

Model Building 4

Lian Morales

4/26/2021

```
# pulling the data from the Los Angeles County GitHub
casedata <- read.csv(text = getURL("https://raw.githubusercontent.com/datadesk/california-coronavirus-d
  filter(county == "Los Angeles") %>%
  mutate(date = date(date), month = month(date)) %>%
  map_df(rev) %>%
  filter(!is.na(new_confirmed_cases) & between(date, date("2020-04-01"), date("2021-03-31")))

# creating the time series
case.ts <- ts(casedata$new_confirmed_cases, start = 1, frequency = 1)

# averaging dec 25th and 26th
case.ts[269] <- 14711
case.ts[270] <- 14712

## april 1 - sep 30
case.ts.1 <- ts(case.ts[1:183], start = 1, frequency = 1)

# april 1 - jul 14
case.ts.1.1 <- ts(case.ts[1:105], start = 1, frequency = 1)
# jul 15 - sep 30
case.ts.1.2 <- ts(case.ts[106:183], start = 1, frequency = 1)

## oct 1 - mar 31
case.ts.2 <- ts(case.ts[184:365], start = 1, frequency = 1)

# oct 1 - dec 20
case.ts.2.1 <- ts(case.ts[184:264], start = 1, frequency = 1)
# dec 21 - mar 31
case.ts.2.2 <- ts(case.ts[265:365], start = 1, frequency = 1)
```

December 21 - March 31

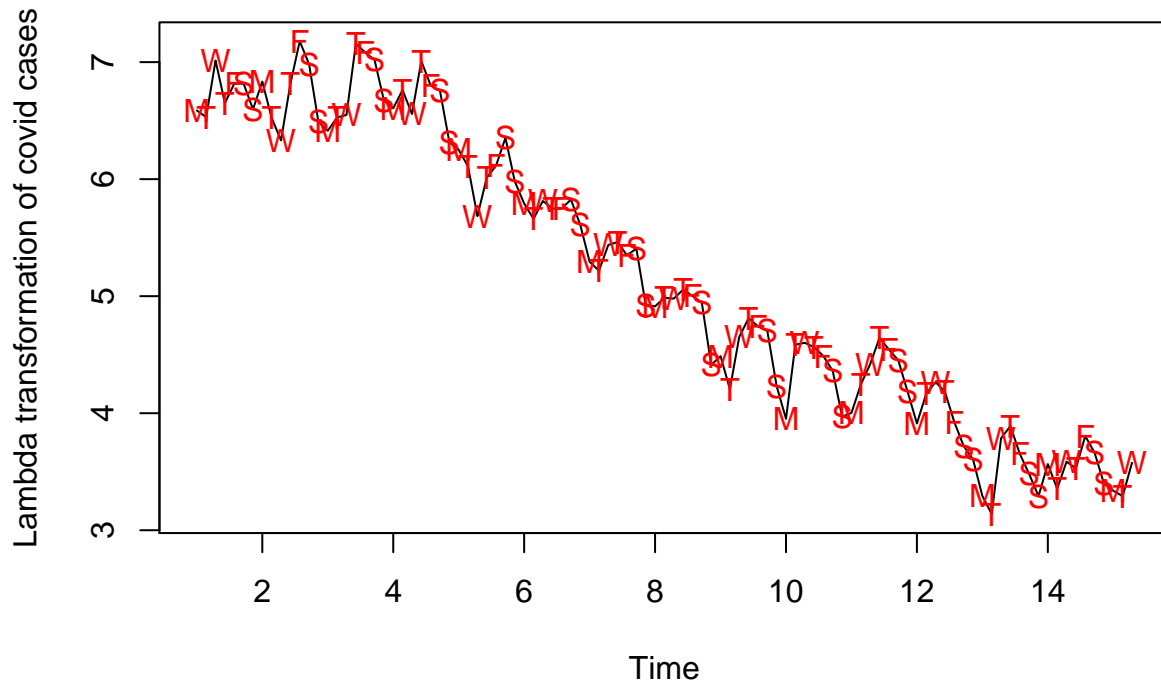
ARIMA(3,1,2) and ARIMA(0,1,3) with drift

```
# transform data October 1 - December 20 to lambda + add seasonality
case.ts.2.2_seas.trans <- ts(case.ts.2.2^0.2, frequency = 7)
case.ts.2.2_trans <- ts(case.ts.2.2^0.2)

##### SEASONAL
```

```
# plot ts of transformed seasonal data
week. <- season(case.ts.2.2_seas.trans)
plot(case.ts.2.2_seas.trans, ylab = "Lambda transformation of covid cases", xlab = "Time", main = "Trans")
points(case.ts.2.2_seas.trans, pch = as.vector(week.), col = "red")
```

