Question 5

Read in the data and filter for only Californians and use `people_fully_vaccinated` to forecast.

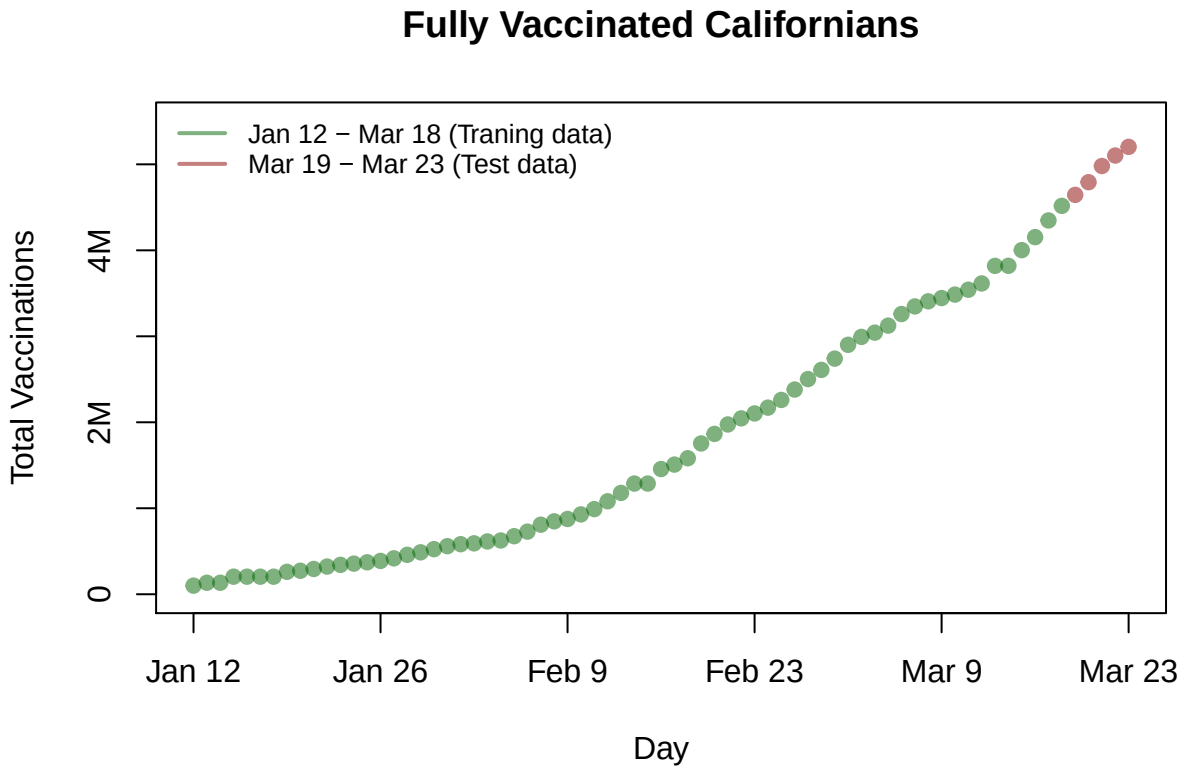
```
covid19 <- read.csv(file = "us_state_vaccinations.csv", header = TRUE)

covid19 <- covid19[covid19['location'] == 'California', 'people_fully_vaccinated']
```

Fill NA's with latest non-NA value.

```
library("zoo")
covid19 <- na.locf(covid19)
```

Divide the data into a training set (first 66 days) and testing set (last 5 days). Provided is a time series plot of the data with training and test sets in different colours.

```
covid19 = as.numeric(unlist(covid19))
covid.training <- head(covid19, 66)
covid.test <- tail(covid19, 5)

plot(covid.training,
     main = "Fully Vaccinated Californians",
     xlab = "Day",
     ylab = "Total Vaccinations",
     xlim = c(1, 71),
     ylim = c(0, 5500000),
     type = "p",
     pch = 19,
     col = adjustcolor("darkgreen", 0.5),
     xaxt = "n",
     yaxt = "n")

lines(y = covid.test,
      x = 67:71,
      type ="p",
      pch = 19,
      col = adjustcolor("darkred", 0.5))

axis(side = 1, at = (14*0:5)+1, labels = c("Jan 12","Jan 26","Feb 9","Feb 23","Mar 9", "Mar 23"))

axis(side = 2, at = (1000000*0:5), labels = c("0","","2M"," ","4M", " "))

legend("topleft",
       lwd = 2,
       bty = "n",
       cex = 0.8,
       legend = c("Jan 12 - Mar 18 (Traning data)",
                  "Mar 19 - Mar 23 (Test data)"),
       col = c(adjustcolor("darkgreen", 0.5),
               adjustcolor("darkred", 0.5)))
```



The framework to predict when at least 50% of Californians will be fully vaccinated is as follows:

