Model Building 3

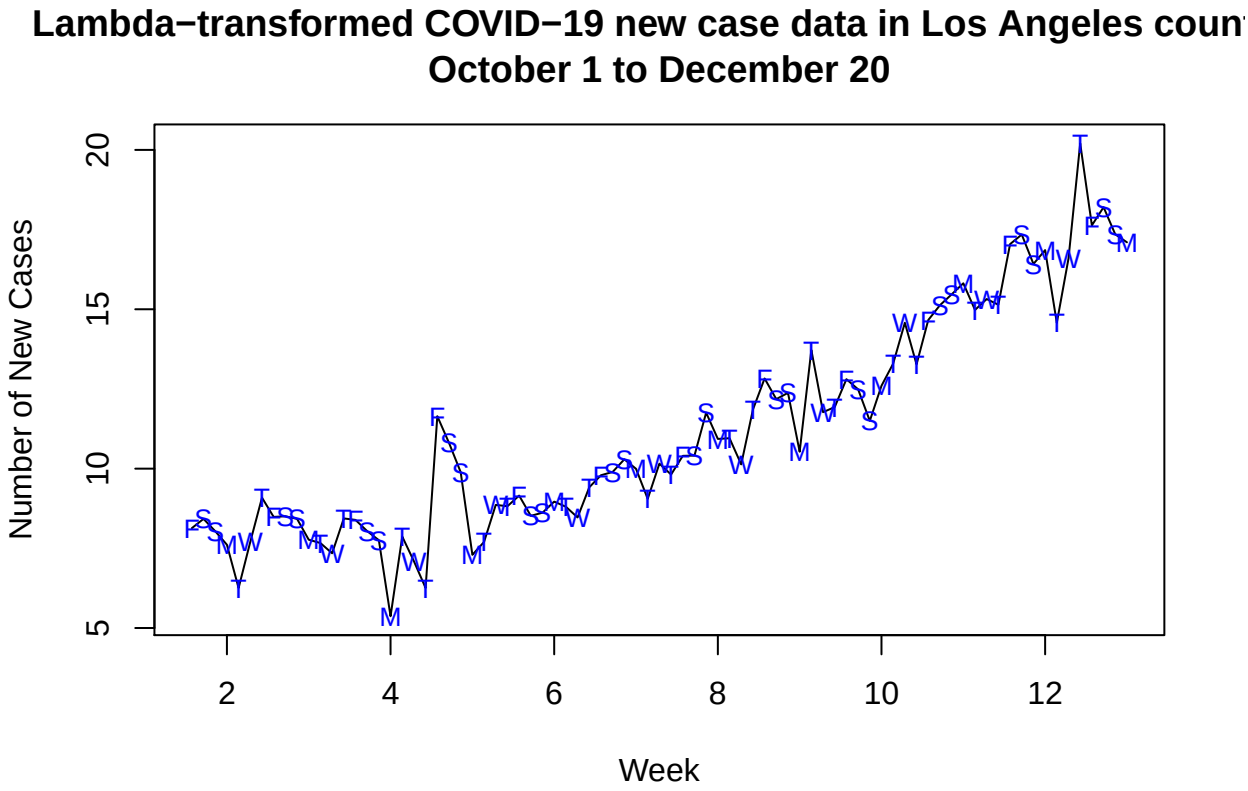
Lian Morales

4/26/2021

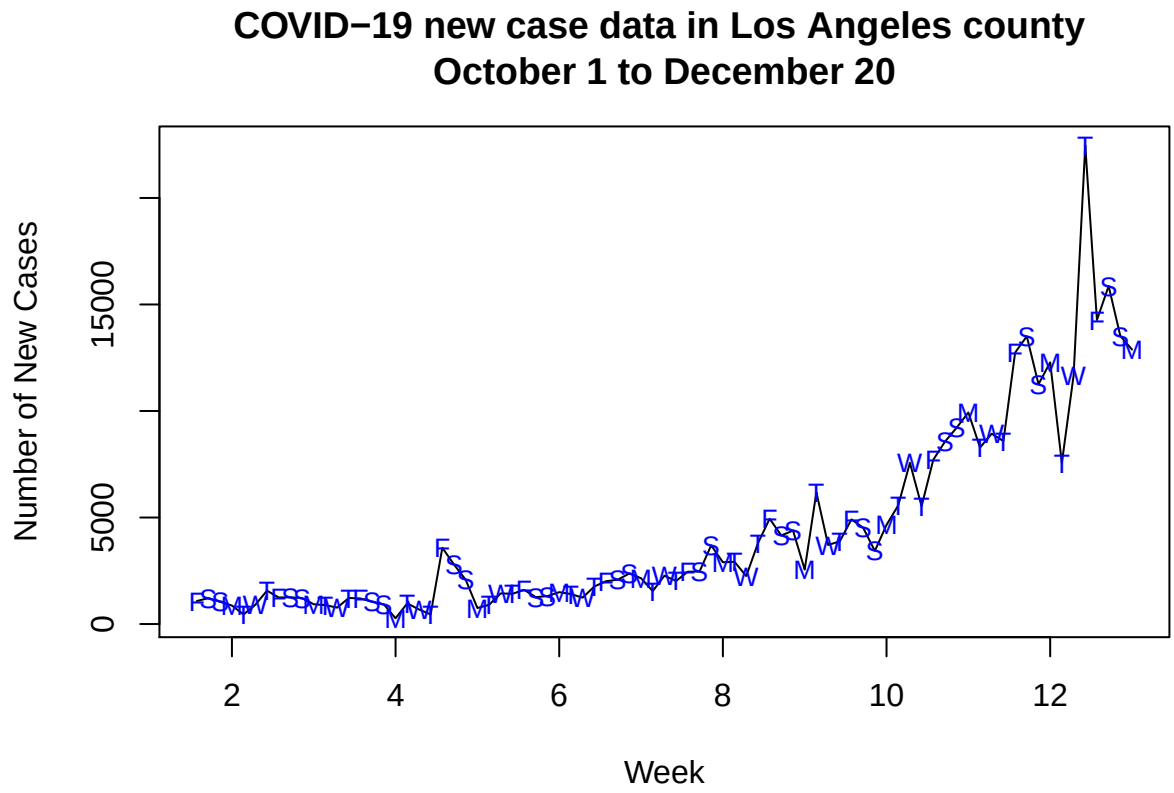
```
# pulling the data from the Los Angeles County GitHub
casedata <- read.csv(text = getURL("https://raw.githubusercontent.com/datadesk/california-coronavirus-da
  filter(county == "Los Angeles") %>%
  mutate(date = date(date), month = month(date)) %>%
  map_df(rev) %>%
  filter(!is.na(new_confirmed_cases)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

# oct 1 - dec 20
case.ts.2.1 <- ts(case.ts[184:264], start = 1, frequency = 1)
