Forecasting The Houston Rockets Wins

Introduction

The Houston Rockets are one of the premier NBA franchises. Last season, they led the league in number of wins with 65. They were also one of the top playoff performers as well, finishing in the top four teams. They seem to have a bright future, and should be considered a top threat to win the NBA championship. One of the ways we can confirm this is to forecast the number of wins over the next few seasons. If the number of wins remain high, then we know that the team is still a top team. Using data from BasketballReference.com, we will forecast the number of wins and win percentage for the Rockets over the next 3 seasons.

Results

First we will set our working directory.

```
setwd("~/Desktop/Summer Project #1")
```

Next we will load our libraries.

```
library(lattice)
library(foreign)
library(MASS)
library(car)
```

```
## Loading required package: carData
```

```
require(stats)
require(stats4)
```

```
## Loading required package: stats4
```

```r
library(KernSmooth)
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```r
library(fastICA)
library(cluster)
library(leaps)
library(mgcv)
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.8-23. For overview type 'help("mgcv-package")'.
```

```r
library(rpart)
library(pan)
library(mgcv)
library(DAAG)
```

```
##
## Attaching package: 'DAAG'
```

```
## The following object is masked from 'package:car':
##
##     vif
```

```
## The following object is masked from 'package:MASS':
##
##     hills
```

```r
library("TTR")
library(tis)
```

```
##
## Attaching package: 'tis'
```

```
## The following object is masked from 'package:TTR':
##
##     lags
```

```
## The following object is masked from 'package:mgcv':
##
##     ti
```

```r
require("datasets")
require(graphics)
library("forecast")
```

```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:tis':
##
##     easter
```

```
## The following object is masked from 'package:nlme':
##
##     getResponse
```

```r
#install.packages("astsa")
#require(astsa)
```

```r
library(nlstools)
```

```
##
## 'nlstools' has been loaded.
```

```
## IMPORTANT NOTICE: Most nonlinear regression models and data set examples
```

```
## related to predictive microbiolgy have been moved to the package 'nlsMicrobio'
```

```r
library(fpp)
```

```
## Loading required package: fma
```

```
##
## Attaching package: 'fma'
```

```
## The following objects are masked from 'package:DAAG':
##
##     milk, ozone
```

```
## The following objects are masked from 'package:MASS':
##
##     cement, housing, petrol
```

```
## Loading required package: expsmooth
```

```
## Loading required package: lmtest
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: tseries
```

```r
library(strucchange)
```

```
## Loading required package: sandwich
```

```r
library(Quandl)
```

```
## Loading required package: xts
```

```r
library(zoo)
library(PerformanceAnalytics)
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```r
library(quantmod)
```

```
## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
##
## Attaching package: 'quantmod'
```

```
## The following object is masked from 'package:tis':
##
##     Lag
```

```r
#library(stockPortfolio)
library(vars)
```

```
## Loading required package: urca
```

```r
library(lmtest)
library(dlnm)
```

```
## This is dlnm 2.3.4. For details: help(dlnm) and vignette('dlnmOverview').
```

```r
library(hts)
library(tseries)
library(rugarch)
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'rugarch'
## The following object is masked from 'package:stats':
##
##     sigma
```

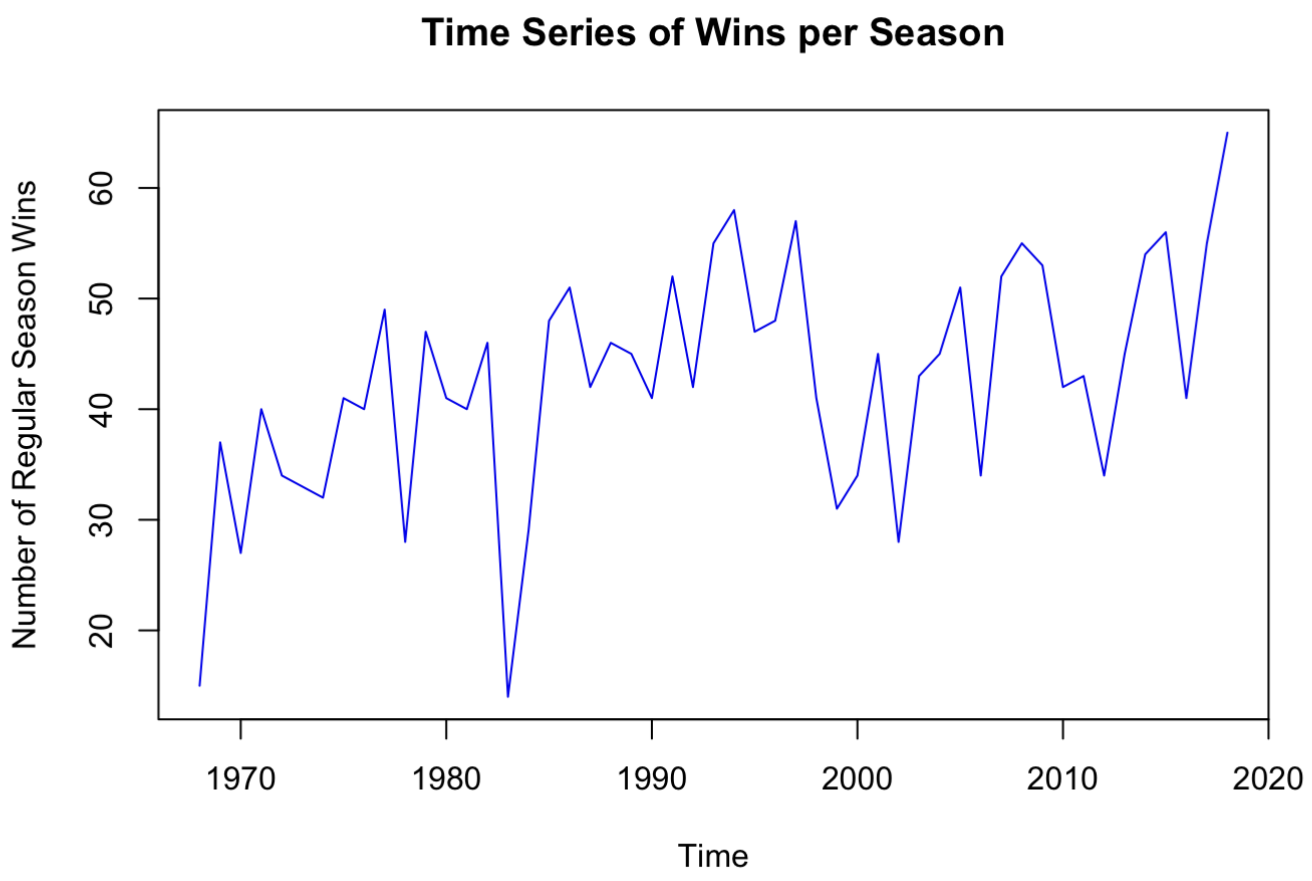
Now we will load in the data.

```
library(readxl)
houston_rockets_wins <- read_excel("Summer Project #1 Data.xlsx")
View(houston_rockets_wins)
```

```
## Warning in system2("/usr/bin/otool", c("-L", shQuote(DSO)), stdout = TRUE):
## running command ''/usr/bin/otool' -L '/Library/Frameworks/R.framework/
## Resources/modules/R_de.so'' had status 69
```

First we will generate a time series of the number of wins per season.

```
wins_ts <- ts(houston_rockets_wins$W, start = 1968, frequency = 1)
plot(wins_ts, xlab = "Time", ylab = "Number of Regular Season Wins", main = "Time Series of Wins per Season", col
 = "blue2")
```



We will also plot a time series of the first order difference. We do this because we want to work with covariance stationary data. Unfortunately the data above is not covarince stationary. We confirm this with the Augmented Dickey-Fuller Test. Looking at the time series further, we see that there seems to be an upward trend. Judging by the context of the data, there is probably no seasonality.