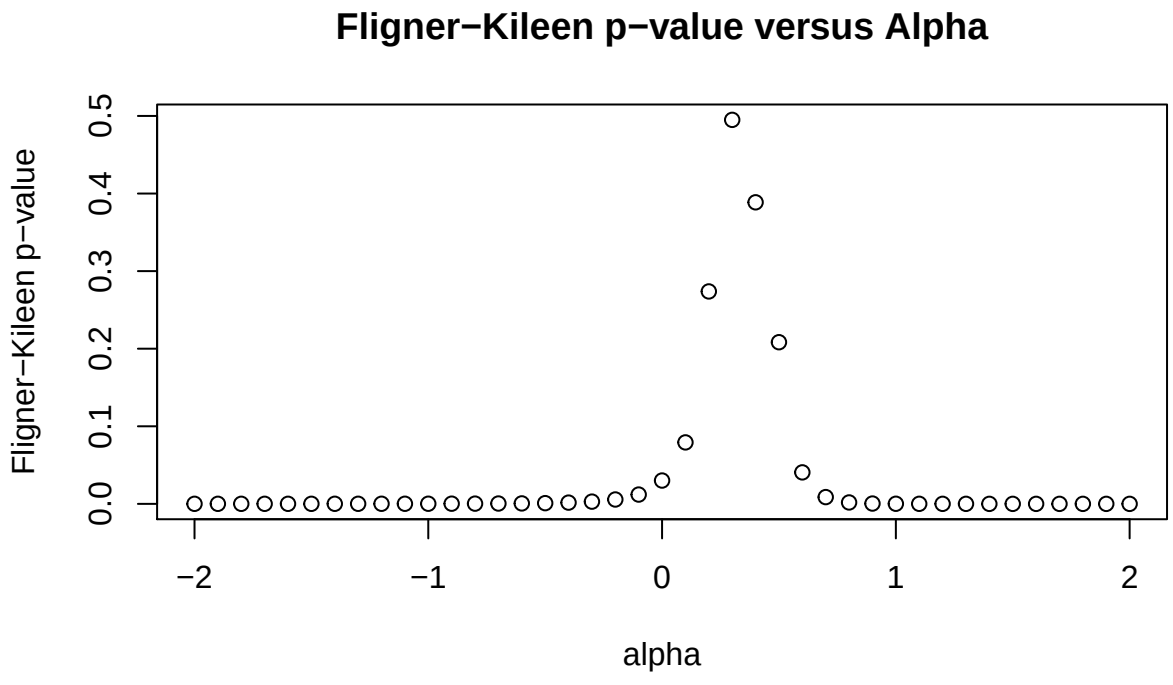
- transform non-stationary data to stationary data
- fit a stationary model
- forecast
- add non-stationarity back

Data with a trend, change points, heteroscedasticity, or seasonality indicate non-stationarity. From the time series plot we observe an increasing trend and heteroscedasticity.

It is important that the variance of the series is stabilized and its trend removed with differencing before we propose ARIMA models.

To stabilize the variance, let's search over the grid `alpha=seq(-2,2,by=0.1)` to find a value α such that $X = (covid19)^{\alpha}$ has a constant variance. Suppose for $\alpha = 0$, the transformation is $X = log(covid19)$. We bin the data into 6 segments, each containing 11 consecutive observations, and generate a plot where the x-axis shows the value of α and the y-axis shows the corresponding p-value of the Fligner-Kileen test of variance homogeneity for the transformed data X. To follow an objective method, among the satisfactory transformations, we choose the one with the largest p-value of Fligner's test.

```
# vector representing segment for the corresponding elements in transformed_covid19
seg <- c(rep(1:6, each=11))

# vector to store p_values from Fligner-Kileen tests
p_values <- c()

# vector containing values of alpha to search
alpha <- seq(-2,2,by=0.1)

for (i in alpha) {

  # apply transformation
  if (i == 0) transformed_covid = log(covid.training)
  else transformed_covid = covid.training^i

  # extract p-value from test
  p_value = fligner.test(transformed_covid, seg)$p.value

  # append p-value to vector
  p_values = c(p_values, p_value)

}

plot(x = alpha, y = p_values,
     ylab = "Fligner-Kileen p-value",
     main = "Fligner-Kileen p-value versus Alpha")
```