```

October 1 - December 20

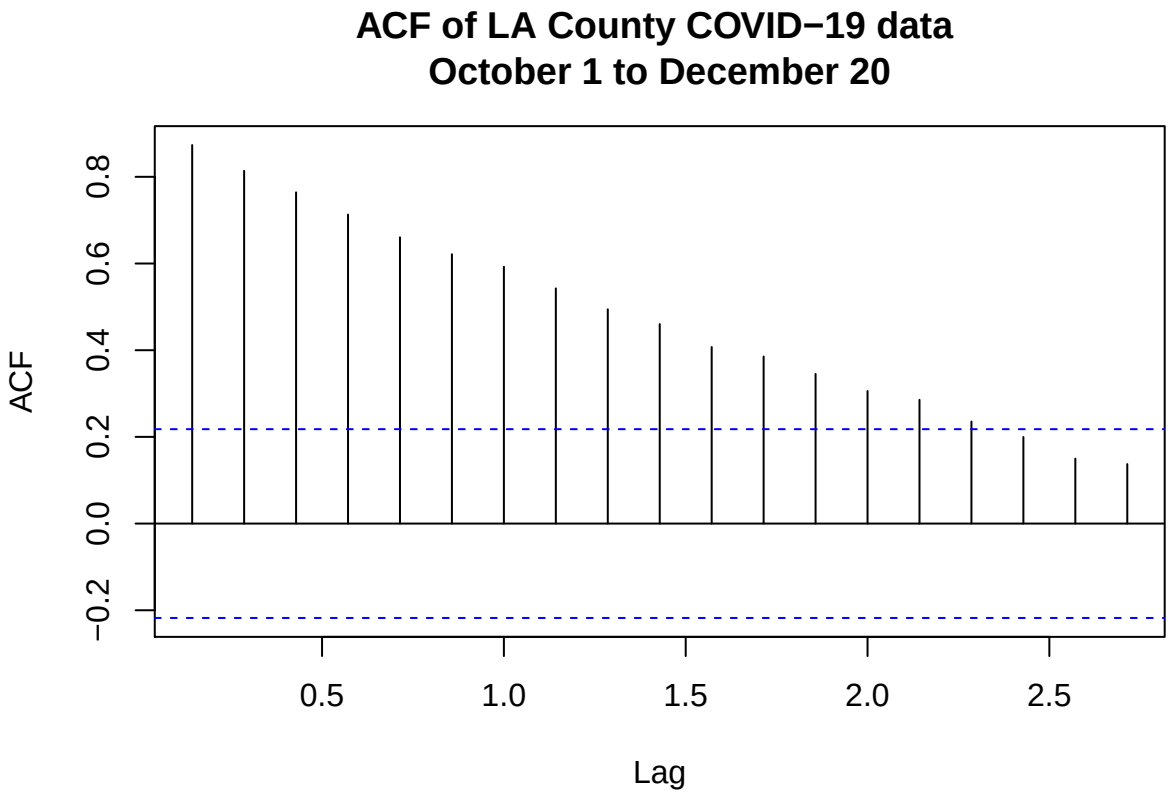
```
set.seed(13)
# transform data October 1 - December 20 to lambda + add seasonality
trans.seasonal.ts.2.1  <- ts(case.ts.2.1^0.3, frequency = 7, start = c(1,5))
seasonal.ts.2.1  <- ts(case.ts.2.1, frequency = 7, start = c(1,5))
trans.ts.2.1 <- ts(case.ts.2.1^0.3)

week. <- season(trans.seasonal.ts.2.1)
plot(trans.seasonal.ts.2.1, ylab = "Number of New Cases", xlab = "Week", main = "Lambda-transformed COV
points(trans.seasonal.ts.2.1, pch = as.vector(week.), col = "blue", cex = 0.8)
```