Transformation by Lambda = 0.3

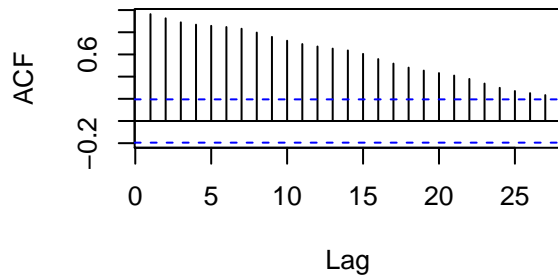


```
par(mfrow = c(2,2))
# acf of transformed data
acf(as.vector(case.ts.2.2_seas.trans), lag.max = 27, main = "ACF of Transformation by Lambda = 0.2")
# pacf of transformed data
pacf(as.vector(case.ts.2.2_seas.trans), lag.max = 27, main = "Transformation by Lambda = 0.2")
# eacf of transformed data
eacf(as.vector(case.ts.2.2_seas.trans)) # seems to show a ARMA(1,1)?
```

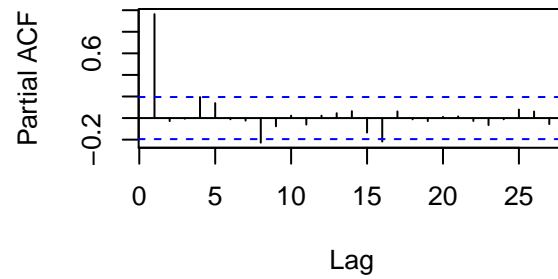
[illegible]

```
# best subsets ARMA approach
par(mfrow = c(1,1))
```

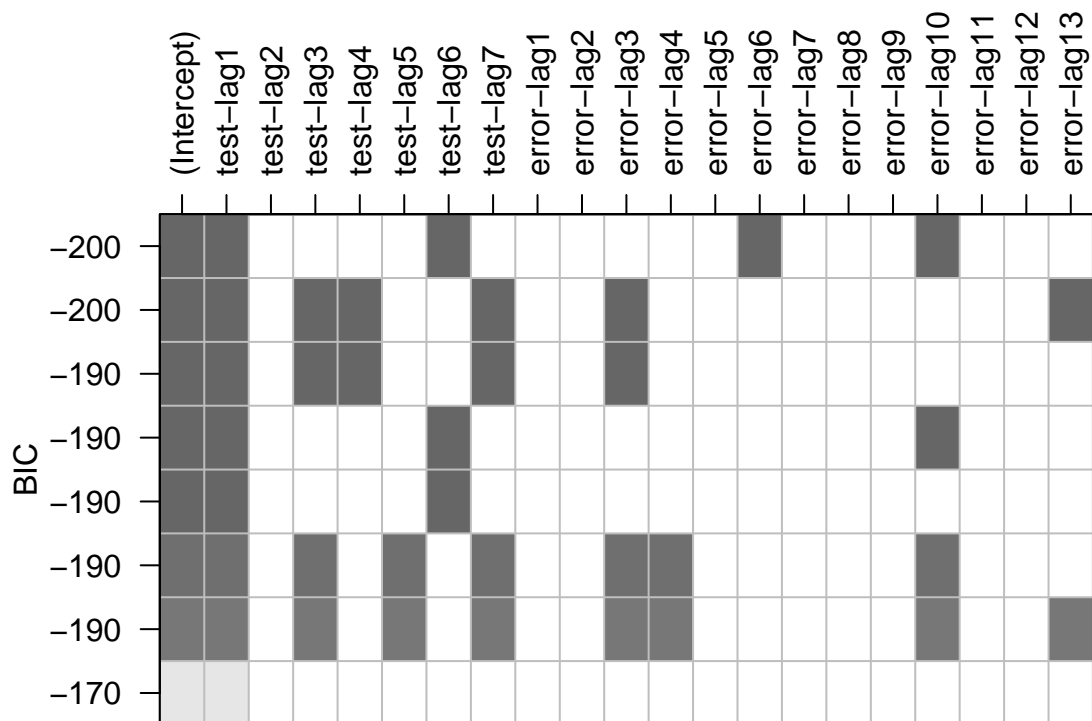
ACF of Transformation by Lambda = 0.