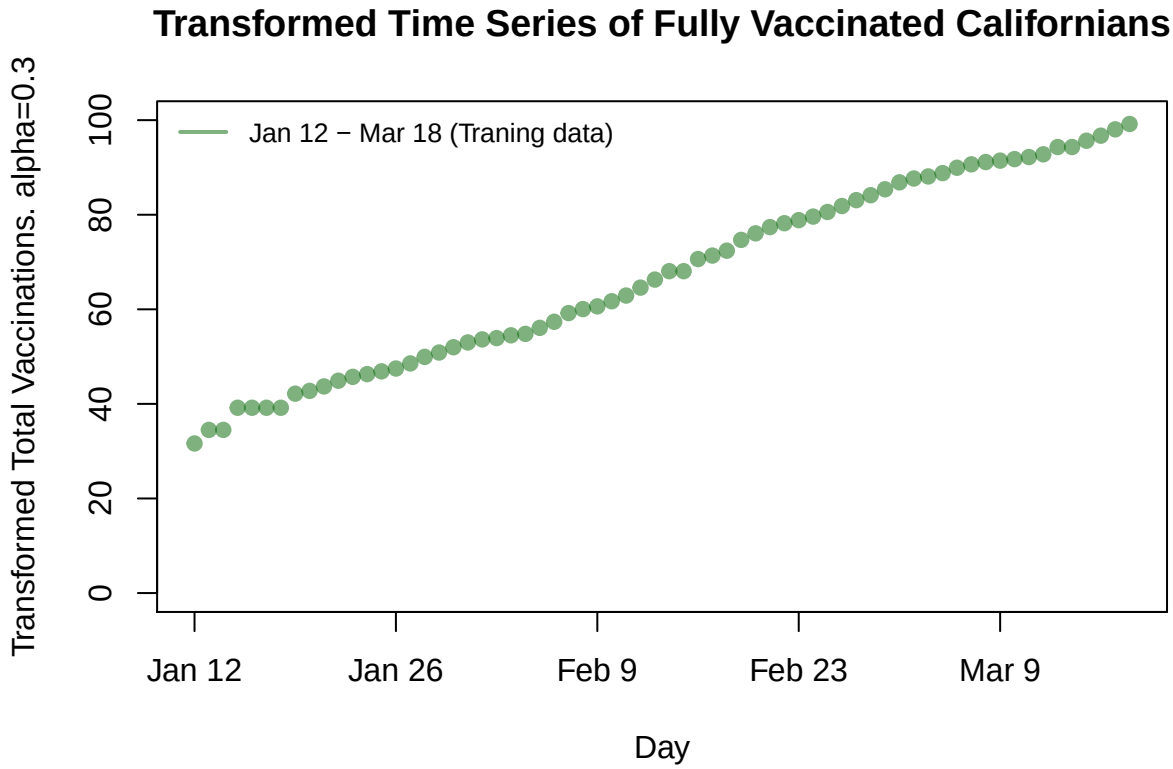
```r
# find alpha corresponding to largest p-value
alpha[which(p_values == max(p_values))]
```

```
## [1] 0.3
```

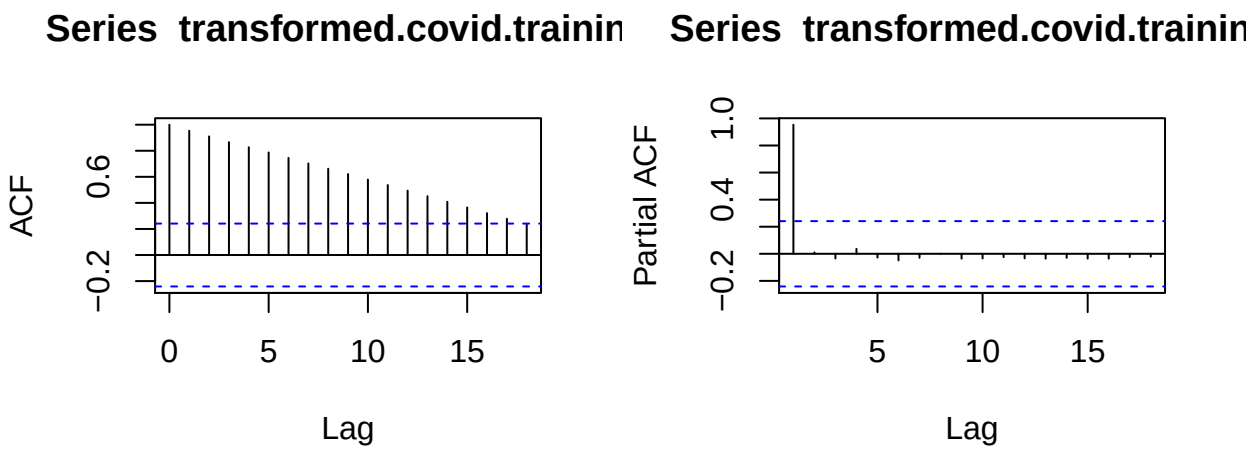
The desired transformation is $\alpha = 0.3$.

Provided is a time series plot of the transformed data along with the ACF and PACF plots.

```r
transformed.covid.training <- covid.training^alpha[which(p_values == max(p_values))]

plot(transformed.covid.training,
     main = "Transformed Time Series of Fully Vaccinated Californians",
     xlab = "Day",
     ylab = "Transformed Total Vaccinations. alpha=0.3",
     xlim = c(1, 66),
     ylim = c(0, 100),
     type = "p",
     pch = 19,
     col = adjustcolor("darkgreen", 0.5),
     xaxt = "n")

axis(side = 1, at = (14*0:5)+1, labels = c("Jan 12","Jan 26","Feb 9","Feb 23","Mar 9", "Mar 23"))

legend("topleft",
       lwd = 2,
       bty = "n",
       cex = 0.8,
       legend = c("Jan 12 - Mar 18 (Traning data)"),
       col = c(adjustcolor("darkgreen", 0.5)))
```