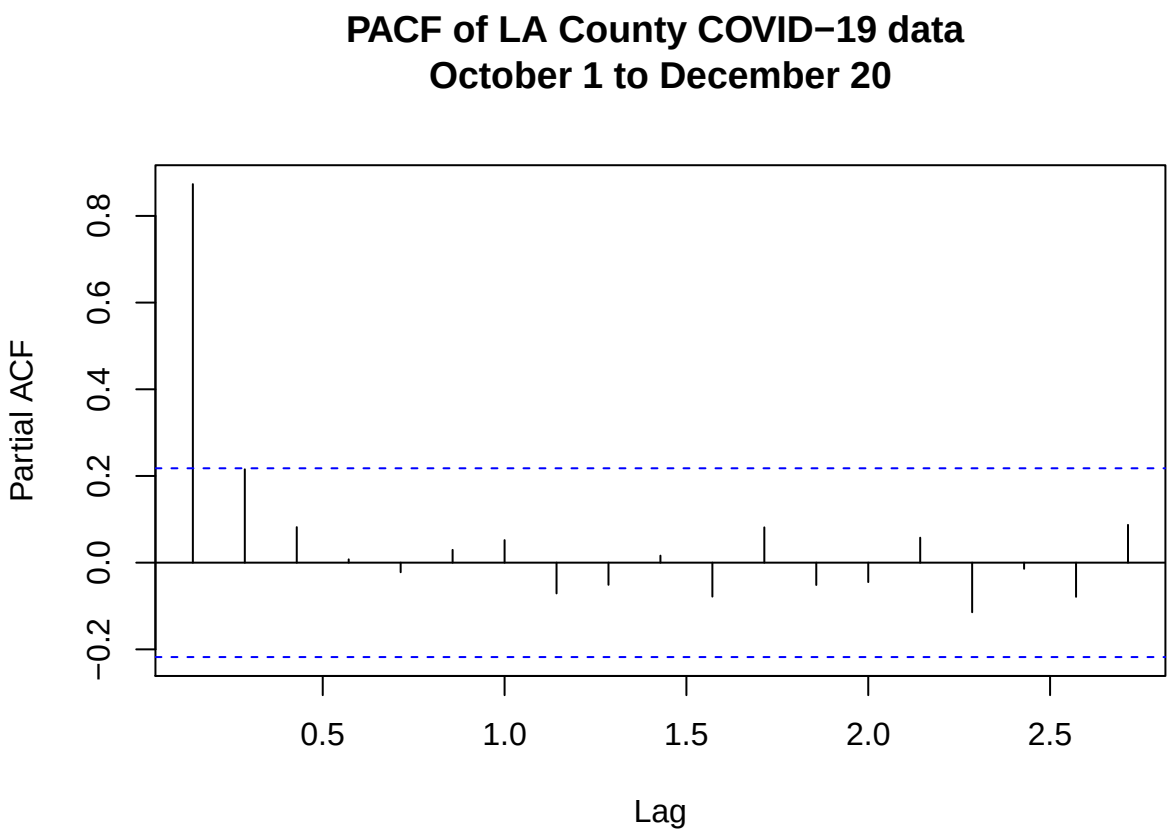

```
# plot ts of seasonal data
week. <- season(seasonal.ts.2.1)
plot(seasonal.ts.2.1, ylab = "Number of New Cases", xlab = "Week", main = "COVID-19 new case data in Los
points(seasonal.ts.2.1, pch = as.vector(week.), col = "blue", cex = 0.8)
```