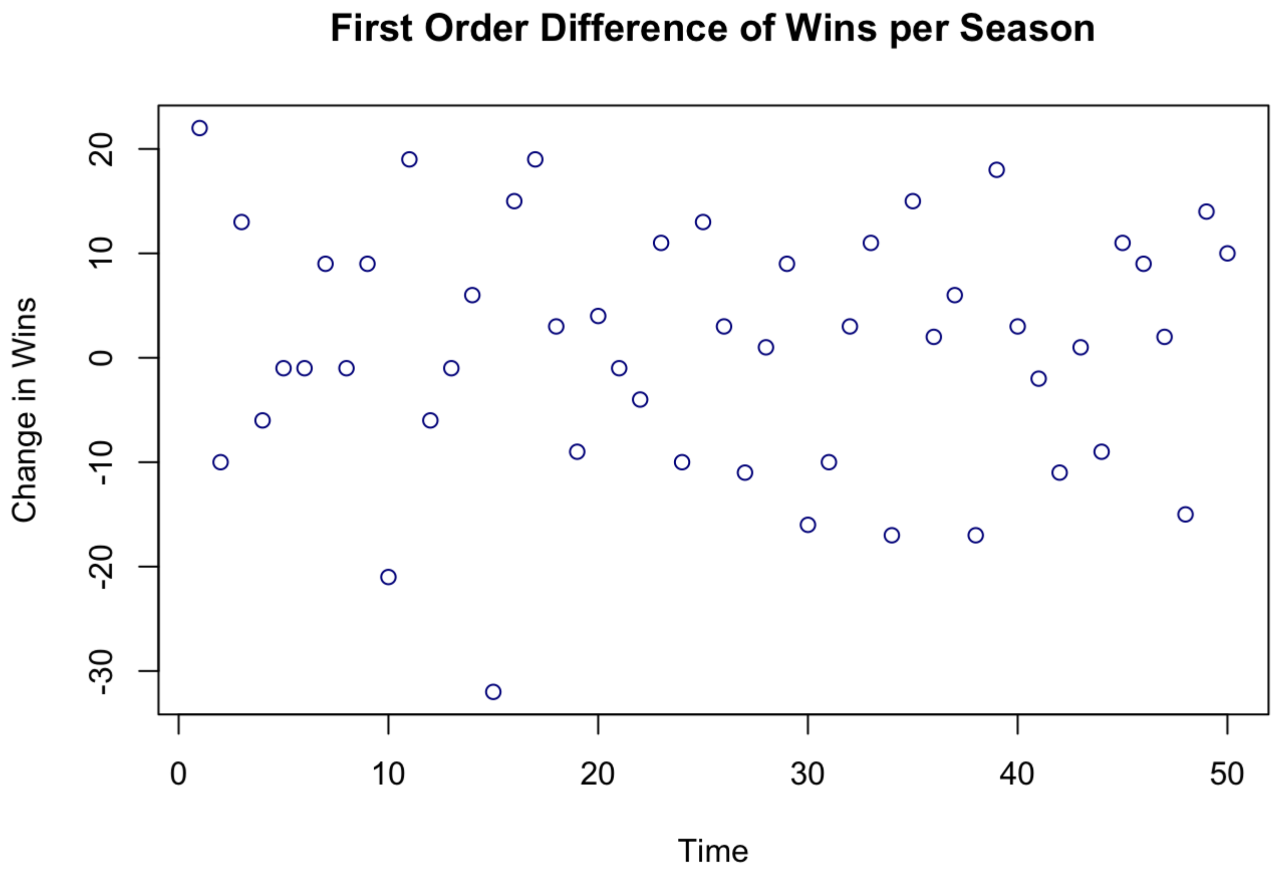
```
adf.test(wins_ts, alternative = "stationary", k = 20)
```

```
##
##  Augmented Dickey-Fuller Test
##
```

```
## data:  wins_ts
## Dickey-Fuller = -1.9834, Lag order = 20, p-value = 0.5811
## alternative hypothesis: stationary
```

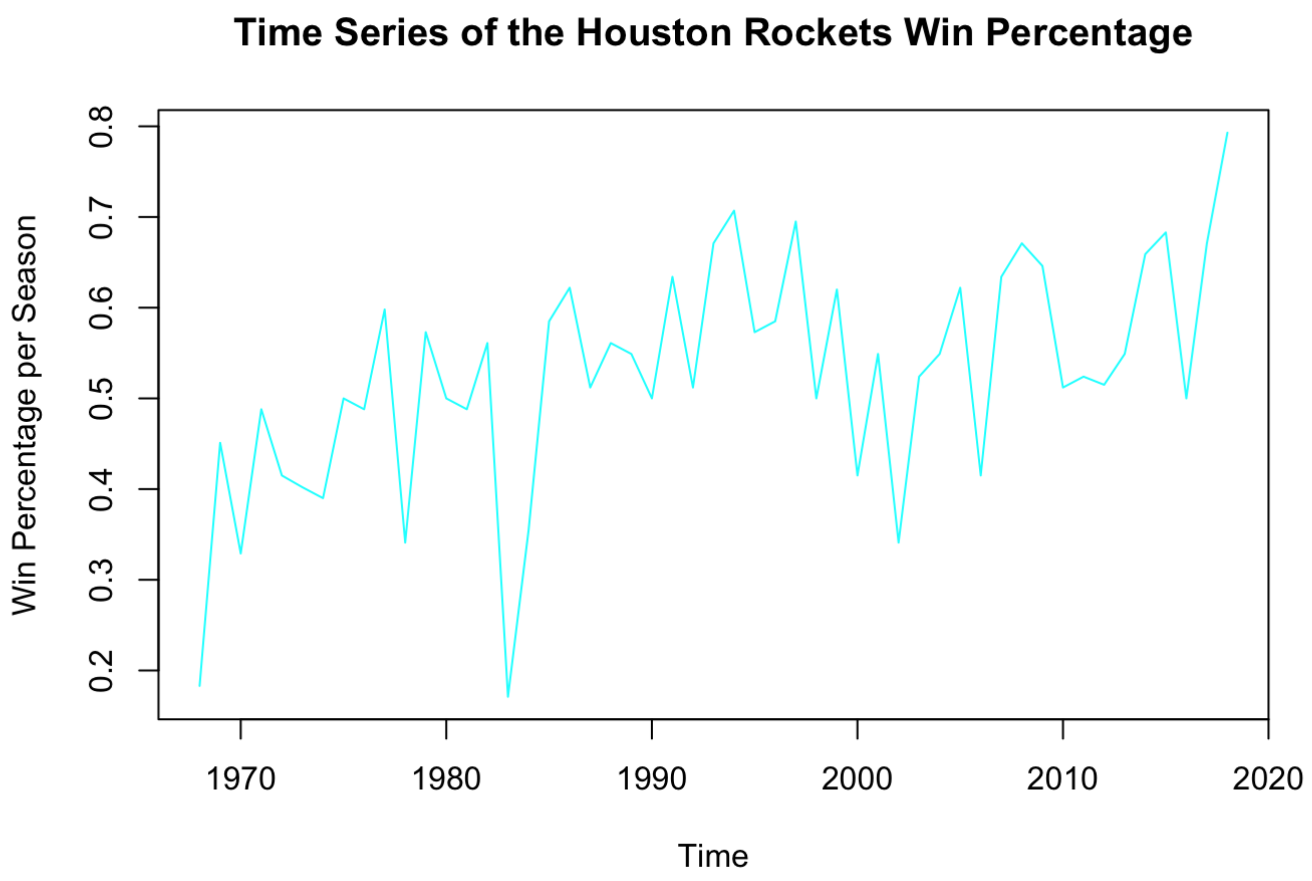
The high p-value of .5811 confirms that we the data is not stationary. We will now define the first order difference and generate a time series of first order difference wins.

```r
library(ggplot2)
first_order_wins <- diff(houston_rockets_wins$W)
first_order_ts <- ts(first_order_wins, start = 1968, frequency = 1)
plot(first_order_wins, xlab = "Time", ylab = "Change in Wins", main = "First Order Difference of Wins per Season"
, col = "blue4")
```