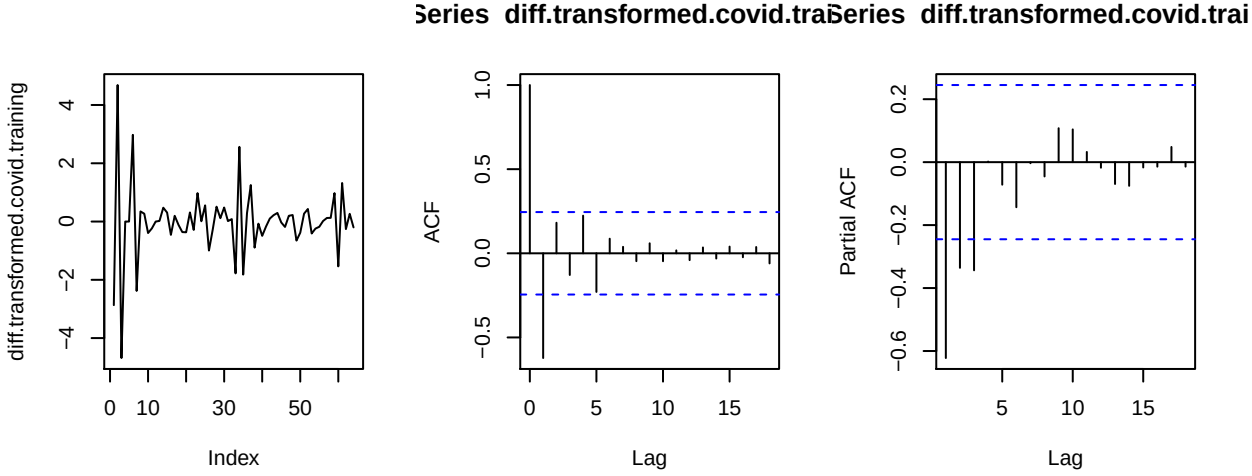


```
par(mfrow=c(1,2))
acf(transformed.covid.training)
pacf(transformed.covid.training)
```



There is a clear increasing trend in the data and the ACF has a linear decay that implies differencing is required to remove the trend.

```
diff.transformed.covid.training <- diff(transformed.covid.training, differences=2)

par(mfrow=c(1,3))
plot(diff.transformed.covid.training, type='l')
acf(diff.transformed.covid.training)
pacf(diff.transformed.covid.training)
```



Up until now, we have applied a power transformation to remove the heteroscedasticity and twice differencing to remove the trend.

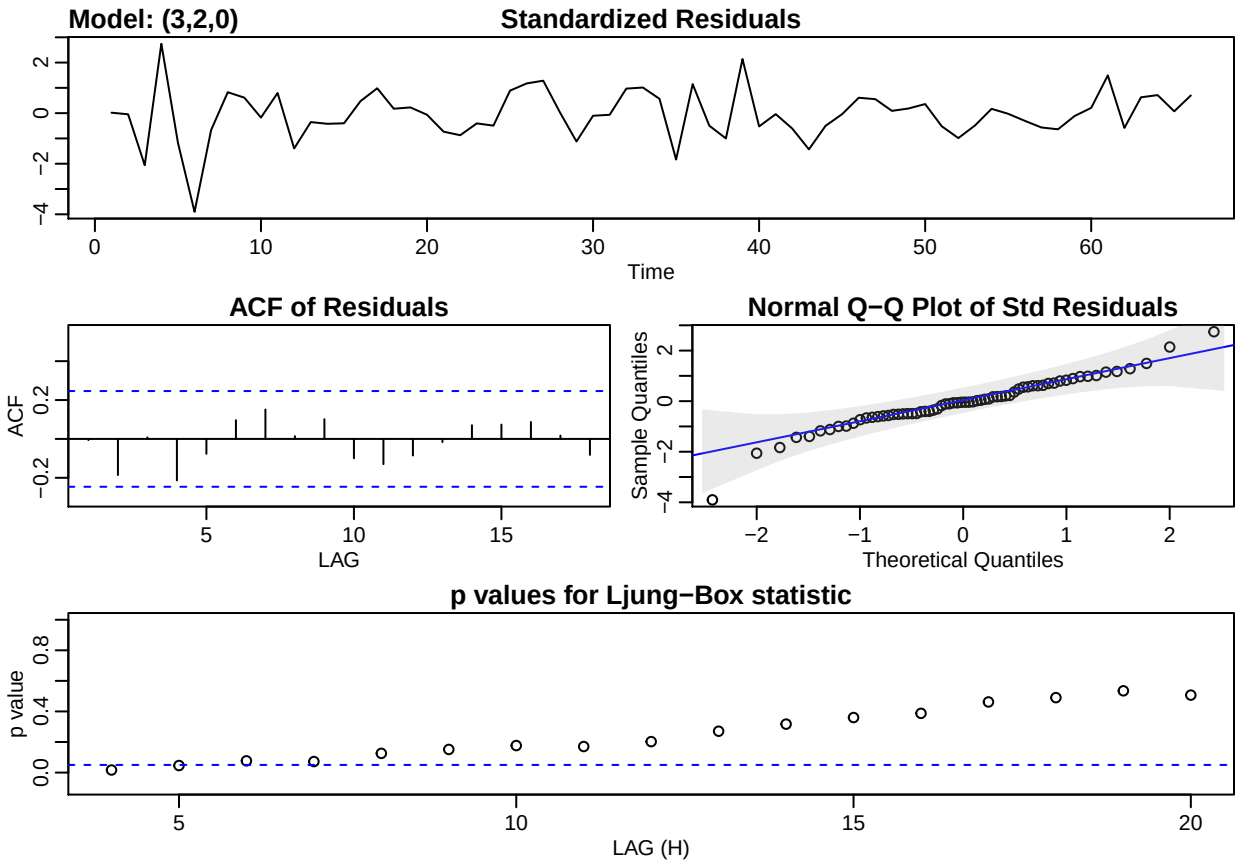
Next, we propose models for the stationary data and perform full residuals diagnostics for the proposed ARIMA models.