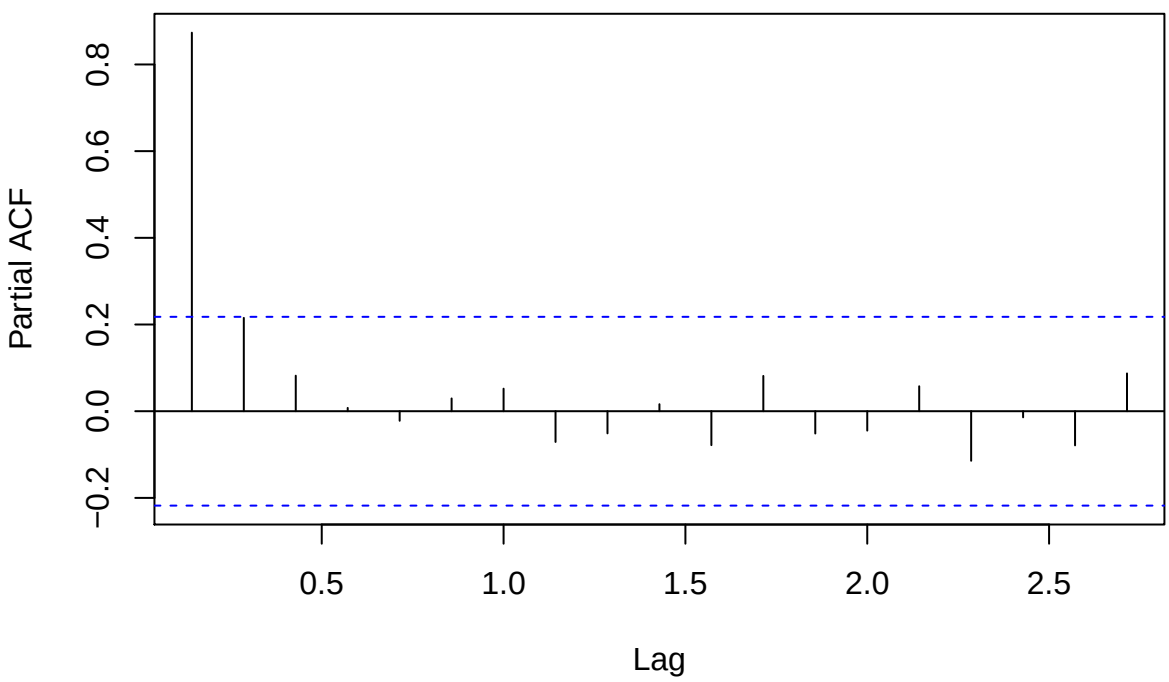

```
# acf of transformed data
acf(seasonal.ts.2.1, main = "ACF of LA County COVID-19 data \nOctober 1 to December 20")
```



```r
# pacf of transformed data
p<-pacf(seasonal.ts.2.1, main = "PACF of LA County COVID-19 data \nOctober 1 to December 20", cex.main =
```




```r
plot(p)
```