Transformation by Lambda = 0.2



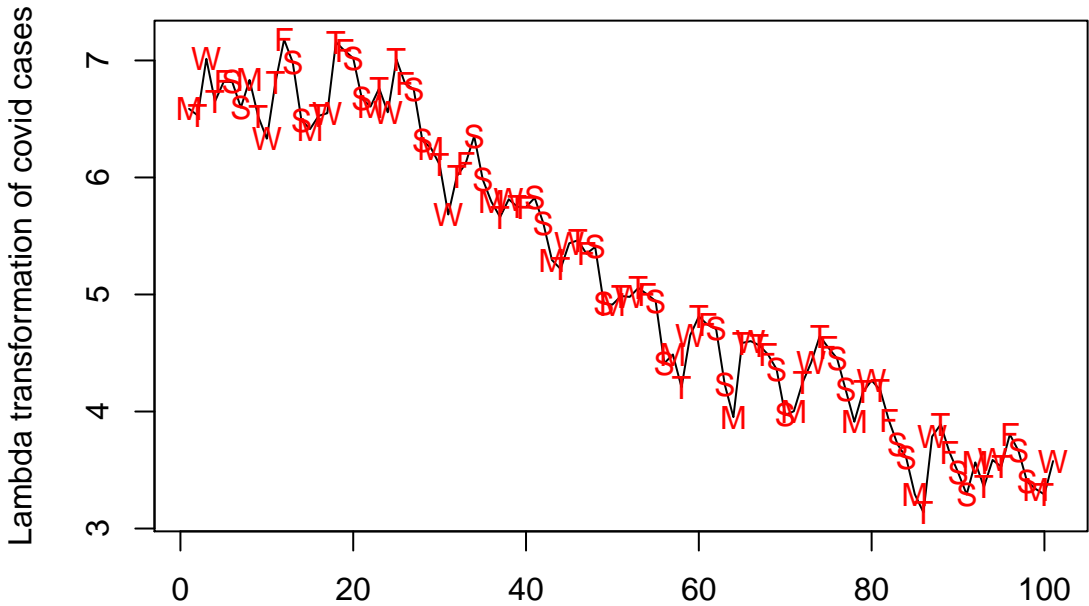
```
res = armasubsets(y=case.ts.2.2_seas.trans, nar= 7, nma = 13, y.name = "test", ar.method = 'ols')
plot(res) # not sure how to analyze this
```



TRANSFORMATION (no seasonality)

```
par(mfrow = c(1,1))
plot(case.ts.2.2_trans, ylab = "Lambda transformation of covid cases", xlab = "Time", main = "Transformation of covid cases")
points(case.ts.2.2_trans, pch = as.vector(week.), col = "red")
```

Transformation by Lambda = 0.2

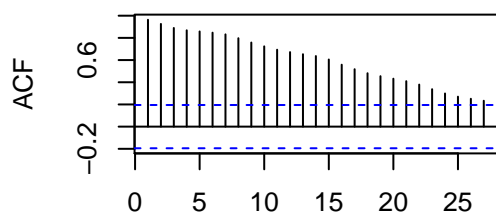


```
par(mfrow = c(2,2))
# acf of transformed data
acf(as.vector(case.ts.2.2_trans), lag.max = 27, main = "ACF of Transformation by Lambda = 0.2")
# pacf of transformed data
pacf(as.vector(case.ts.2.2_trans), lag.max = 27, main = "Transformation by Lambda = 0.2")
# eacf of transformed data
eacf(as.vector(case.ts.2.2_trans)) # seems to show a ARMA(1,1)?
```

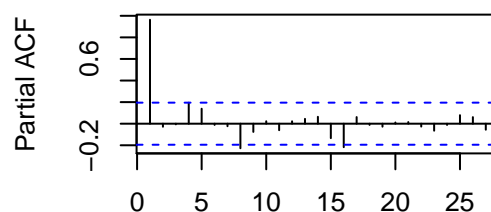
[illegible]

```
# best subsets ARMA approach
par(mfrow = c(1,1))
```