- Model 1: the ACF has an exponential decay and the PACF cuts off after lag 3, so we consider ARIMA(3,2,0).
- Model 2: the PACF has an exponential decay and the ACF cuts off after lag 1, so we consider ARIMA(0,2,1).
- Model 3: since in models 1 and 2 we justified exponential decay on the ACF and PACF, it could be exponential decay on both, so we consider ARIMA(1,2,1).

Model 1: ARIMA(3, 2, 0)

```
library(astsa)

model1 <- sarima(transformed.covid.training, p=3, d=2, q=0, details = TRUE)
```

```
## initial  value -0.195944
## iter   2 value -0.259502
## iter   3 value -0.355965
## iter   4 value -0.450607
## iter   5 value -0.452855
## iter   6 value -0.455779
## iter   7 value -0.455836
## iter   8 value -0.455842
## iter   8 value -0.455842
## final  value -0.455842
## converged
## initial  value -0.125312
## iter   2 value -0.182048
## iter   3 value -0.218676
## iter   4 value -0.220231
## iter   5 value -0.220533
## iter   6 value -0.220782
## iter   7 value -0.220853
## iter   8 value -0.220890
## iter   9 value -0.220892
## iter  10 value -0.220892
## iter  11 value -0.220892
## iter  12 value -0.220892
## iter  13 value -0.220892
## iter  14 value -0.220892
## iter  14 value -0.220892
## iter  14 value -0.220892
## final  value -0.220892
## converged
```




```
seg <- c(rep(1:5, each=11), rep(6,11))
fligner.test(model1$fit$residuals, seg)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  model1$fit$residuals and seg
## Fligner-Killeen:med chi-squared = 7.3465, df = 5, p-value = 0.1961
```

```
shapiro.test(model1$fit$residuals)
```

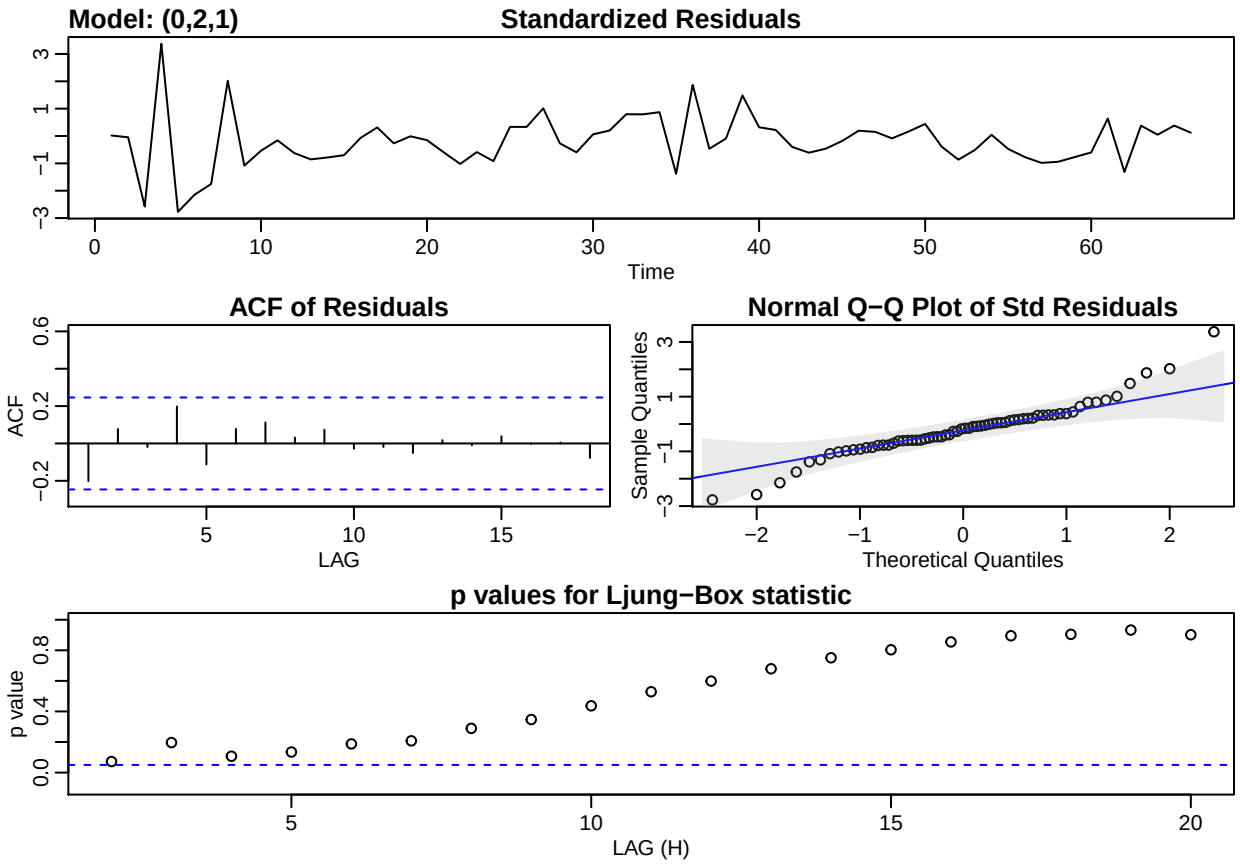
```
##
##  Shapiro-Wilk normality test
##
## data:  model1$fit$residuals
## W = 0.95251, p-value = 0.01316
```

