Steam Online analysis project

2020/5/28

Contents

Abstract 1

Introduction 1

Data source and description 2

Data Explorarotory Analysis 2

Model diagnostic 7

Forecasting the future players number 12

Conclusion 14

References 15

Appendix: Rcode 15

Abstract

In this project I choose the Steam online gamers (Monthly average onlines) from the Steam data. My question is to find a proper model to fit the players data, and forecast the future online players for several months.

In this project, I will first analysis the stationality and seasonality using exploratory plots, and use necessary transformations to make the data meet the ARIMA model assumption. Then I would use the sample ACF and PACF plot to find the candidate models. I would select the best candidate model using AICc, and then diagnostic the residual for its normality, independence and check the characteristic root of the model. If all of the tests pass, I would forecast the future players number use the model.

Introduction

Steam is a video game digital distribution service by Valve. The Steam platform is the largest digital distribution platform for PC gaming, holding around 75% of the market space in 2013. By 2017, users purchasing games through Steam totaled roughly US$4.3 billion, representing at least 18% of global PC game sales. By 2019, the service had over 34,000 games with over 95 million monthly active users.

The success of steam is because Steam changed the way people playing games. At the time of Steam's release, around 2003, the only way you could acquire PC games was to go to your local store and get the physical disc yourself. But with Steam, it's always a few clicks and a bit of online cash away from any game at any time. And it was completely legal. Also, Steam platform feeds tons of small game studios, as not all of the small game studios can afford the distribution cost.

Data source and description

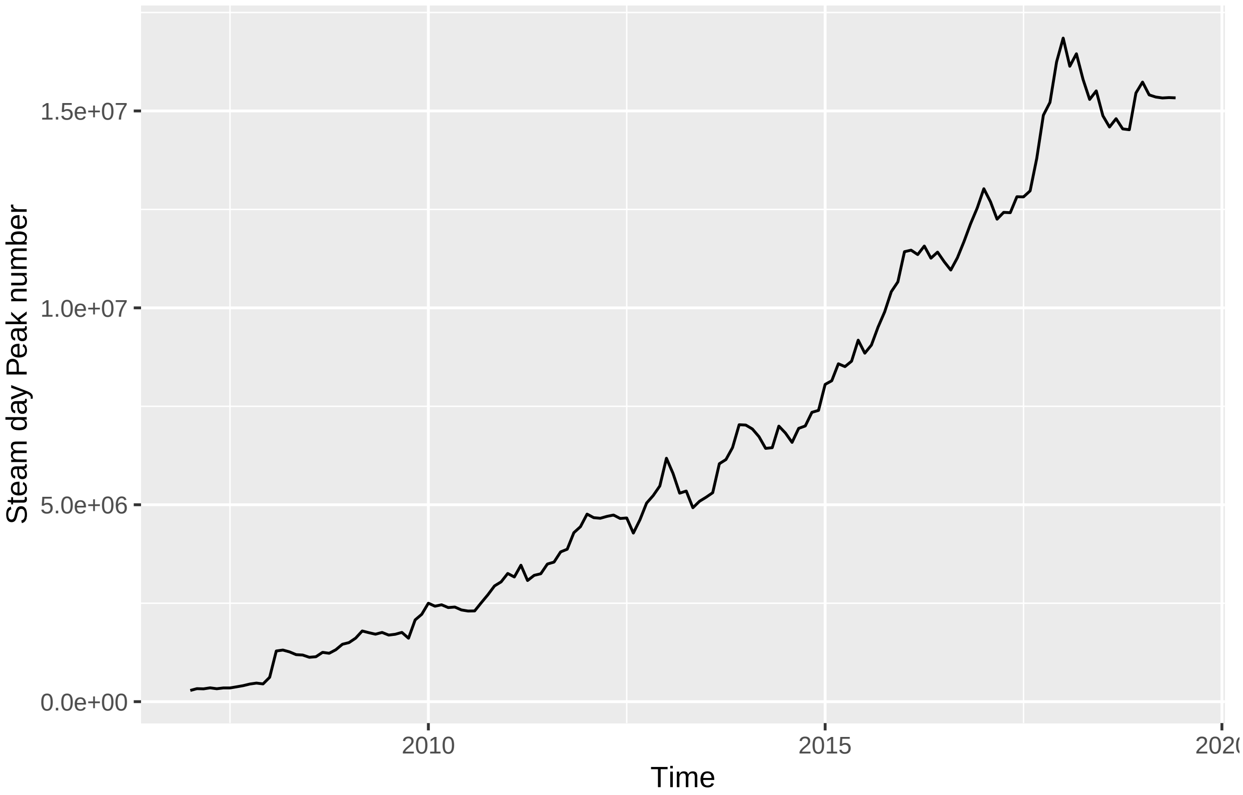
The data is from the Steam data base: https://steamdb.info/app/753/graphs/ It starts from 2007. Read in the data and it is:

```
##           Jan      Feb      Mar      Apr      May      Jun
## 2007 284567.8 330409.3 325230.4 351102.0 328161.8 348647.0
```