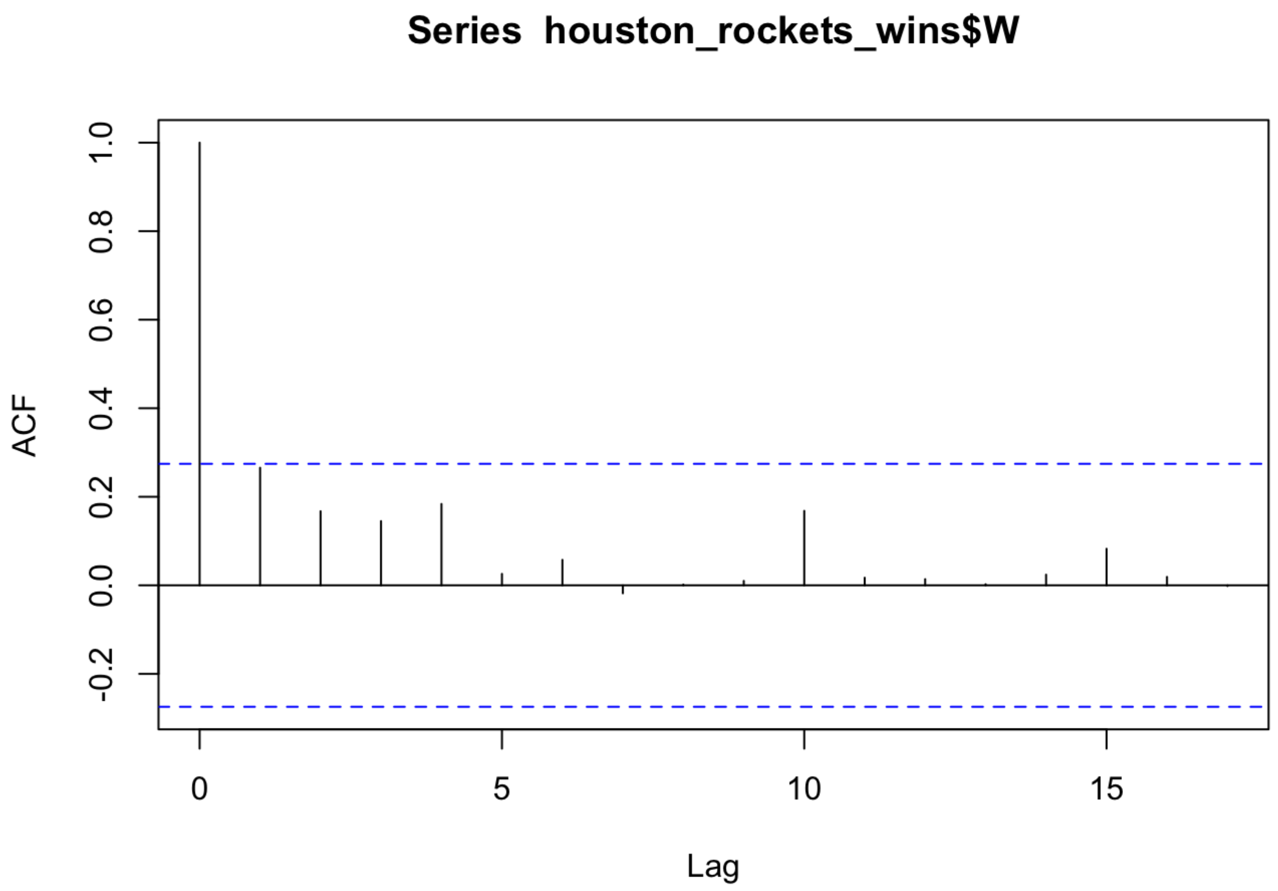
The results seem to be centered around zero and the data looks covariance stationary. For completeness, we will generate a time series of the win percentage.

```
win_percentage_ts <- ts(houston_rockets_wins$`W/L%`, start = 1968, frequency = 1)
plot(win_percentage_ts, ylab = "Win Percentage per Season", main = "Time Series of the Houston Rockets Win Percen
tage", col = "cyan")
```