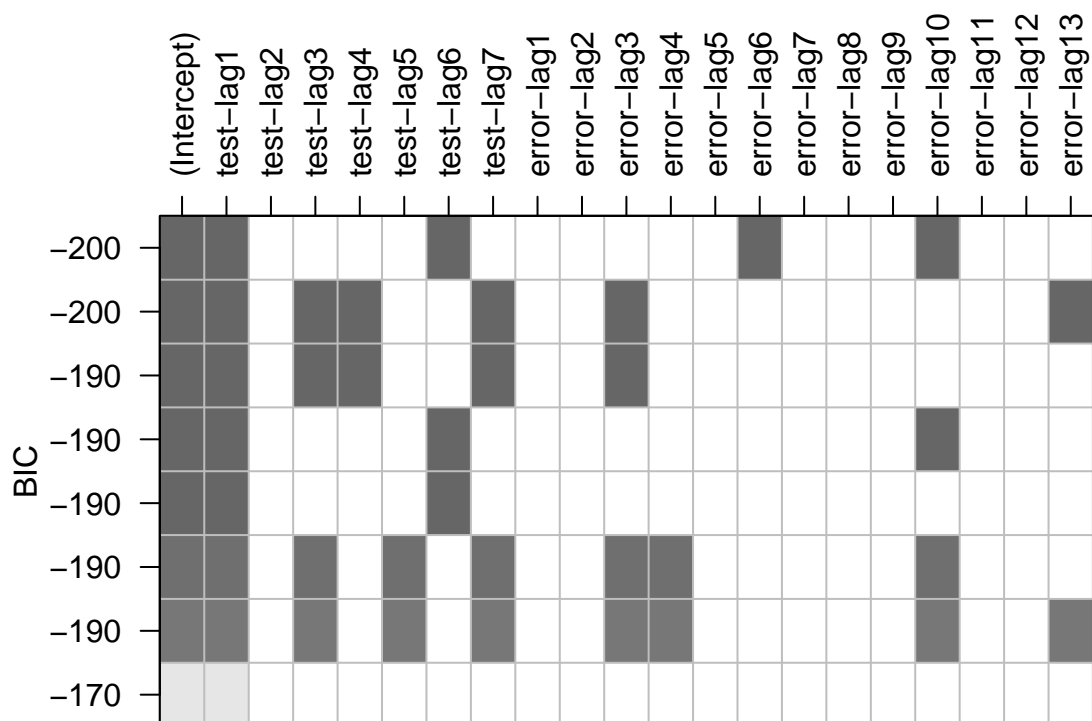
ACF of Transformation by Lambda = 0.



Transformation by Lambda = 0.2



```
res = armasubsets(y=case.ts.2.2_trans, nar= 7, nma = 13, y.name = "test", ar.method = 'ols')
plot(res) # not sure how to analyze this
```



```
# simulated model prediction
auto.arima(case.ts.2.2_seas.trans)
```

```
## Series: case.ts.2.2_seas.trans
## ARIMA(3,1,2) with drift
##
## Coefficients:
##      ar1      ar2      ar3      ma1      ma2      drift
##      0.9620 -0.5989 -0.2510 -1.3706  0.8304 -0.0322
## s.e.  0.1253  0.1356  0.1178  0.0989  0.0696  0.0104
##
## sigma^2 estimated as 0.04263: log likelihood=17.74
## AIC=-21.49  AICc=-20.27  BIC=-3.25
```

```
auto.arima(case.ts.2.2_trans)
```

```
## Series: case.ts.2.2_trans
## ARIMA(0,1,3) with drift
##
## Coefficients:
##          ma1          ma2          ma3          drift
##      -0.2309  -0.1892  -0.2612  -0.0344
## s.e.   0.0952   0.1070   0.1048   0.0078
##
## sigma^2 estimated as 0.05548:  log likelihood=4.46
## AIC=1.08   AICc=1.72   BIC=14.1
```