- ARIMA(3, 2, 0)
  - Homoscedasticity: the time series of the residuals has no trend and the p-value of Fligner's test (0.1961) confirms the validity of homoscedasticity.
  - Normality: a few points in the QQplot deviate from the straight line resulting in the small p-value of Shapiro-Wilk's test (0.01316). While the normality assumption is not quite valid, it not heavily violated either.
  - White Noise: a few p-values of Ljung-Box statistic are below the threshold which raises concerns about the independence of the residuals.

Model 2: ARIMA(0, 2, 1)

```
library(astsa)

model2 <- sarima(transformed.covid.training, p=0, d=2, q=1, details = TRUE)
```

```
## initial  value 0.188759
## iter   2 value -0.105041
## iter   3 value -0.128231
## iter   4 value -0.128461
## iter   5 value -0.130250
## iter   6 value -0.130278
## iter   7 value -0.130279
## iter   7 value -0.130279
## final  value -0.130279
## converged
## initial  value -0.165956
## iter   2 value -0.203913
## iter   3 value -0.206846
## iter   4 value -0.207428
## iter   5 value -0.207431
## iter   6 value -0.207433
## iter   6 value -0.207433
## iter   6 value -0.207433
## final  value -0.207433
## converged
```




```
seg <- c(rep(1:5, each=11), rep(6,11))
fligner.test(model2$fit$residuals, seg)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  model2$fit$residuals and seg
## Fligner-Killeen:med chi-squared = 16.336, df = 5, p-value =
## 0.005947
```

```
shapiro.test(model2$fit$residuals)
```

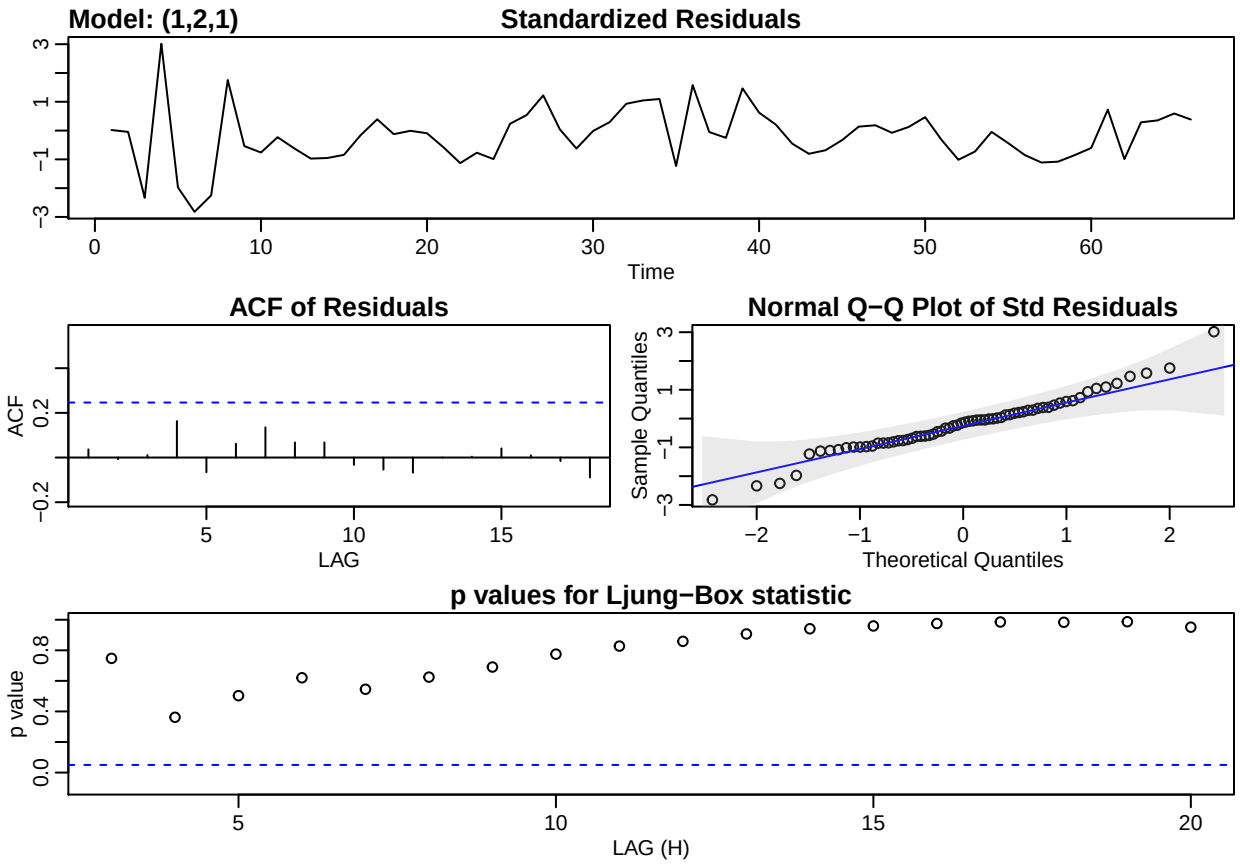
```
##
##  Shapiro-Wilk normality test
##
## data:  model2$fit$residuals
## W = 0.94062, p-value = 0.003409
```