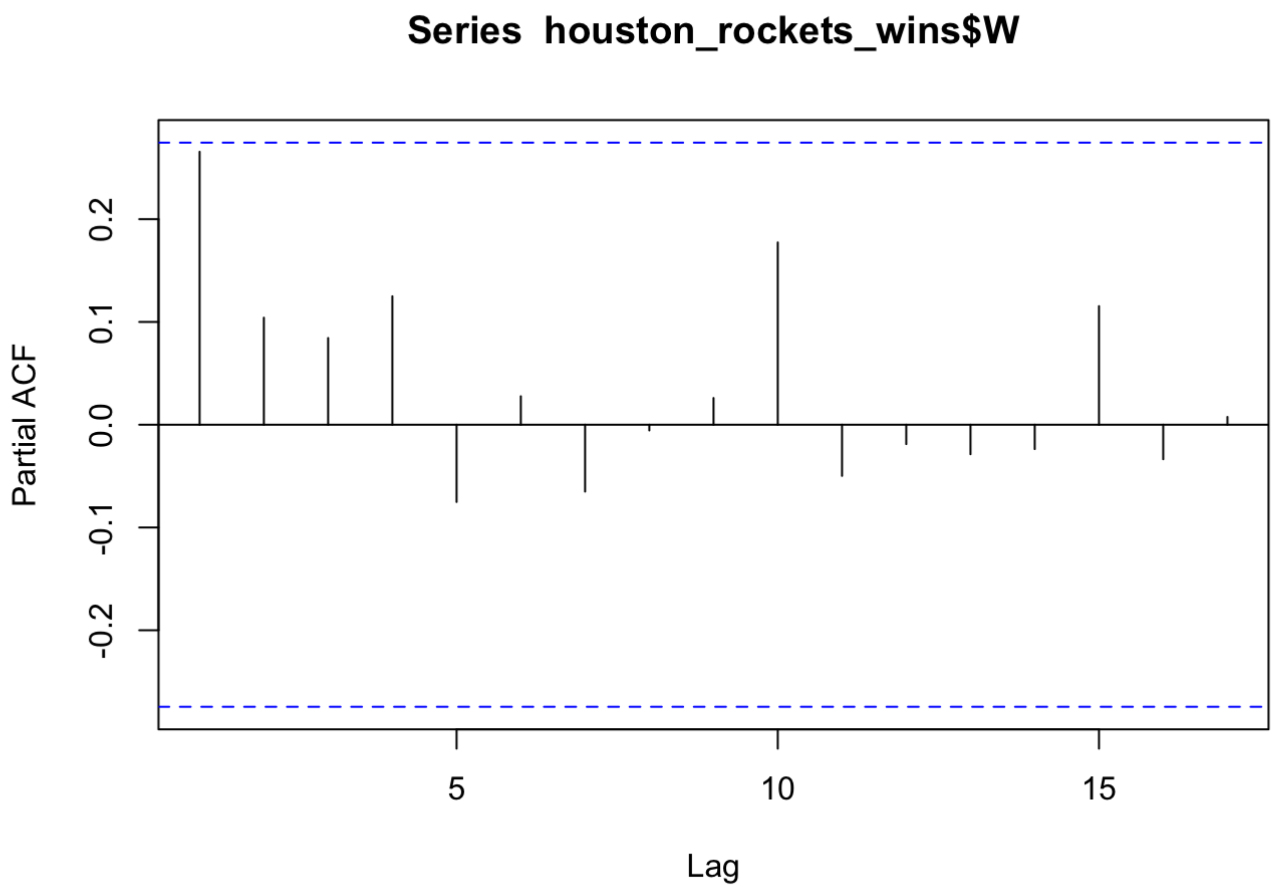


We will also generate the ACF and PACF of the Number of Wins Per Season. The ACF allows us to tell how much the number of wins in the current season depends on the number of wins in the previous seasons. This will allow us to examine any cyclical movements in the data as well. Intuitively, we expect that the ACF will decay to zero as the number of wins in the current season may depend on the number of wins in the previous season or perhaps the season before the previous season, but probably wont depend on the number of wins from 11 or 12 seasons ago. We will also generate the ACF and PACF of the first order difference as the first order difference data is stationary.

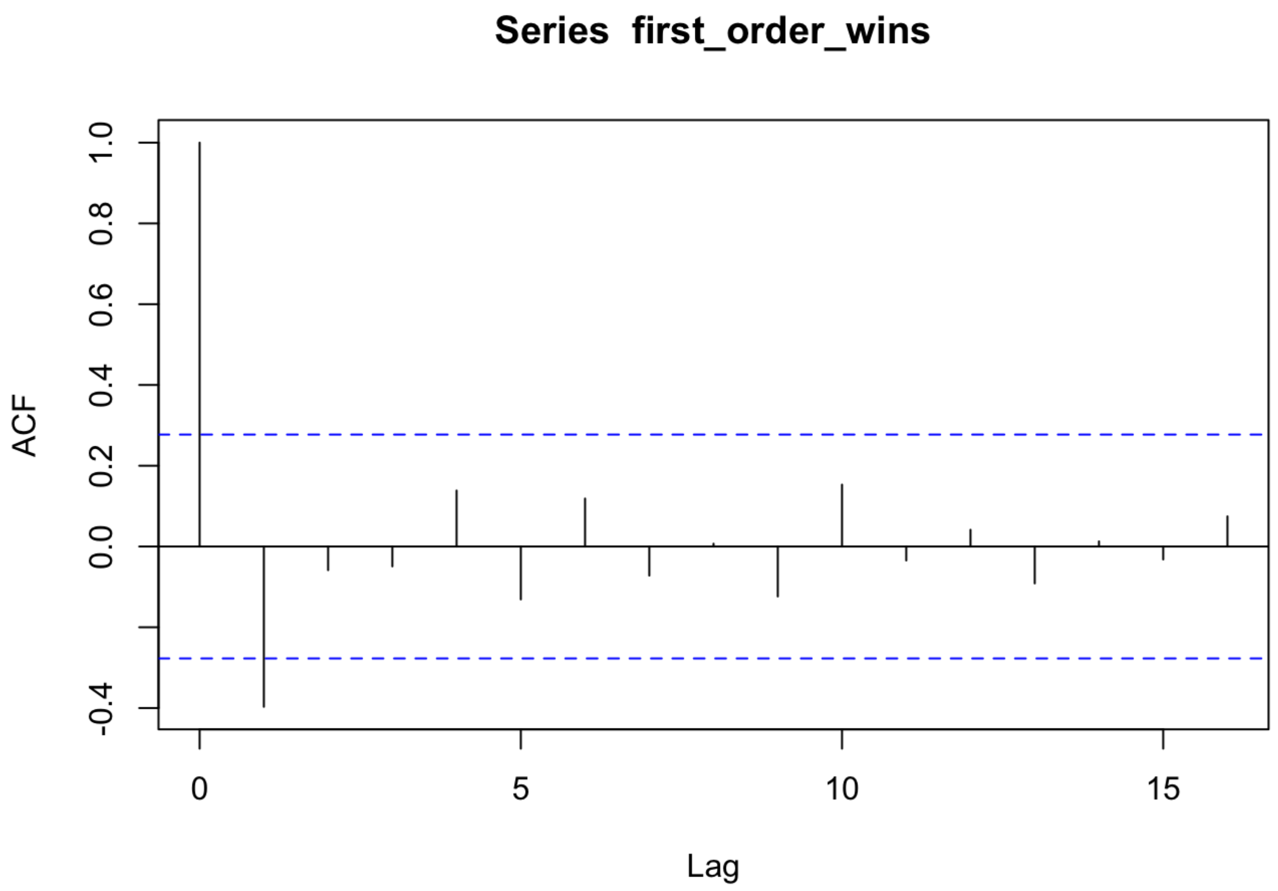
```
acf(houston_rockets_wins$W)
```



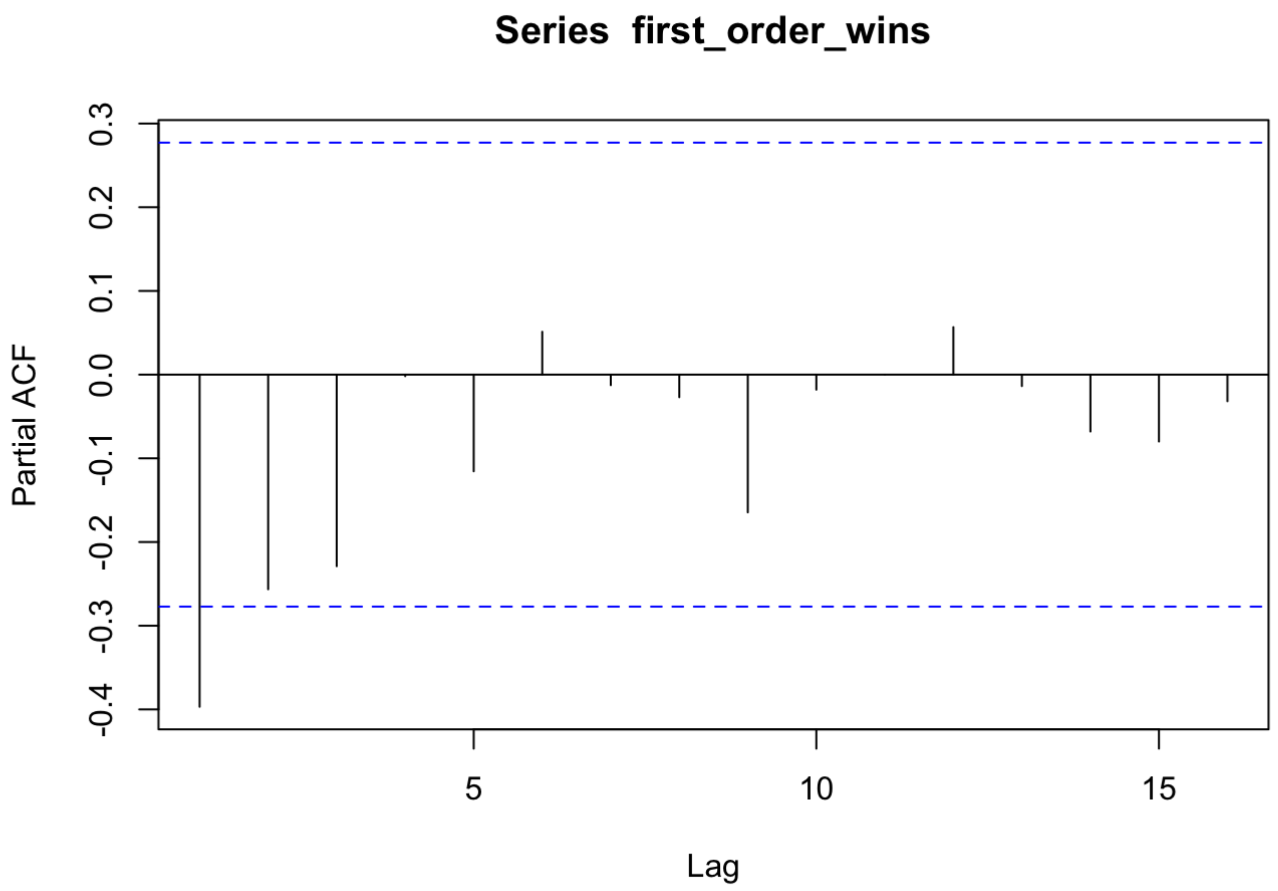
```
pacf(houston_rockets_wins$W)
```



```
acf(first_order_wins)
```



```
pacf(first_order_wins)
```