Completing analysis with suggested models

```
# simulate aimra models for seasonal and transformed data
arima.seas_trans.2.2 <- arima(case.ts.2.2_seas.trans,
                              order = c(3,1,2))
```

```
## Warning in log(s2): NaNs produced
```

```
arima.trans.2.2 <- arima(case.ts.2.2_trans, order = c(0,1,3))
```

```
# residual tests
```

```
# check for correlation of error terms
```

```
Box.test(rstandard(arima.seas_trans.2.2), type = "Ljung-Box")
```

```
##
```

```
## Box-Ljung test
```

```
##
```

```
## data:  rstandard(arima.seas_trans.2.2)
```

```
## X-squared = 0.45794, df = 1, p-value = 0.4986
```

```
Box.test(rstandard(arima.trans.2.2), type = "Ljung-Box")
```

```
##
```

```
## Box-Ljung test
```

```
##
```

```
## data:  rstandard(arima.trans.2.2)
```

```
## X-squared = 0.35416, df = 1, p-value = 0.5518
```

```
# check for independence of error terms
```

```
runs(rstandard(arima.seas_trans.2.2))
```

```
## $pvalue
```

```
## [1] 0.81
```

```
##
```

```
## $observed.runs
## [1] 51
##
## $expected.runs
## [1] 49.31683
##
## $n1
## [1] 61
##
## $n2
## [1] 40
##
## $k
## [1] 0
```

```
runs(rstandard(arima.trans.2.2))
```

```
## $pvalue
## [1] 0.325
##
## $observed.runs
## [1] 53
##
## $expected.runs
## [1] 47.89109
##
## $n1
## [1] 64
##
## $n2
## [1] 37
##
## $k
## [1] 0
```

```
# check for normality of error terms
shapiro.test(rstandard(arima.seas_trans.2.2))
```

```
##
## Shapiro-Wilk normality test
##
## data:  rstandard(arima.seas_trans.2.2)
## W = 0.9867, p-value = 0.4103
```

```
shapiro.test(rstandard(arima.trans.2.2))
```

```
##
## Shapiro-Wilk normality test
##
## data:  rstandard(arima.trans.2.2)
## W = 0.97873, p-value = 0.1024
```

Ljung-Box

Seasonal Transformed Model

For the seasonal transformed data, the residuals from the Ljung-Box test have a high p-value of 0.4986, which indicate that the residuals are uncorrelated.

Transformed Model

Similarly, in the transformed data with no seasonality, the p-value is very high, 0.5518, therefore we do not have evidence against the null hypothesis that the error terms are uncorrelated.

Runs Test

Seasonal Transformed Model

For the runs test, we notice that the seasonal transformed model has observed runs of 51 versus expected runs of 49.31683. The p-value of this test is large, equalling 0.81, which means we do not have statistically significant evidence against independence of the error terms in the model.

Transformed Model

For the transformed model, the model has an observed runs of 53 and expected of 47.89109. Similar to the seasonal model, this runs test has a high p-value of 0.325. Although this is not as high as the seasonal model, it still indicated that we do not have statistically significant evidence against independence of the error terms in the model.

Shapiro Wilks Test

Seasonal Transformed Model

For the seasonal model, we notice that the p-value for this test is large, with 0.4103, indicating that we do not have evidence against the null hypothesis that the data are normal.

Transformed Model

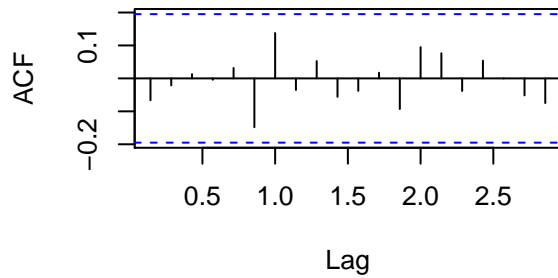
Similar to the seasonal model, the transformed data have a p-value of 0.1024 indicating that we do not have evidence against the null hypothesis that the data are normal.

Visualize residuals

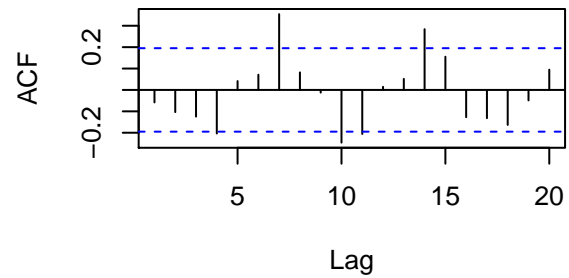
```
par(mfrow = c(2,2))
# acf of seasonal and transformed residuals
acf(rstandard(arima.seas_trans.2.2))
acf(rstandard(arima.trans.2.2))

# pacf of seasonal and transformed residuals
pacf(rstandard(arima.seas_trans.2.2))
pacf(rstandard(arima.trans.2.2))
```

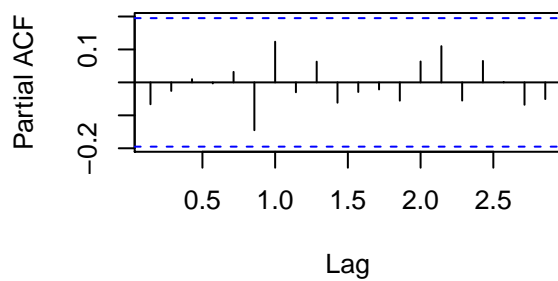

Series rstandard(arima.seas_trans.2.2)



