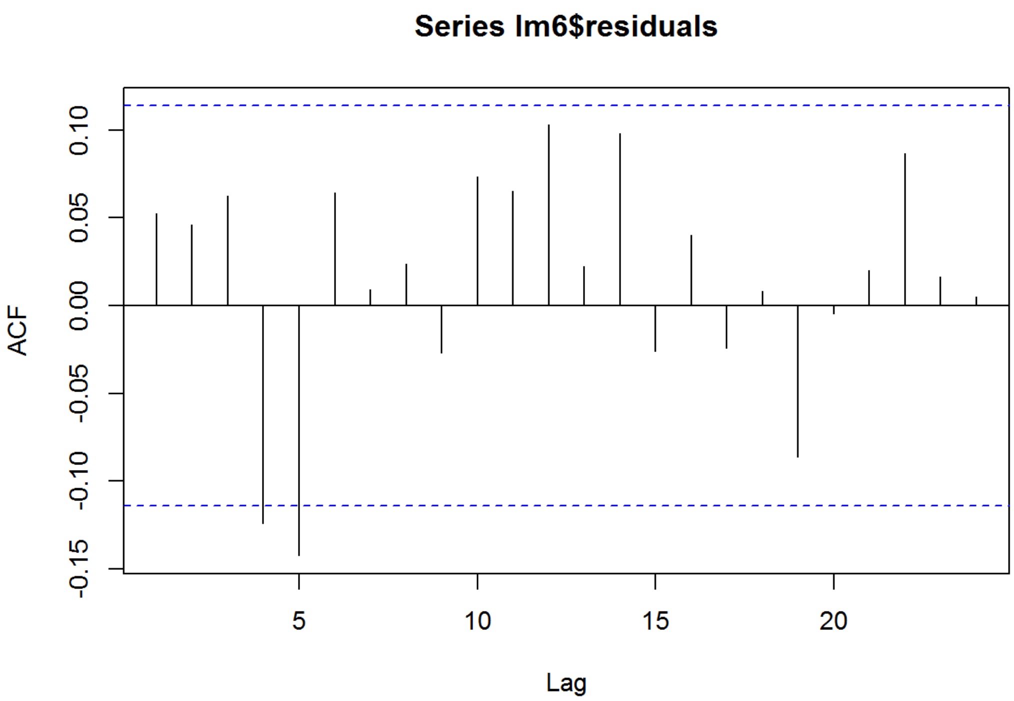
5.Use ARIMA (2,0,0) model for the residuals. Adjust the Input gas rate and Output CO2 % with the AR coefficient. Combine with the linear regression model. Plot the residuals.

```r
lm5 <- Arima(df[,2], order = c(2,0,0), xreg=df[,1])
acf(lm5$residuals)
```



6.Use ARIMA (2,0,2) model for the residuals. Adjust the Input gas rate and Output CO2 % with the AR and MA coefficients. Combine with the linear regression model. Plot the residuals.

```r
lm6 <- Arima(df[,2], order = c(2,0,2), xreg=df[,1])
acf(lm6$residuals)
```



7.Use fractional ARIMA model (aka ARFIMA) for the output gas CO2% - plot the residuals, acf and pacf plots of the model. You can use an R package like fracdiff (or arfima from forecast pkg) – be careful to determine which lag to choose when executing this test.