Looking at the ACF, we see that it does indeed decay to zero as we had expected. The PACF also seems to decay to zero but is significant for the first period. This gives us evidence that an autoregressive process of order 1, or ARIMA(1,1,0) may be the best model to forecast with. We will use ARIMA as opposed to AR since the first order difference is stationary, meaning that the data is the data is integrated of order 1, meaning that the first ofder difference is stationary. This means that the I in ARIMA is equal to 1 and not 0.

Looking at the ACF and PACF of the first order difference, we see similar results, except the fact that the PACF remains significant for 2 or perhaps 3 periods. This gives us evidence that an autoregressive process of order 2 or 3 may be the best model to forecast with. The way that we will choose the model among all the potential models is that we will select the model that has the smallest AIC value. We will also simulate an ARIMA(4) for completeness.

```r
arima(wins_ts, order = c(1,1,0))
```

```
##
## Call:
## arima(x = wins_ts, order = c(1, 1, 0))
##
## Coefficients:
##            ar1
##        -0.4188
## s.e.    0.1330
##
## sigma^2 estimated as 112.6:  log likelihood = -189.13,  aic = 382.27
```

```r
arima(wins_ts, order = c(2,1,0))
```

```
##
## Call:
## arima(x = wins_ts, order = c(2, 1, 0))
##
## Coefficients:
##            ar1      ar2
##        -0.5429  -0.2925
## s.e.    0.1418   0.1428
##
## sigma^2 estimated as 103.6:  log likelihood = -187.14,  aic = 380.27
```

```r
arima(wins_ts, order = c(3,1,0))
```

```
##
## Call:
## arima(x = wins_ts, order = c(3, 1, 0))
##
## Coefficients:
```

```
##          ar1      ar2      ar3
##      -0.6293  -0.4396  -0.2826
## s.e.   0.1434   0.1566   0.1448
##
## sigma^2 estimated as 95.83:  log likelihood = -185.32,  aic = 378.64
```

```
arima(wins_ts, order = c(4,1,0))
```

```
##
## Call:
## arima(x = wins_ts, order = c(4, 1, 0))
##
## Coefficients:
##          ar1      ar2      ar3      ar4
##      -0.6357  -0.4498  -0.2961  -0.0223
## s.e.   0.1496   0.1711   0.1709   0.1498
##
## sigma^2 estimated as 95.79:  log likelihood = -185.31,  aic = 380.62
```

Here we see that the ARIMA(3,1,0) model has the smallest AIC value. This makes sense since we saw that the PACF of the first order difference has 3 rather large spikes. We will forecast with this model. However before forecasting, we will run diagnostics on this model.

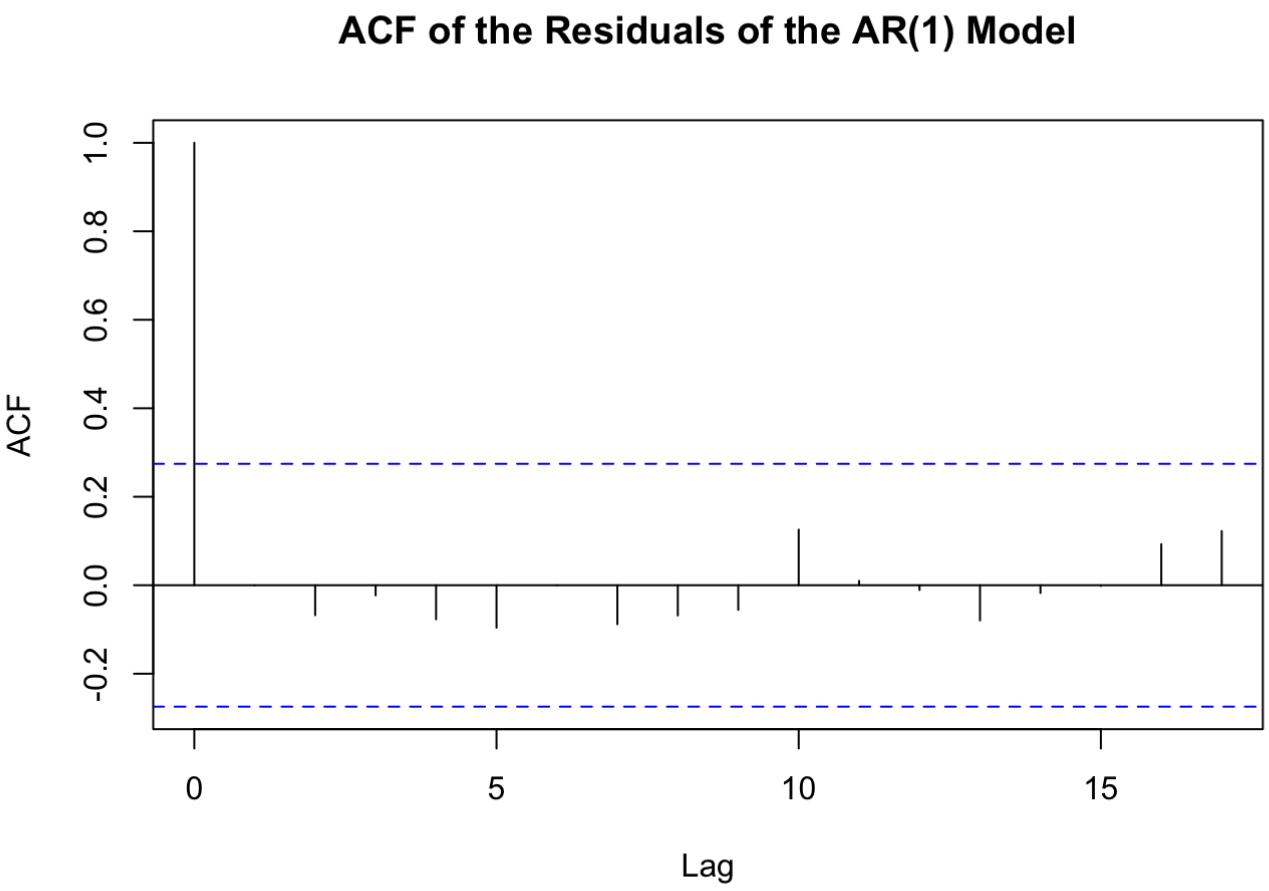
First we will plot the residuals vs. fitted values. We want this graph to be centered around zero as we want the average of the residuals to be zero.

```
ar3_model <- arima(wins_ts, order = c(3,1,0))
#plot(fitted.values(ar3_model), residuals(ar3_model), xlab = "Fitted Values", ylab = "Residuals", main = "Residuals vs. Fitted Values Graph", col = "darkslateblue")
```