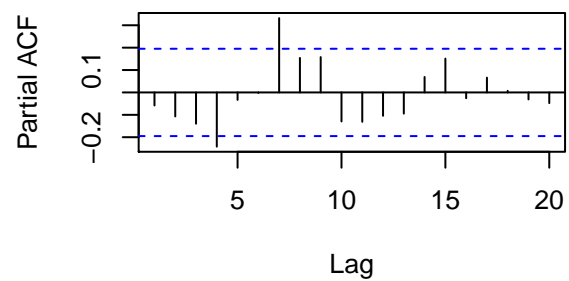
Series rstandard(arima.trans.2.2)



Series rstandard(arima.seas_trans.2.2)

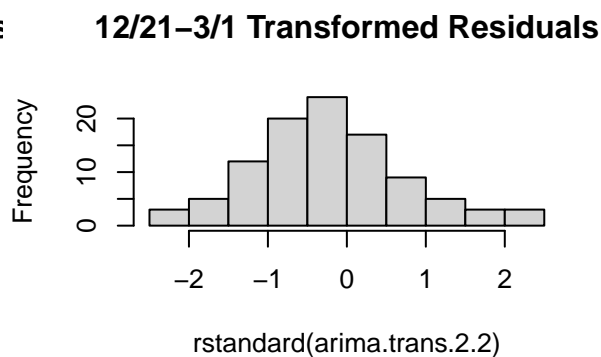
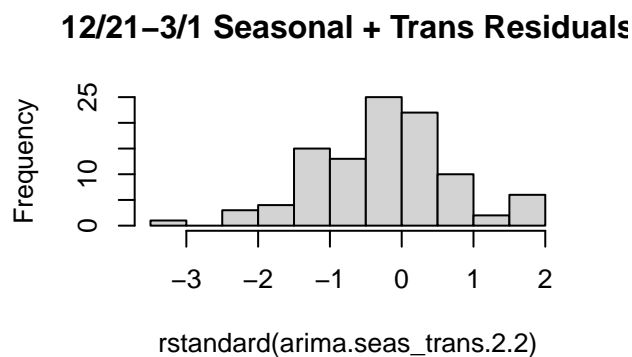
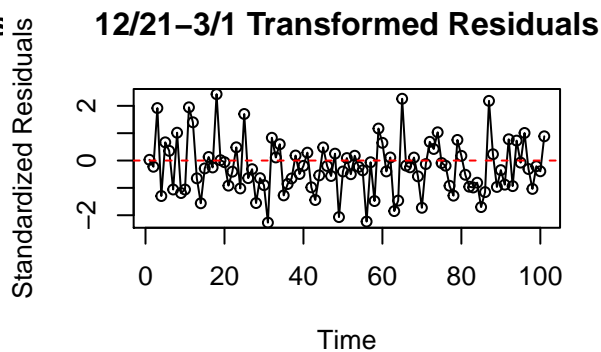
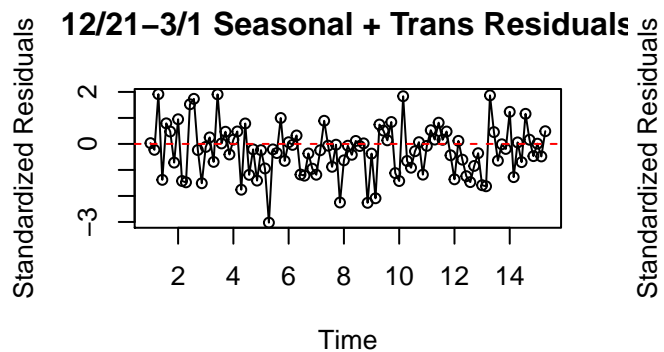


Series rstandard(arima.trans.2.2)