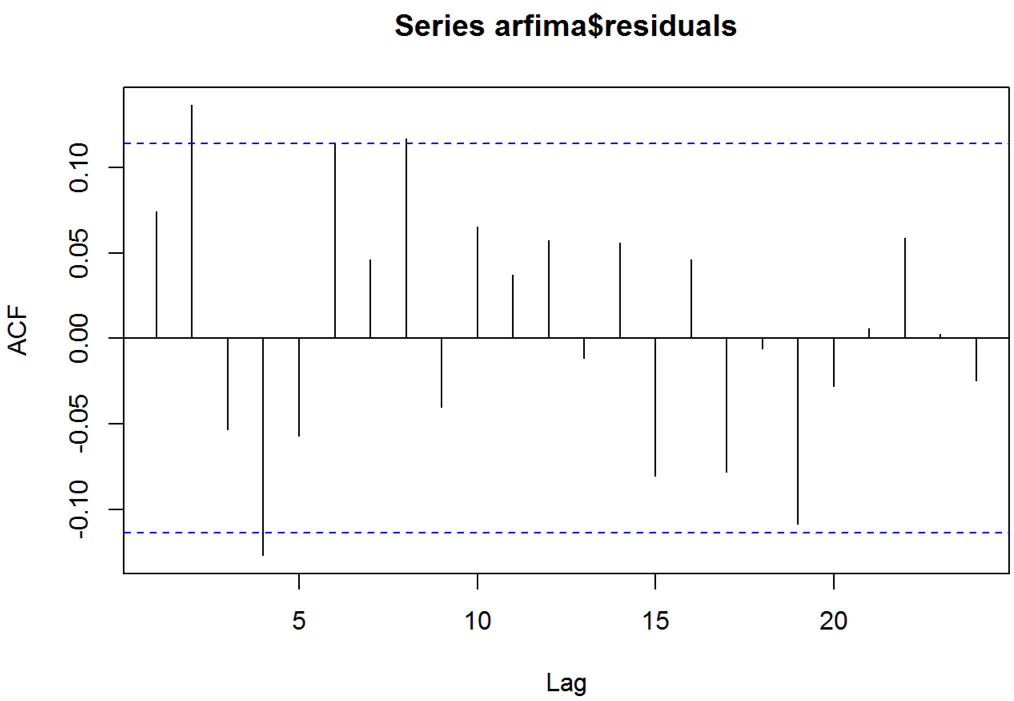
```r
?arfima
```

```
## starting httpd help server ... done
```

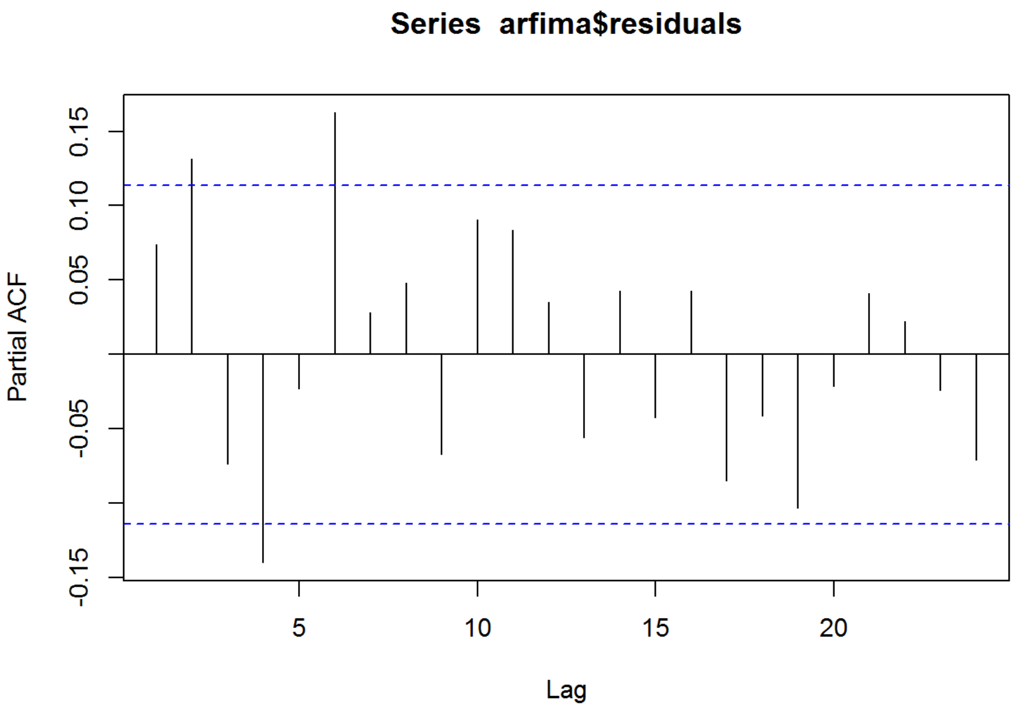
```r
(arfima <- arfima(df[,2]))
```

```
##
## Call:
##   arfima(y = df[, 2])
##
## *** Warning during (fdcov) fit: unable to compute correlation matrix; maybe change 'h'
##
## Coefficients:
##           d      ar.ar1      ar.ar2      ar.ar3      ar.ar4      ma.ma1
##  0.4038515  0.2676080  0.5678993  0.4520453 -0.6989489 -1.4302172
##      ma.ma2
## -0.9999975
## sigma[eps] = 0.3283512
## a list with components:
##  [1] "log.likelihood"  "n"               "msg"
##  [4] "d"               "ar"              "ma"
##  [7] "covariance.dpq"  "fnormMin"        "sigma"
## [10] "stderror.dpq"    "correlation.dpq" "h"
## [13] "d.tol"           "M"               "hessian.dpq"
## [16] "length.w"        "call"            "residuals"
## [19] "x"               "fitted"          "series"
```

```r
acf(arfima$residuals)
```




```r
pacf(arfima$residuals)
```