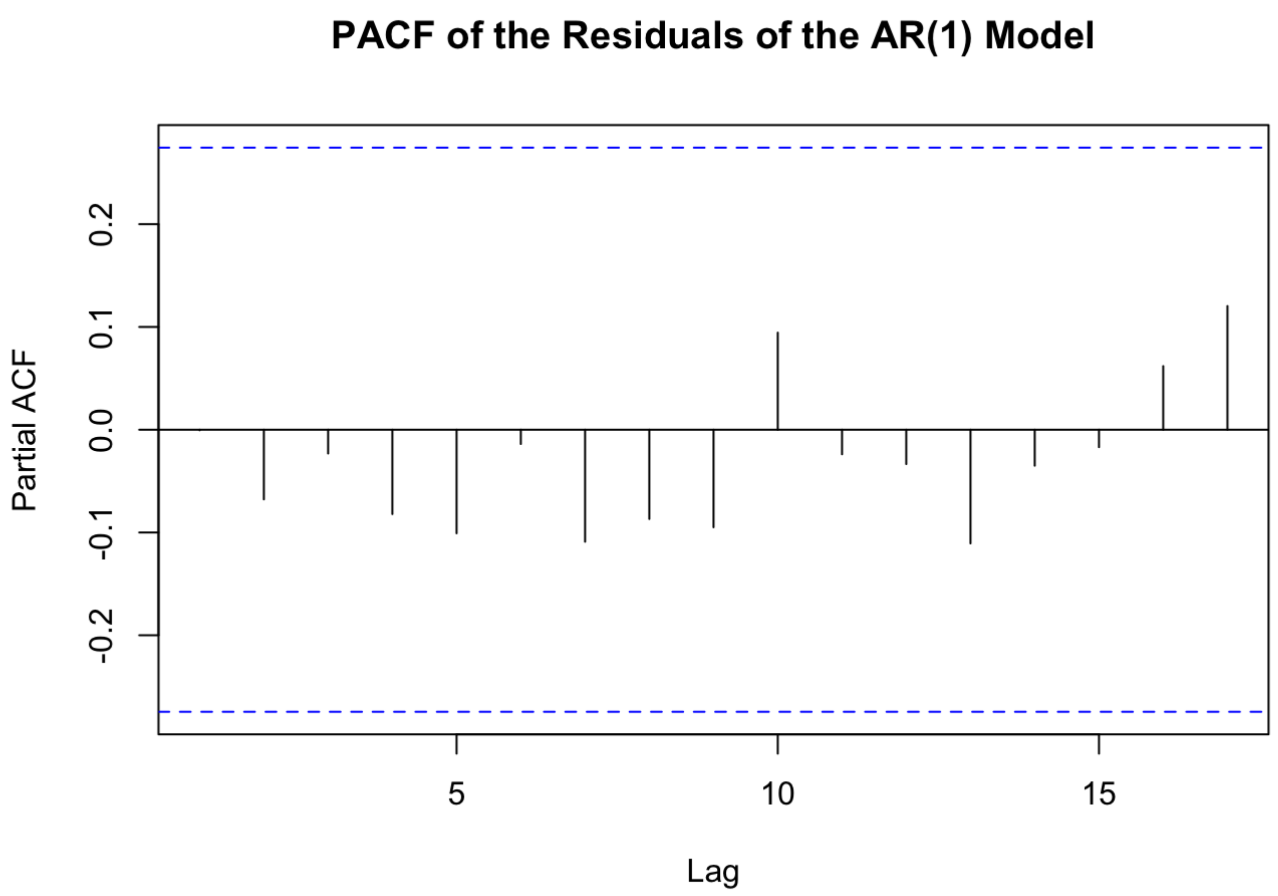
We see that the residuals vs fitted values for most years is centered at 0 on the y-axis. This is good because we want the residuals of the model to be centered at zero. This is a sign that the ARIMA(3,1,0) model captures the dynamics of the model.

Another way we can confirm that the model captures the dynamics of the model is to generate the ACF and PACF of the residuals. We will do that now.

```
acf(ar3_model$residuals, main = "ACF of the Residuals of the AR(1) Model")
```



```
pacf(ar3_model$residuals, main = "PACF of the Residuals of the AR(1) Model")
```



Looking at the ACF and PACF of the Residuals, we see that all spikes beside lag 0 are contained in the blue dashed lines, or Bartlett Bands. The reason that the lag 0 spike is outside the Bartlett Band is becuase the lag 0 spike on the ACF is always 1 regardless of the data. In this case, the number of wins in the current season has a total correlation with the number of wins in the current season. Beside that, all spikes on both graphs are contianed. This means that the ACF and PACF are not statistically significant and statistically equal to zero. This proves that the residuals are white noise. This means that they have a mean of zero and a constant variance. The residuals are essentially random, which is what we wanted. Our model captures all non random aspects of the data and all that is left is the random fluctuations.

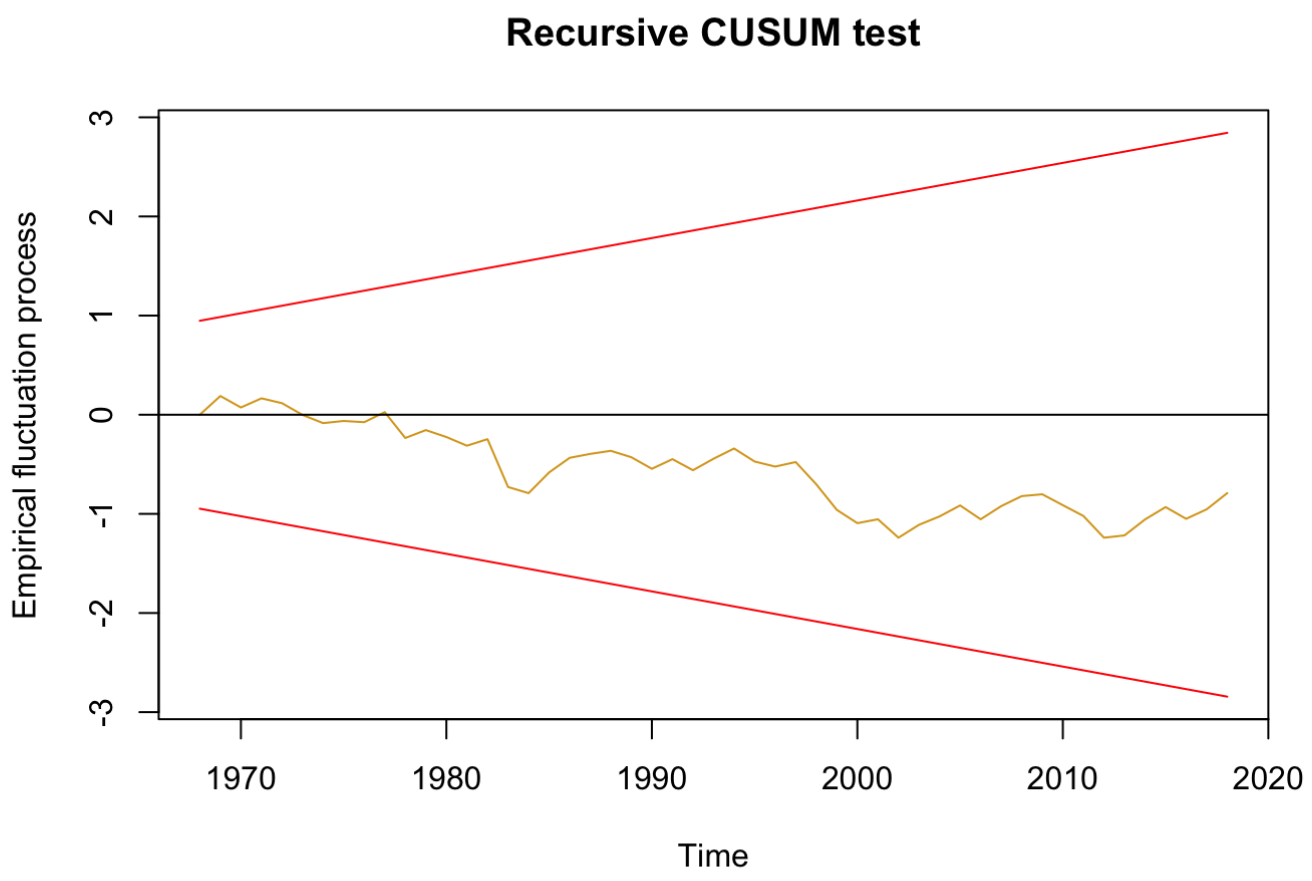
We need to see how sturdy our model is as well. Sometimes, the addition of a few data observations can cause structrual breaks in the model. This is something we want to avoid. We will generate the CUSUM, or cumulative sum model.

```
library(qcc)
```

```
## Package 'qcc' version 2.7
```

```
## Type 'citation("qcc")' for citing this R package in publications.
```