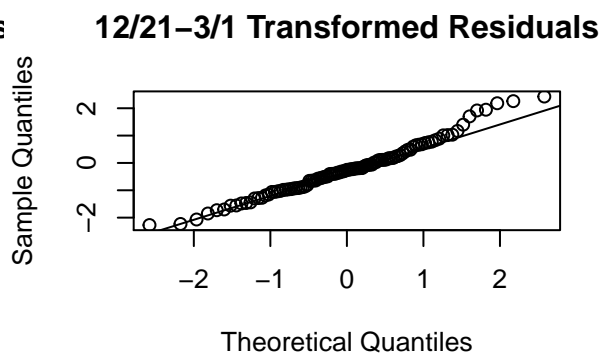
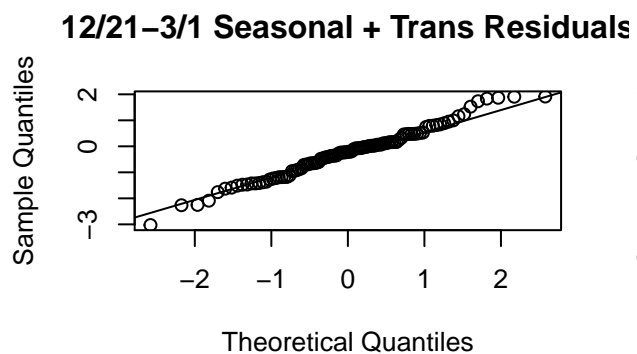
```
# plot residuals
plot(rstandard(arima.seas_trans.2.2),
     type = "o",
     main = "12/21-3/1 Seasonal + Trans Residuals",
     ylab = "Standardized Residuals")
abline(h=0,lty=2,col="red")
plot(rstandard(arima.trans.2.2),
     type = "o",
     main = "12/21-3/1 Transformed Residuals",
     ylab = "Standardized Residuals")
abline(h=0,lty=2,col="red")

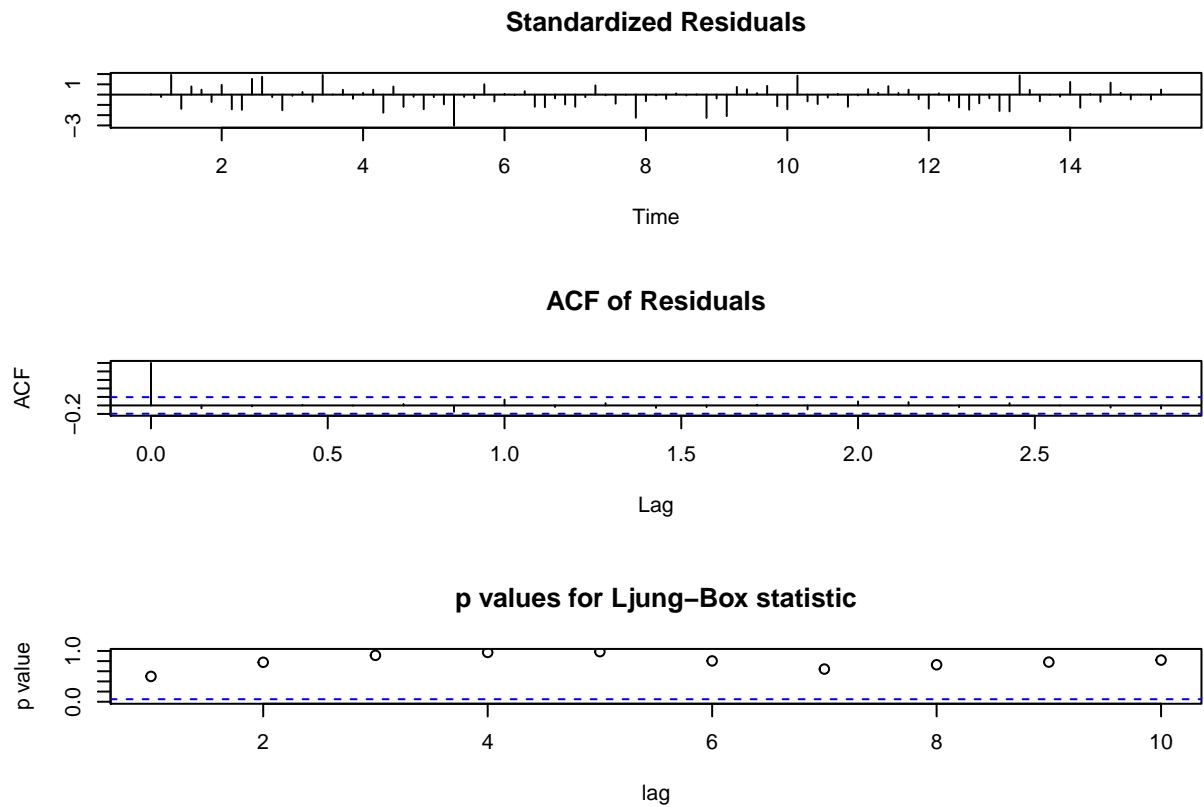
# Histogram of residuals
hist(rstandard(arima.seas_trans.2.2),
     main = "12/21-3/1 Seasonal + Trans Residuals")
hist(rstandard(arima.trans.2.2),
     main = "12/21-3/1 Transformed Residuals")
```