8.Perform Summaries, Durbin-Watson and Box-Ljung tests for each model and build table to compare AICs and p-values for each test across the ARIMA and ARFIMA models. The null hypothesis for the Durbin-Watson test states that there is no correlation among the residuals, while the alternative hypothesis states that there is autocorrelation among the residuals. The null hypothesis for the Ljung-Box test states that there is no autocorrelation between the residuals while the alternative hypothesis states that the residuals are autocorrelative.

The box.test is producing a significant p-value below, and the durbinWatson is producing a siginficant z value below. This tell us that the residuals are autocorrelated.

```
summary(lm1)
```

```
##
## Call:
## lm(formula = df[, 2] ~ df[, 1])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.2139 -2.0642  0.0954  2.2743  6.5750
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  53.4269     0.1633 327.116   <2e-16 ***
## df[, 1]      -1.4460     0.1523  -9.495   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.806 on 294 degrees of freedom
## Multiple R-squared:  0.2347, Adjusted R-squared:  0.2321
## F-statistic: 90.16 on 1 and 294 DF,  p-value: < 2.2e-16
```

```
durbinWatsonTest(lm1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.9326736     0.1302252       0
##  Alternative hypothesis: rho != 0
```

```
Box.test(lm1$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm1$residuals
## X-squared = 260.1, df = 1, p-value < 2.2e-16
```

The box.test below is producing a significant p-value, and the durbinWatson is producing a siginficant z value below. This tell us that the residuals are autocorrelated.

```
summary(lm2)
```

```
## Series: df[, 2]
## Regression with ARIMA(0,0,1) errors
##
## Coefficients:
##          ma1  intercept  df[, 1]
##       0.9425    53.4383  -1.3427
## s.e.  0.0149     0.1698   0.1539
##
## sigma^2 estimated as 2.287:  log likelihood=-542.05
## AIC=1092.09   AICc=1092.23   BIC=1106.85
##
## Training set error measures:
##                         ME     RMSE      MAE        MPE     MAPE     MASE
## Training set -0.0001428604 1.504701 1.226996 -0.1498997 2.303027 2.047307
##                   ACF1
## Training set 0.8355757
```

```
durbinWatsonTest(c(lm2$residuals))
```

```
## [1] 0.3246348
```

```
Box.test(lm2$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm2$residuals
## X-squared = 208.76, df = 1, p-value < 2.2e-16
```