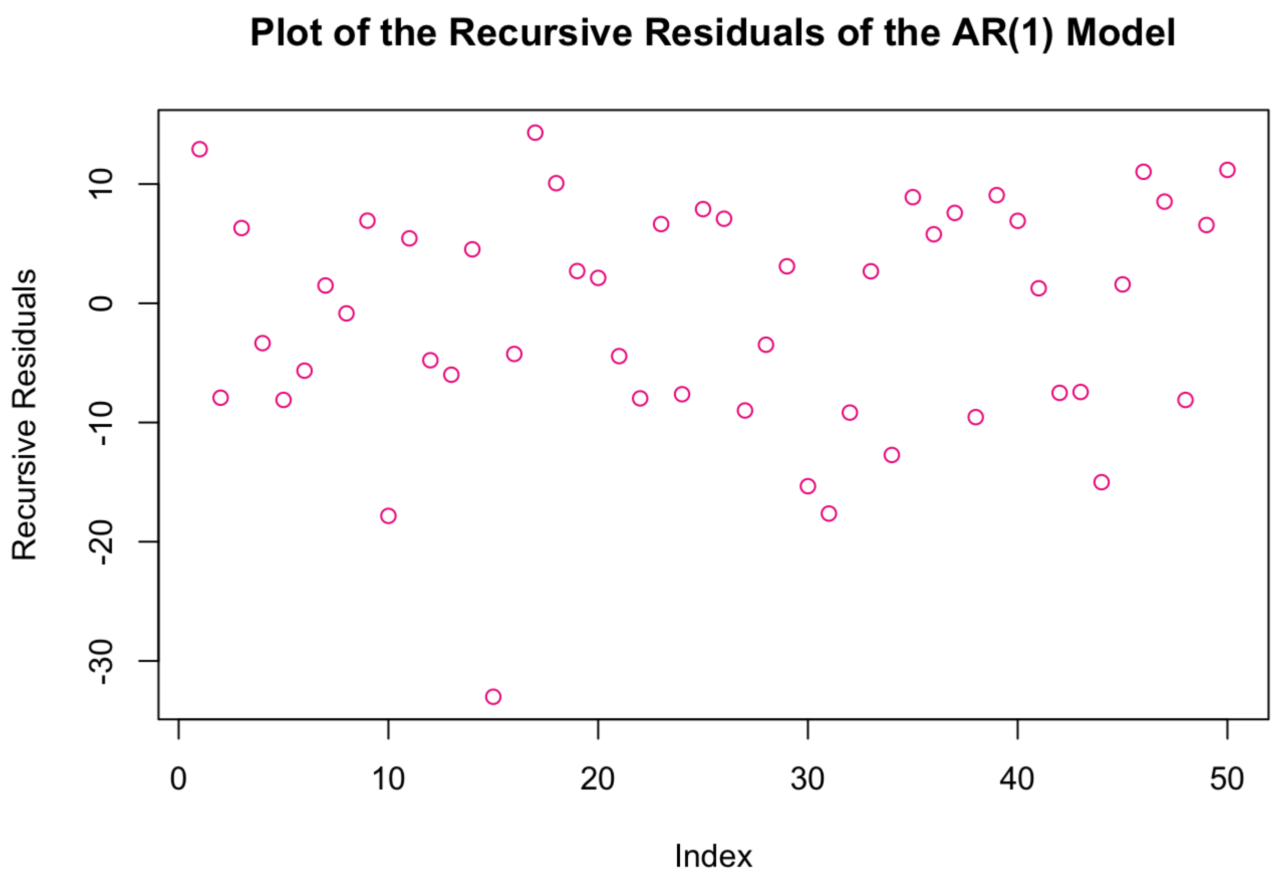
```
library(strucchange)
plot(efp(ar3_model$residuals~1, data = houston_rockets_wins, type = "Rec-CUSUM"), col = "goldenrod")
```



This model seems to be stable. We see that the golden line is bounded by the two red lines. This means that there are no structural breaks in the model. Adding a few observations will not destroy the performance of the model. This is important because we want the model to be be sturdy and able to take in many observations.

Next we will check another form of the residuals, the recursive residuals. Like the normal residuals, we want these to be centered at zero.