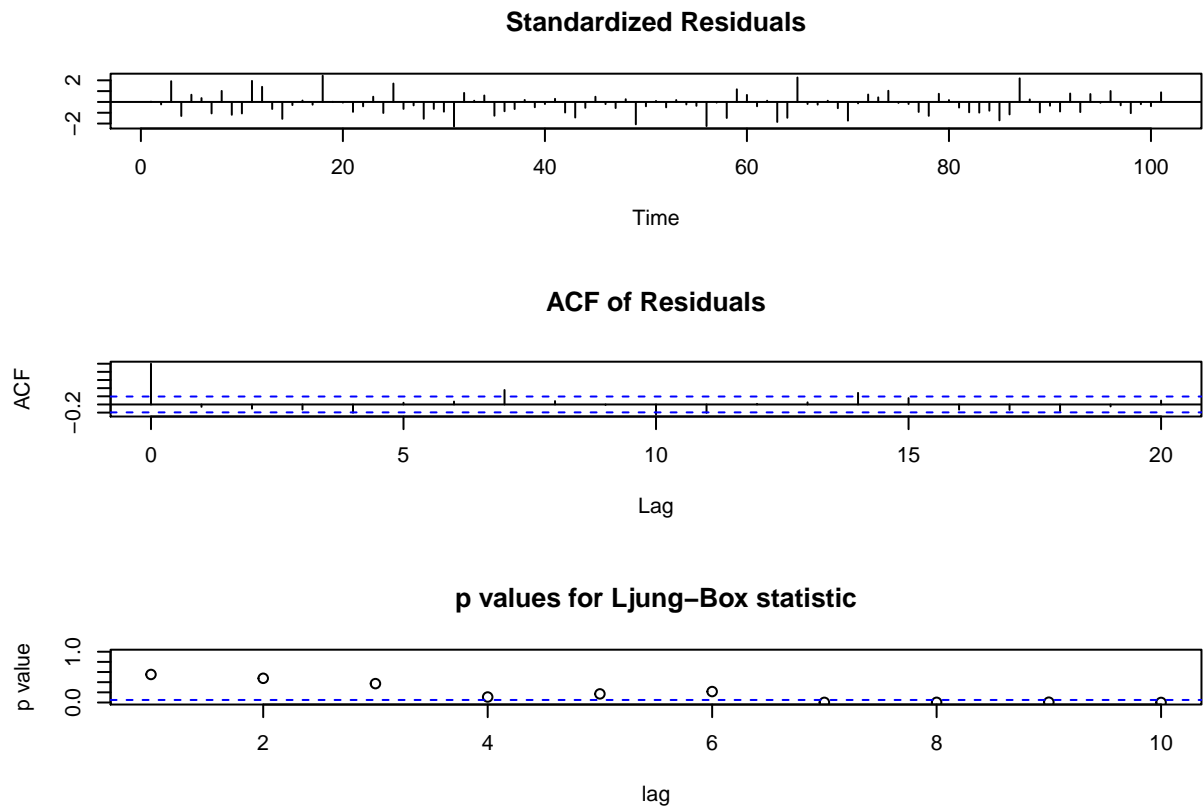
```
# qqnorm plots of residuals
qqnorm(rstandard(arima.seas_trans.2.2),
       main = "12/21-3/1 Seasonal + Trans Residuals")
qqline(rstandard(arima.seas_trans.2.2))
qqnorm(rstandard(arima.trans.2.2),
       main = "12/21-3/1 Transformed Residuals")
qqline(rstandard(arima.trans.2.2))

# Diagnostic Plot
tsdiag(arima.seas_trans.2.2)
```





```
tsdiag(arima.trans.2.2)
```



We notice in the residual visualization, they are very similar, and do not show strong signs of correlation.

We can see this in the acf and pacfs, where they show a zero mean white noise process. However, in the transformed non seasonal data we see that the pacf is showing pattern indicating that we will need to use the ARIMA(3,1,2) model. In the plot of the residuals, we see that they are generally centered around zero, however there are some outliers that are beyond 3, which we should look out for in our analysis. Both histograms are centered around zero, however we notice in the seasonal model some outliers. In the normality plots, we notice that there is a general linear trend for both data sets, however, the tail ends of data points are very far away from the line, which indicate the data is not normal. In the `tsdiag()`, Diagnostic Plots for Time-Series Fits function it is further reiterating our points of zero mean, white noise, independent process of residuals that are not exactly normal. In choosing one transformation over the other, we will go for the ARIMA(3,1,2) model.