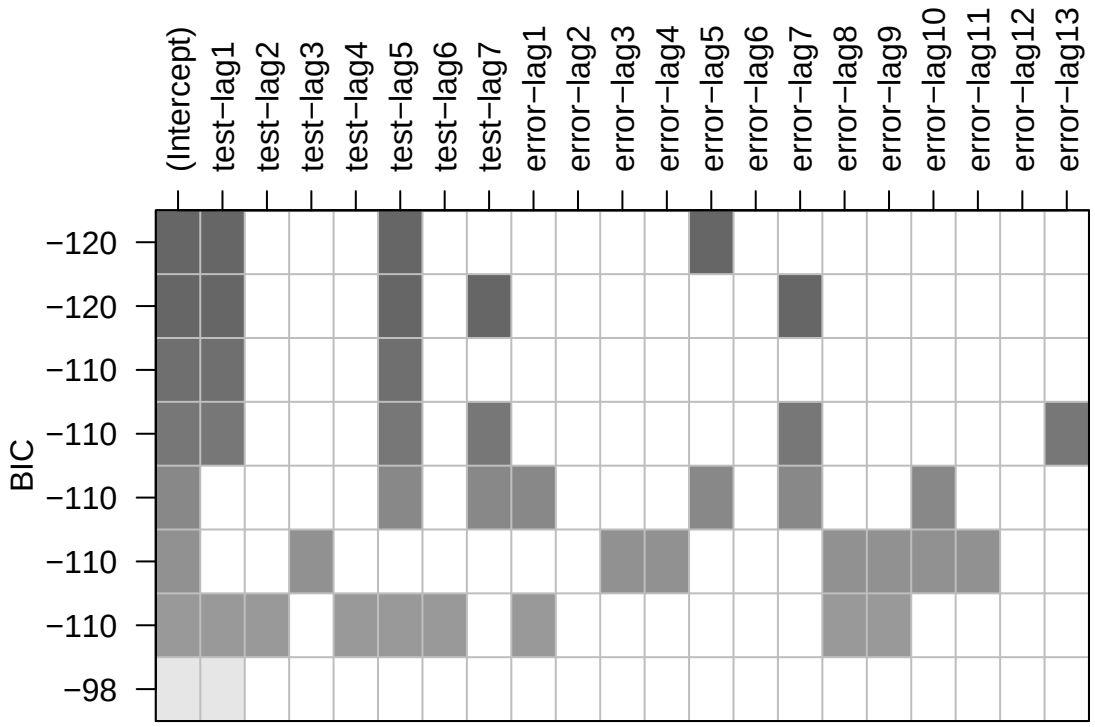


In this plot there is not much of a clear trend of seasonality, however we do notice that on Wednesday there usually tend to have higher values while on Monday, Tuesday tend to have lower values. Therefore it is important to explore seasonality in this model.

```
# eacf of transformed data
eacf(trans.ts.2.1)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x x x x x x x  x  x  x
## 1 x o o o o o o o o o o  o  o  o
## 2 x o o o o o o o o o o  o  o  o
## 3 x o o x o o o o o o o  o  o  o
## 4 x x x x o o o o o o o  o  o  o
## 5 o x o x o o o o o o o  o  o  o
## 6 o x o x o o o o o o o  o  o  o
## 7 x o o x o o o o o o o  o  o  o
```

```
# eacf of transformed data with seasonality
plot(armasubsets(y=trans.seasonal.ts.2.1, nar= 7, nma = 13, y.name = "test", ar.method = 'ols'))
```




```
# simulated model prediction
auto.arima(trans.seasonal.ts.2.1)
```

```
## Series: trans.seasonal.ts.2.1
## ARIMA(0,1,2)(0,0,1)[7] with drift
##
## Coefficients:
##           ma1      ma2    sma1   drift
##       -0.5695  -0.2049  0.1970  0.1270
## s.e.   0.1128   0.1093  0.1245  0.0344
##
## sigma^2 estimated as 1.249:  log likelihood=-120.86
## AIC=251.73   AICc=252.54   BIC=263.64
```

```
auto.arima(trans.ts.2.1)
```