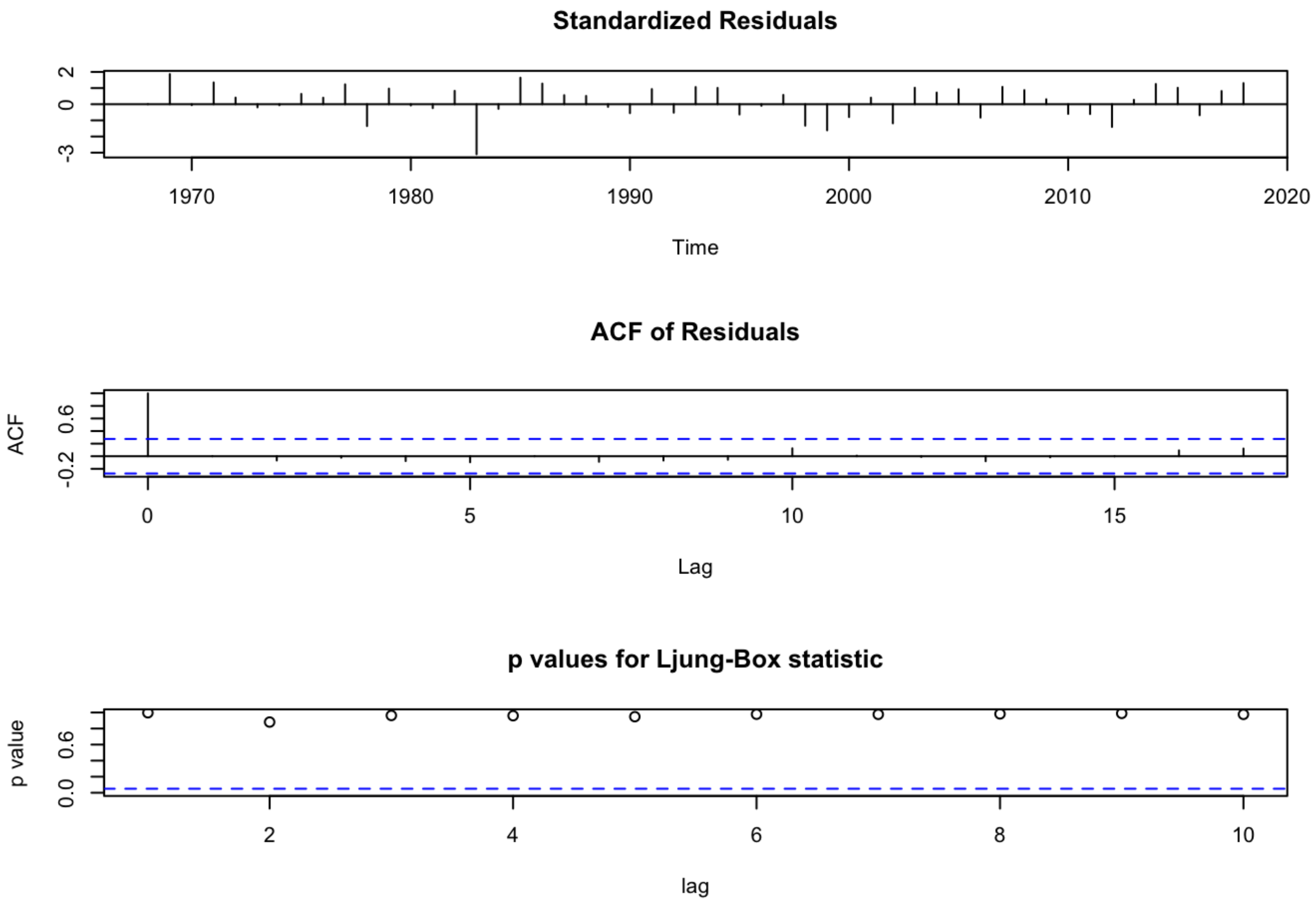
```
win_recursive_residuals <- recresid(ar3_model$residuals~1)
plot(win_recursive_residuals, ylab = "Recursive Residuals", main = "Plot of the Recursive Residuals of the AR(1)
 Model", col = "deeppink2")
```



Here we see that the recursive residuals are roughly centered at zero, perhaps the mean is slightly greater than zero.

We will summarize the diagnostics of the AR(3,1,0) model with the tsdiag. This will output the graph of the residuals, ACF and PACF of the residuals and the p-values.

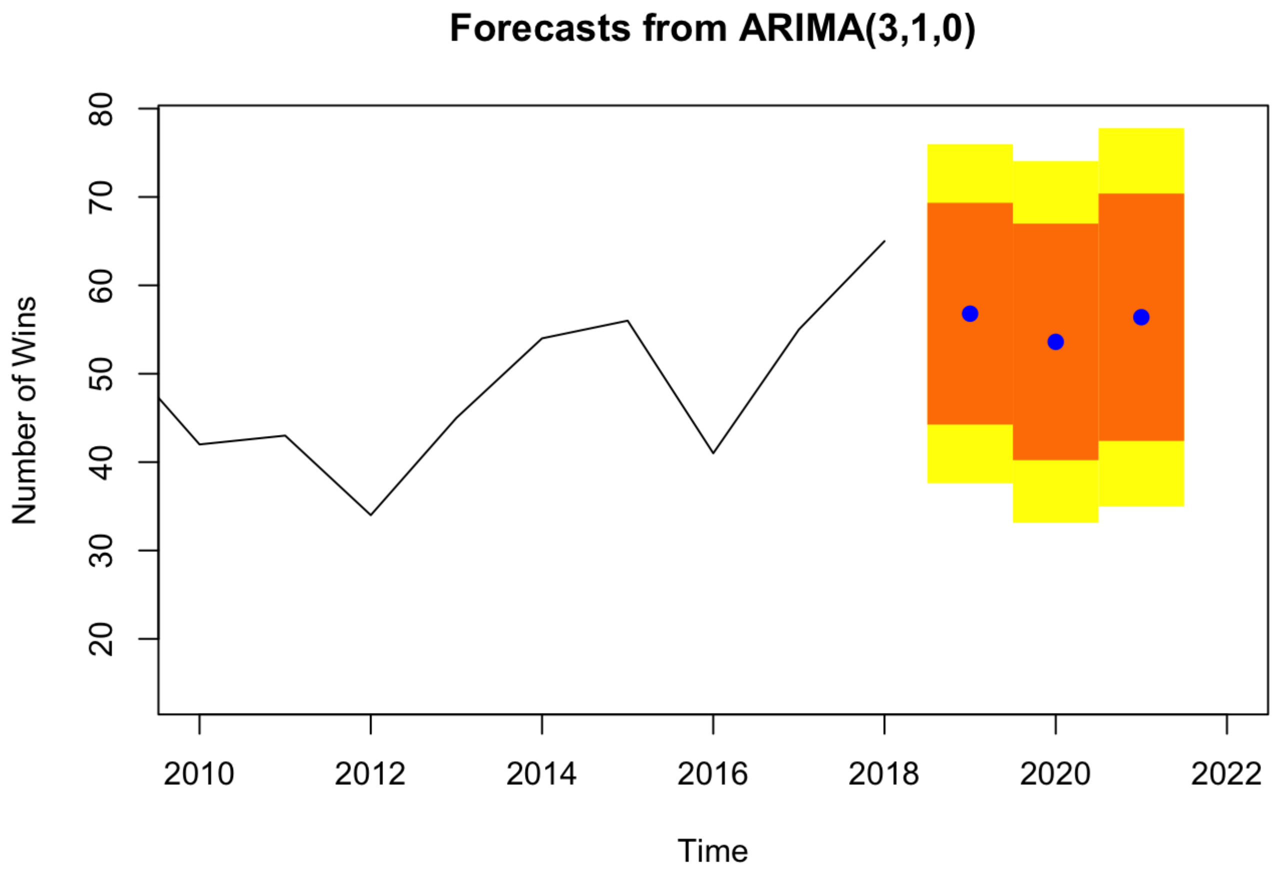
```
tsdiag(ar3_model)
```



Here we see that the residuals are centered at zero, and that the p-value remains high, indicating that the previous number of wins are less and less significant to the current number of wins.

Finally will conduct a three step forecast to examine the number of wins over the next three seasons.

```
plot(forecast(ar3_model, h = 3), shadecols = "oldstyle", xlim = c(2010,2022), xlab = "Time", ylab = "Number of Wi
ns")
```



We see a small drop in number of wins over the next three seasons. This result makes intuitive sense because winning over 60 games in the NBA is usually very rare. In addition, the players on the team become older and their performace usually dwindels. Finally, it is difficult for a team to retain all of its players because of the hard salary cap. Usually an NBA team will not be able to allocate its monetary resources to all the players satisfaction, which will cause the players to find a new team that will pay them more.

It is also worth noting that the error bands are quite large. This means that there the range of potential regular season wins is rather big.

Conclusion

Using an ARIMA model, specifically an ARIMA(3,1,0) model, we are able to successfully conduct a 3-step ahead forecast with the number of regular season wins. As mentioned before, there is a small drop off in the number of wins over the next three seasons. Despite this, it is safe to assume that the Houston Rockets will remain in the top tier of NBA teams, and a huge threat to win the NBA Championship.

References

“Houston Rockets Franchise Index”, basketball-reference.com, Sports Reference, https://www.basketball-reference.com/teams/HOU/.