- ARIMA(0, 2, 1)
  - Homoscedasticity: the time series of the residuals has no trend but the p-value of Fligner's test (0.005947) raises concern. The constant variance assumption of the residuals is not valid.
  - Normality: the points in the QQplot deviate from the straight line resulting in the small p-value of Shapiro-Wilk's test (0.003409). The normality assumption of the residuals is not valid.
  - White Noise: the p-values of the Ljung-Box statistic are above the threshold and there is no concern in the ACF plot. We can safely conclude the residuals are realizations of white noise.

Model 3: ARIMA(1, 2, 1)

```
library(astsa)

model3 <- sarima(transformed.covid.training, p=1, d=2, q=1, details = TRUE)
```

```
## initial  value 0.150463
## iter   2 value -0.163179
## iter   3 value -0.218471
## iter   4 value -0.246419
## iter   5 value -0.257607
## iter   6 value -0.258271
## iter   7 value -0.258533
## iter   8 value -0.258693
## iter   9 value -0.258695
## iter   9 value -0.258695
## iter   9 value -0.258695
## final  value -0.258695
## converged
## initial  value -0.226257
## iter   2 value -0.231111
## iter   3 value -0.231693
## iter   4 value -0.231952
## iter   5 value -0.232105
## iter   6 value -0.232113
## iter   7 value -0.232115
## iter   8 value -0.232116
## iter   9 value -0.232116
## iter   9 value -0.232116
## iter   9 value -0.232116
## final  value -0.232116
## converged
```




```
seg <- c(rep(1:5, each=11), rep(6,11))
fligner.test(model3$fit$residuals, seg)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  model3$fit$residuals and seg
## Fligner-Killeen:med chi-squared = 14.177, df = 5, p-value =
## 0.01452
```

```
shapiro.test(model3$fit$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model3$fit$residuals
## W = 0.96917, p-value = 0.09931
```

- ARIMA(1, 2, 1)
  - Homoscedasticity: the time series of the residuals has no trend and the p-value of Fligner's test (0.01452) does not raise much concern.
  - Normality: a few points in the QQplot slightly deviate from the straight line resulting in the p-value of Shapiro-Wilk's test (0.09931). The normality assumption is valid.
  - White Noise: the p-values of the Ljung-Box statistic are above the threshold and there is no concern in the ACF plot. We can safely conclude the residuals are realizations of white noise.

Let's compare the three proposed ARIMA models in terms of their prediction power using PRESS and MSE.

```
par(mfrow=c(3,1))
pred.model1 <- sarima.for(transformed.covid.training, n.ahead=5, p=3, d=2, q=0)
pred.model2 <- sarima.for(transformed.covid.training, n.ahead=5, p=0, d=2, q=1)
pred.model3 <- sarima.for(transformed.covid.training, n.ahead=5, p=1, d=2, q=1)
```

```
transformed.covid.test <- covid.test^alpha[which(p_values == max(p_values))]

PRESS.1 <- sum((pred.model1$pred-transformed.covid.test)^2)
PRESS.2 <- sum((pred.model2$pred-transformed.covid.test)^2)
PRESS.3 <- sum((pred.model3$pred-transformed.covid.test)^2)

tab = rbind(1:3, c(PRESS.1, PRESS.2, PRESS.3))
row.names(tab) = c("ARIMA Model", "PRESS")
dimnames(tab)[[2]] = rep("", 3)
tab
```

```
##
## ARIMA Model 1.000000 2.000000 3.0000000
## PRESS       6.009946 1.187855 0.1984051
```

Based on the PRESS statistic, **ARIMA(1, 2, 1)** predicts the best.

```
MSE.1 <- sum((pred.model1$pred-transformed.covid.test)^2)/3
MSE.2 <- sum((pred.model1$pred-transformed.covid.test)^2)/1
MSE.3 <- sum((pred.model1$pred-transformed.covid.test)^2)/2

tab = rbind(1:3, c(MSE.1, MSE.2, MSE.3))
row.names(tab) = c("ARIMA Model", "MSE")
dimnames(tab)[[2]] = rep("", 3)
tab
```

```
##
## ARIMA Model 1.000000 2.000000 3.000000
## MSE         2.003315 6.009946 3.004973
```

Based on the MSE statistic which accounts for the number of parameters, **ARIMA(3, 2, 0)** predicts the best.

We choose the superior model to be **ARIMA(1, 2, 1)** considering its complexity, PRESS statistic and residuals analysis.