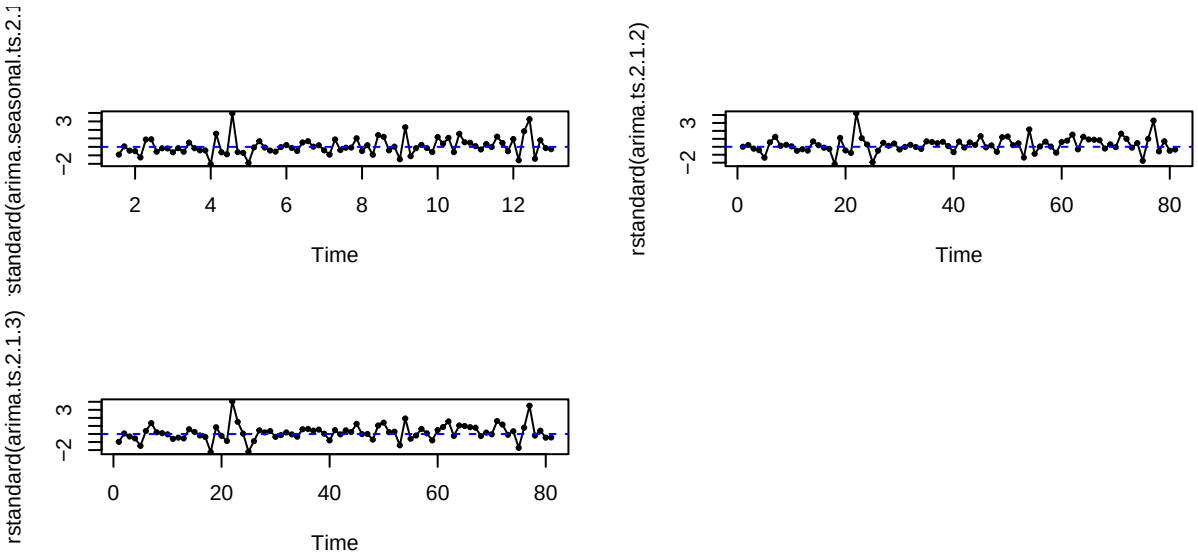
```
## Series: trans.ts.2.1
## ARIMA(0,1,2) with drift
##
## Coefficients:
##           ma1      ma2   drift
##       -0.5567  -0.1929  0.1271
## s.e.   0.1126   0.1061  0.0326
##
## sigma^2 estimated as 1.274:  log likelihood=-122.03
## AIC=252.06   AICc=252.59   BIC=261.59
```

ARIMA(0,1,2)x(0,0,1)[7] # auto.arima of trans.seasonal.ts.2.1 ARIMA(0,1,2) ARIMA(1,0,1) # eacf

Completing analysis with suggested models

```
# simualte aimra models for seasonal and transformed data

arima.seasonal.ts.2.1.1 <- arima(trans.seasonal.ts.2.1, order = c(1,0,0), seasonal = list(order = c(1,0
arima.ts.2.1.2 <- arima(trans.ts.2.1, order = c(0,1,2))
arima.ts.2.1.3 <- arima(trans.ts.2.1, order = c(1,0,1), method = "ML")
```

```
par(mfrow = c(3,2))
# plot residuals
plot(rstandard(arima.seasonal.ts.2.1.1),type = "o", pch=20)
abline(h=0,lty=2,col="blue")
plot(rstandard(arima.ts.2.1.2),type = "o", pch=20)
abline(h=0,lty=2,col="blue")
plot(rstandard(arima.ts.2.1.3),type = "o", pch=20)
abline(h=0,lty=2,col="blue")
```




```
# AIC
arima.seasonal.ts.2.1.1$aic
```

```
## [1] 276.8643
```

```
arima.ts.2.1.2$aic
```

```
## [1] 256.1096
```

```
arima.ts.2.1.3$aic
```

```
## [1] 265.5846
```

```r
# residual tests

# check for correlation of error terms
Box.test(rstandard(arima.seasonal.ts.2.1.1), type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  rstandard(arima.seasonal.ts.2.1.1)
## X-squared = 4.5964, df = 1, p-value = 0.03204
```

```r
Box.test(rstandard(arima.ts.2.1.2), type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  rstandard(arima.ts.2.1.2)
## X-squared = 0.48346, df = 1, p-value = 0.4869
```

```r
Box.test(rstandard(arima.ts.2.1.3), type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  rstandard(arima.ts.2.1.3)
## X-squared = 0.22094, df = 1, p-value = 0.6383
```

```r
# check for independence of error terms
runs(rstandard(arima.seasonal.ts.2.1.1))
```

```
## $pvalue
## [1] 0.355
##
## $observed.runs
## [1] 45
##
## $expected.runs
## [1] 40.45679
##
## $n1
## [1] 47
##
## $n2
## [1] 34
##
## $k
## [1] 0
```

```r
runs(rstandard(arima.ts.2.1.2))
```

```
## $pvalue
## [1] 0.676
##
## $observed.runs
## [1] 42
##
## $expected.runs
## [1] 39.71605
##
## $n1
## [1] 32
##
## $n2
## [1] 49
##
## $k
## [1] 0
```

```
runs(rstandard(arima.ts.2.1.3))
```

```
## $pvalue
## [1] 0.692
##
## $observed.runs
## [1] 43
##
## $expected.runs
## [1] 40.75309
##
## $n1
## [1] 35
##
## $n2
## [1] 46
##
## $k
## [1] 0
```

```
# check for normality of error terms
shapiro.test(rstandard(arima.seasonal.ts.2.1.1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(arima.seasonal.ts.2.1.1)
## W = 0.93108, p-value = 0.0003019
```

```
shapiro.test(rstandard(arima.ts.2.1.2))
```

```
##
##  Shapiro-Wilk normality test
##
```