The box.test below is producing a significant p-value, and the durbinWatson is producing a siginficant z value below. This tell us that the residuals are autocorrelated.

```
summary(lm3)
```

```
## Series: df[, 2]
## Regression with ARIMA(1,0,0) errors
##
## Coefficients:
##          ar1  intercept  df[, 1]
##       0.9798    54.0720   0.7352
## s.e.  0.0104     1.7694   0.1250
##
## sigma^2 estimated as 0.497:  log likelihood=-316.63
## AIC=641.26   AICc=641.4   BIC=656.02
##
## Training set error measures:
##                        ME      RMSE       MAE          MPE     MAPE
## Training set 0.0003473741 0.7013957 0.5598622 -0.007866482 1.046802
##                   MASE      ACF1
## Training set 0.9341593 0.8260301
```

```
durbinWatsonTest(c(lm3$residuals))
```

```
## [1] 0.347722
```

```
Box.test(lm3$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm3$residuals
## X-squared = 204.02, df = 1, p-value < 2.2e-16
```

The box.test below is producing a significant p-value, and the durbinWatson is producing a siginficant z value below. This tell us that the residuals are autocorrelated.

```
summary(lm4)
```

```
## Series: df[, 2]
## Regression with ARIMA(0,0,2) errors
##
## Coefficients:
##          ma1     ma2  intercept  df[, 1]
##       1.6064  0.8878    53.4617  -1.0091
## s.e.  0.0293  0.0223     0.1837   0.1524
##
## sigma^2 estimated as 0.8328:  log likelihood=-393.12
## AIC=796.24   AICc=796.44   BIC=814.69
##
## Training set error measures:
##                        ME      RMSE       MAE         MPE     MAPE
## Training set 9.859198e-05 0.9064172 0.7355315 -0.08852639 1.382055
##                  MASE      ACF1
## Training set 1.227273 0.6797242
```

```
durbinWatsonTest(c(lm4$residuals))
```

```
## [1] 0.6370152
```

```
Box.test(lm4$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm4$residuals
## X-squared = 138.15, df = 1, p-value < 2.2e-16
```

The box.test below is producing a significant p-value, and the durbinWatson is producing a siginficant z value below. This tell us that the residuals are autocorrelated.

```
summary(lm5)
```

```
## Series: df[, 2]
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
##          ar1      ar2  intercept  df[, 1]
##       1.8054  -0.8451    53.5678   0.4458
## s.e.  0.0302   0.0303     0.5435   0.1114
##
## sigma^2 estimated as 0.1421:  log likelihood=-132.04
## AIC=274.09   AICc=274.29   BIC=292.54
##
## Training set error measures:
##                         ME      RMSE       MAE          MPE      MAPE
## Training set -0.0003742586 0.3744067 0.2690612 -0.003509094 0.5047478
##                   MASE      ACF1
## Training set 0.4489426 0.3230147
```

```
durbinWatsonTest(c(lm5$residuals))
```

```
## [1] 1.352616
```

```
Box.test(lm5$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm5$residuals
## X-squared = 31.198, df = 1, p-value = 2.33e-08
```

The box.test below is not producing a significant p-value, and the durbinWatson is not producing a siginficant z value below. This tell us that the residuals are not autocorrelated.

```
summary(lm6)
```

```
## Series: df[, 2]
## Regression with ARIMA(2,0,2) errors
##
## Coefficients:
##          ar1      ar2     ma1     ma2  intercept  df[, 1]
##       1.6465  -0.7037  0.4341  0.3198    53.5751   0.0816
## s.e.  0.0494   0.0492  0.0656  0.0576     0.5963   0.1111
##
## sigma^2 estimated as 0.1179:  log likelihood=-103.68
## AIC=221.36   AICc=221.75   BIC=247.2
##
## Training set error measures:
##                         ME      RMSE      MAE          MPE      MAPE
## Training set -0.0003015333 0.3398728 0.249059 -0.004567643 0.4663328
##                   MASE       ACF1
## Training set 0.4155679 0.05260454
```

```
durbinWatsonTest(c(lm6$residuals))
```

```
## [1] 1.892594
```

```
Box.test(lm6$residuals,type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  lm6$residuals
## X-squared = 0.82743, df = 1, p-value = 0.363
```