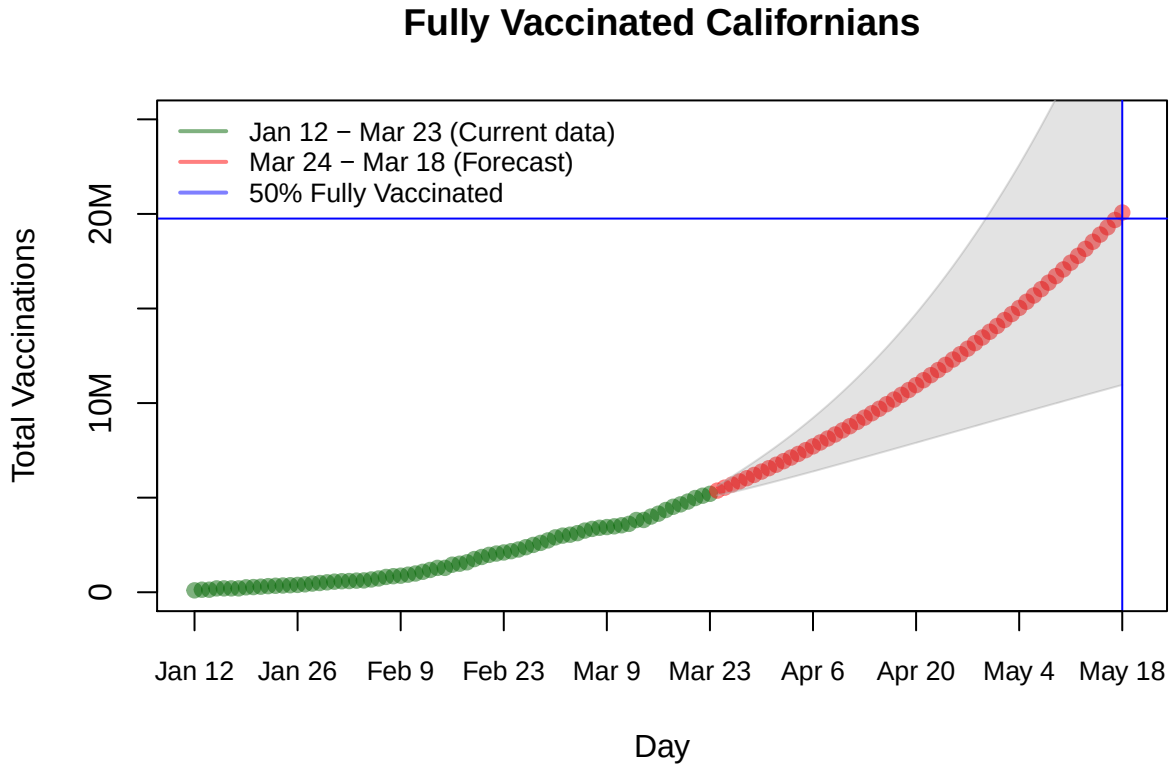
Now, we use the superior model to predict (along with a 95% prediction interval) when at least 50% of the Californians will be fully vaccinated.

```
transformed.covid19 <- covid19^alpha[which(p_values == max(p_values))]

final.model <- sarima.for(transformed.covid19, n.ahead = 56, p=1, d=2, q=1)
```

```
fit <- (final.model$pred)^(1/0.3)
lwr <- (final.model$pred-1.96*final.model$se)^(1/0.3)
upr <- (final.model$pred+1.96*final.model$se)^(1/0.3)
```

```r
# data.frame(fit, lwr, upr)
```

```r
plot(covid19,
     main = "Fully Vaccinated Californians",
     xlab = "Day",
     ylab = "Total Vaccinations",
     xlim = c(1, 128),
     ylim = c(0, 25000000),
     type = "p",
     pch = 19,
     col = adjustcolor("darkgreen", 0.5),
     xaxt = "n",
     yaxt = "n")

lines(y = fit,
      x = 72:127,
      type ="p",
      pch = 19,
      col = adjustcolor("red", 0.5))

lines(y = lwr,
      x = 72:127,
      type ="l",
      pch = 19,
      col = adjustcolor("grey", 0.5))

lines(y = upr,
      x = 72:127,
      type ="l",
      pch = 19,
      col = adjustcolor("grey", 0.5))

polygon(c(72:127, rev(72:127)), c(lwr, rev(upr)), col = "#69696930", border = NA)
abline(h=19755000, v=127, col="blue")

axis(side = 1,
     at = (14*0:9)+1,
     cex.axis = 0.85,
     labels = c("Jan 12", "Jan 26", "Feb 9", "Feb 23", "Mar 9", "Mar 23",
                "Apr 6", "Apr 20", "May 4", "May 18"))

axis(side = 2, at = (5000000*0:5), labels = c("0","","10M","","20M", ""))

legend("topleft",
       lwd = 2,
       bty = "n",
       cex = 0.8,
       legend = c("Jan 12 - Mar 23 (Current data)",
                  "Mar 24 - Mar 18 (Forecast)",
                  "50% Fully Vaccinated"),
       col = c(adjustcolor("darkgreen", 0.5),
               adjustcolor("red", 0.5),
               adjustcolor("blue", 0.5)))
```



According to our model, at least 50% of Californians (~20M, source: Google) will be fully vaccinated on **May 18, 2021**. However, it could be as early as May or as late as the end of the year (not plotted).

For reference, according to Our World in Data, only 40% of Californians (~15.5M) have been fully vaccinated as of May 18, 2021. The most likely reason behind the overestimate is the drop of Californians that received at least one dose of a vaccine between April and May (data not included in our dataset). Fewer Californians that receive at least one dose of a vaccine impact the number of Californians that become fully vaccinated.