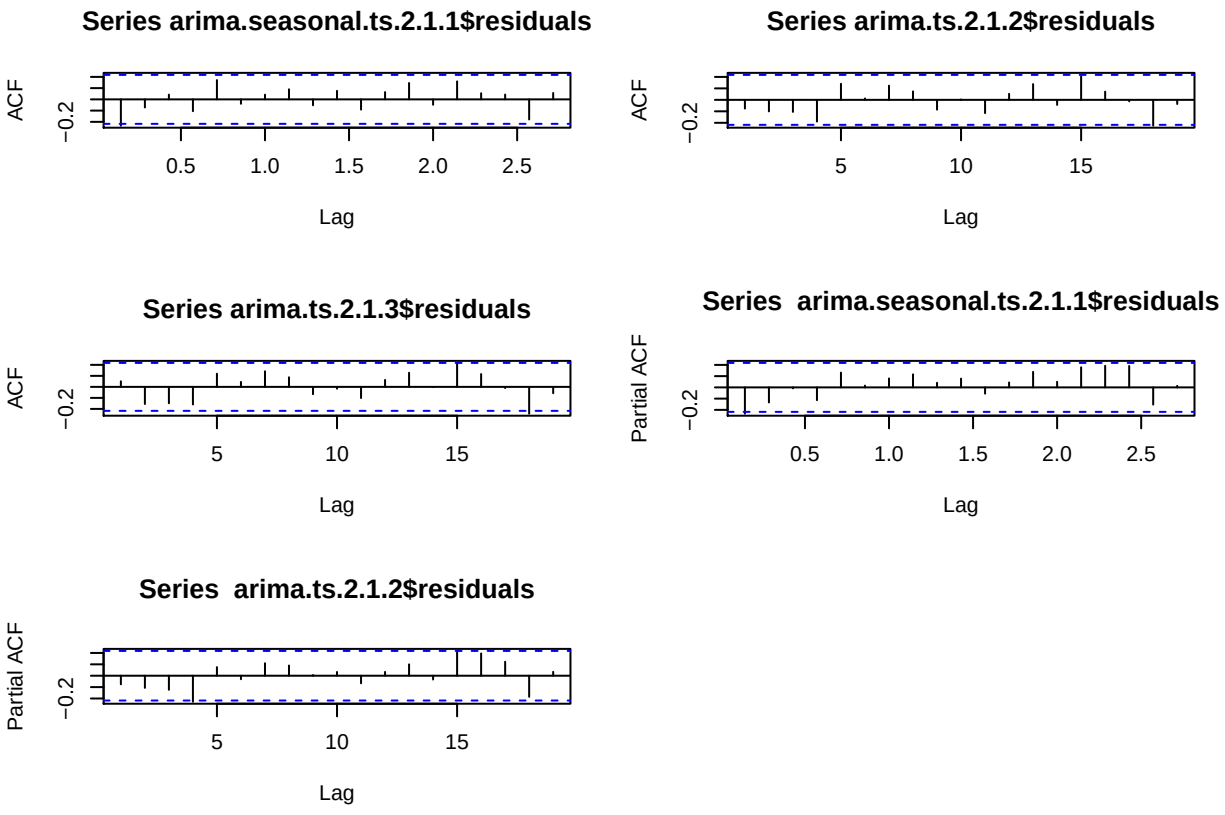
```
## data:  rstandard(arima.ts.2.1.2)
## W = 0.93084, p-value = 0.0002935
```

```
shapiro.test(rstandard(arima.ts.2.1.3))
```

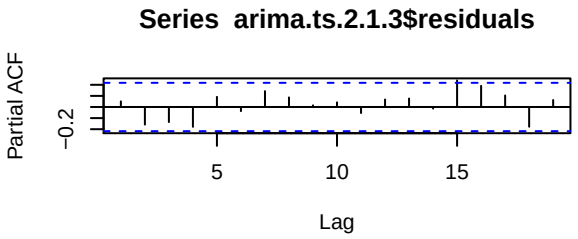
```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(arima.ts.2.1.3)
## W = 0.93131, p-value = 0.0003102
```