The box.test below is not producing a significant p-value, and the durbinWatson is not producing a siginficant z value below. This tell us that the residuals are not autocorrelated.

```
summary(arfima)
```

```
##
## Call:
##   arfima(y = df[, 2])
##
## *** Warning during (fdcov) fit: unable to compute correlation matrix; maybe change 'h'
##
## Coefficients:
##        Estimate
## d         0.404
## ar.ar1    0.268
## ar.ar2    0.568
## ar.ar3    0.452
## ar.ar4   -0.699
## ma.ma1   -1.430
## ma.ma2   -1.000
## sigma[eps] = 0.3283512
## [d.tol = 0.0001221, M = 100, h = 9.708e-07]
## Log likelihood: -91.13 ==> AIC = 198.2652 [8 deg.freedom]
```

```
durbinWatsonTest(c(arfima$residuals))
```

```
## [1] 1.847561
```

```
Box.test(arfima$residuals, type = c("Ljung-Box"))
```

```
##
##  Box-Ljung test
##
## data:  arfima$residuals
## X-squared = 1.6392, df = 1, p-value = 0.2004
```

9.Based on ACF plots and test results, which ARIMA model gives the best result in terms of residuals being close to white noise ?

The linear regression model with order ar=2 and ma=2 produces the best model out of all the initial options. It is also the only model to have insignificant p-values for both the Box-Ljung test and the Durbin Watson test. There is a potential debate for the best model between the (2,0,2)arima model and the arfima model. The arfima model also produced non-signifcant values for both tests meaning that the residuals produced were not autocorrelated. The (2,0,2) arima model had higher p values, and none of the lags that are used with (2,0,2) are significant with regards to autocorrelation. If 4 lags are used for the arfima model as is advised due to (4,.4,2)arfima, two lags would be considered significant with regards to autocorrelation.

```
aics <- c(AIC(lm1),lm2$aic,lm3$aic,lm4$aic,lm5$aic,lm6$aic, AIC(arfima))
durbin <- c(2*pnorm(durbinWatsonTest(c(lm1$residuals))), 2*pnorm(durbinWatsonTest(c(lm2$residuals))), 2*pnor
m(durbinWatsonTest(c(lm3$residuals))), 2*pnorm(durbinWatsonTest(c(lm4$residuals))), 2*pnorm(durbinWatsonTest
(c(lm5$residuals))),2*pnorm(durbinWatsonTest(c(lm6$residuals))), 2*pnorm(durbinWatsonTest(c(arfima$residuals
))))
Ljung <- c(Box.test(lm1$residuals, type = c("Ljung-Box"))[3], Box.test(lm2$residuals, type = c("Ljung-Box"))
[3],Box.test(lm3$residuals, type = c("Ljung-Box"))[3],Box.test(lm4$residuals, type = c("Ljung-Box"))[3],Box.
test(lm5$residuals, type = c("Ljung-Box"))[3],Box.test(lm6$residuals, type = c("Ljung-Box"))[3],Box.test(arf
ima$residuals, type = c("Ljung-Box"))[3])
cbind("AIC" = aics,"Durbin Watson Test"=durbin,"Ljung-Box Test" = Ljung)
```

```
##         AIC      Durbin Watson Test Ljung-Box Test
## p.value 1454.814 1.103612           0
## p.value 1092.091 1.254543           0
## p.value 641.2606 1.271951           0
## p.value 796.2365 1.475885           0
## p.value 274.0857 1.823822           2.329734e-08
## p.value 221.363  1.941588           0.3630161
## p.value 198.2652 1.935334           0.2004331
```

```
df_final <- cbind("AIC" = aics,"Durbin Watson Test"=durbin,"Ljung-Box Test" = Ljung)
rownames(df_final) <- c('lm1','lm2','lm3','lm4','lm5','lm6','arfima')
df_final
```

```
##        AIC      Durbin Watson Test Ljung-Box Test
## lm1    1454.814 1.103612           0
## lm2    1092.091 1.254543           0
## lm3    641.2606 1.271951           0
## lm4    796.2365 1.475885           0
## lm5    274.0857 1.823822           2.329734e-08
## lm6    221.363  1.941588           0.3630161
## arfima 198.2652 1.935334           0.2004331
```