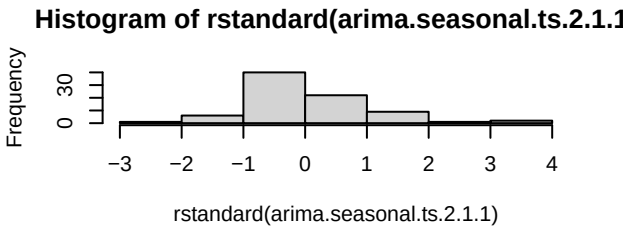
Visualize residuals

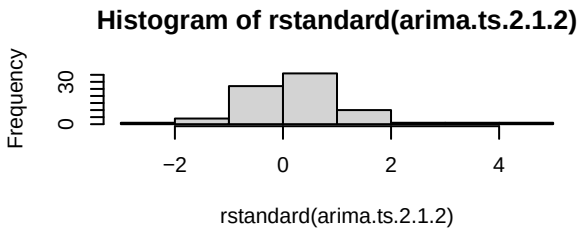
```
layout(matrix(c(1,2,3,4,5,0), nrow = 3, ncol = 2, byrow = TRUE))

acf(arima.seasonal.ts.2.1.1$residuals)
acf(arima.ts.2.1.2$residuals)
acf(arima.ts.2.1.3$residuals)

# pacf of seasonal and transformed residuals
pacf(arima.seasonal.ts.2.1.1$residuals)
pacf(arima.ts.2.1.2$residuals)
```

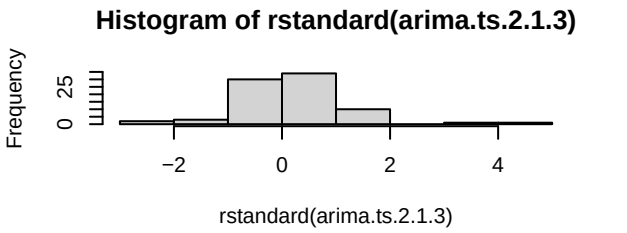


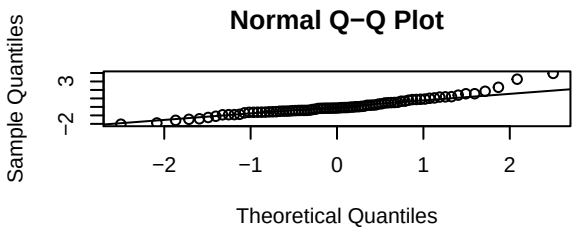
```
pacf(arima.ts.2.1.3$residuals)



# Histogram of residuals
hist(rstandard(arima.seasonal.ts.2.1.1))
hist(rstandard(arima.ts.2.1.2))
hist(rstandard(arima.ts.2.1.3))

# qqnorm plots of residuals
qqnorm(rstandard(arima.seasonal.ts.2.1.1))
qqline(rstandard(arima.seasonal.ts.2.1.1))
```



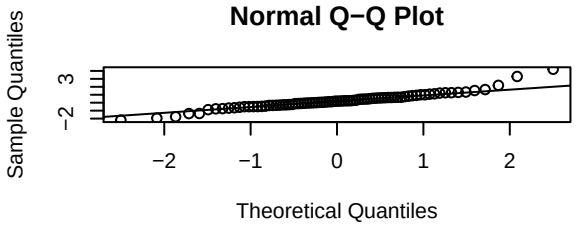


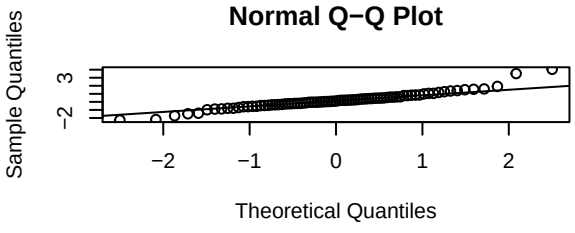







```
qqnorm(rstandard(arima.ts.2.1.2))
qqline(rstandard(arima.ts.2.1.2))

qqnorm(rstandard(arima.ts.2.1.3))
qqline(rstandard(arima.ts.2.1.3))
```





```
fit.1 <- Arima(trans.ts.2.1, order = c(0,1,2))
fit.1.ts <- ts(fit.1$fitted, frequency = 7, start = c(1,5))
fit.2 <-Arima(trans.ts.2.1, order = c(1,0,1), method = "ML")

par(mfrow = c(2,1))
plot(trans.ts.2.1, ylab = "cases^0.3", main = "IMA(1,2) accuracy")
lines(fit.1$fitted, col = "lightblue4")
plot(trans.ts.2.1, ylab = "cases^0.3", main = "ARMA(1,1) accuracy")
lines(fit.2$fitted, col = "darkorange")
```