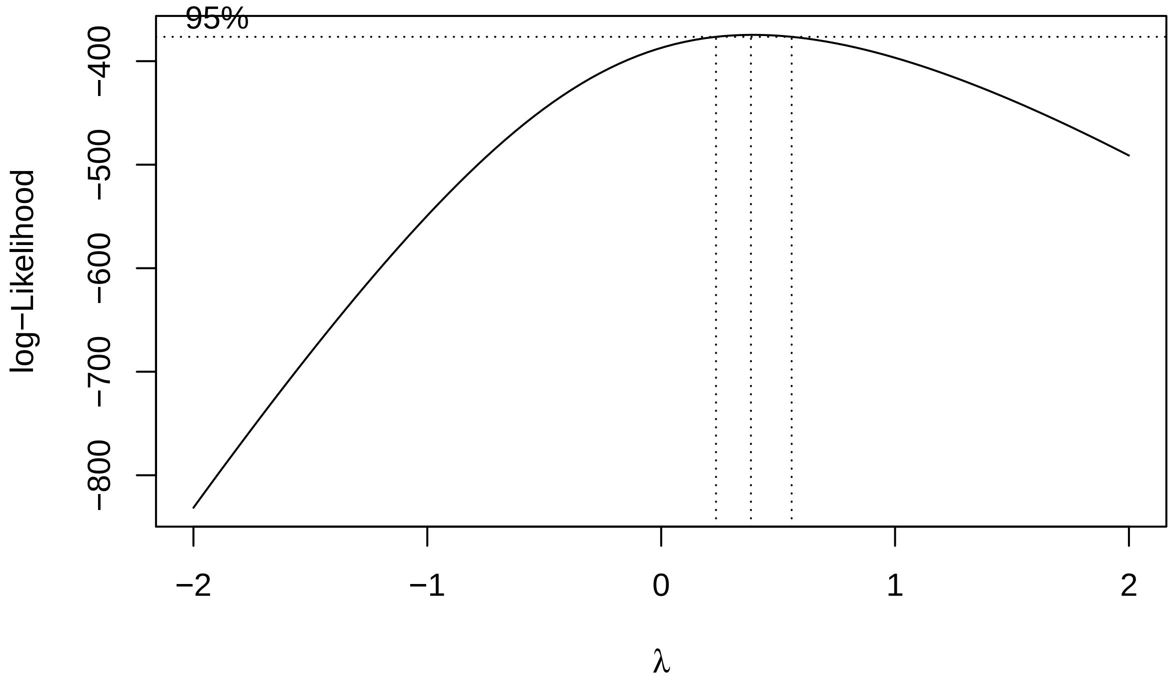
So it is from 2007 Jan to 2020 May. The monthly average is directly obtained from the sample mean of every available day peak number.

Data Explorarotory Analysis

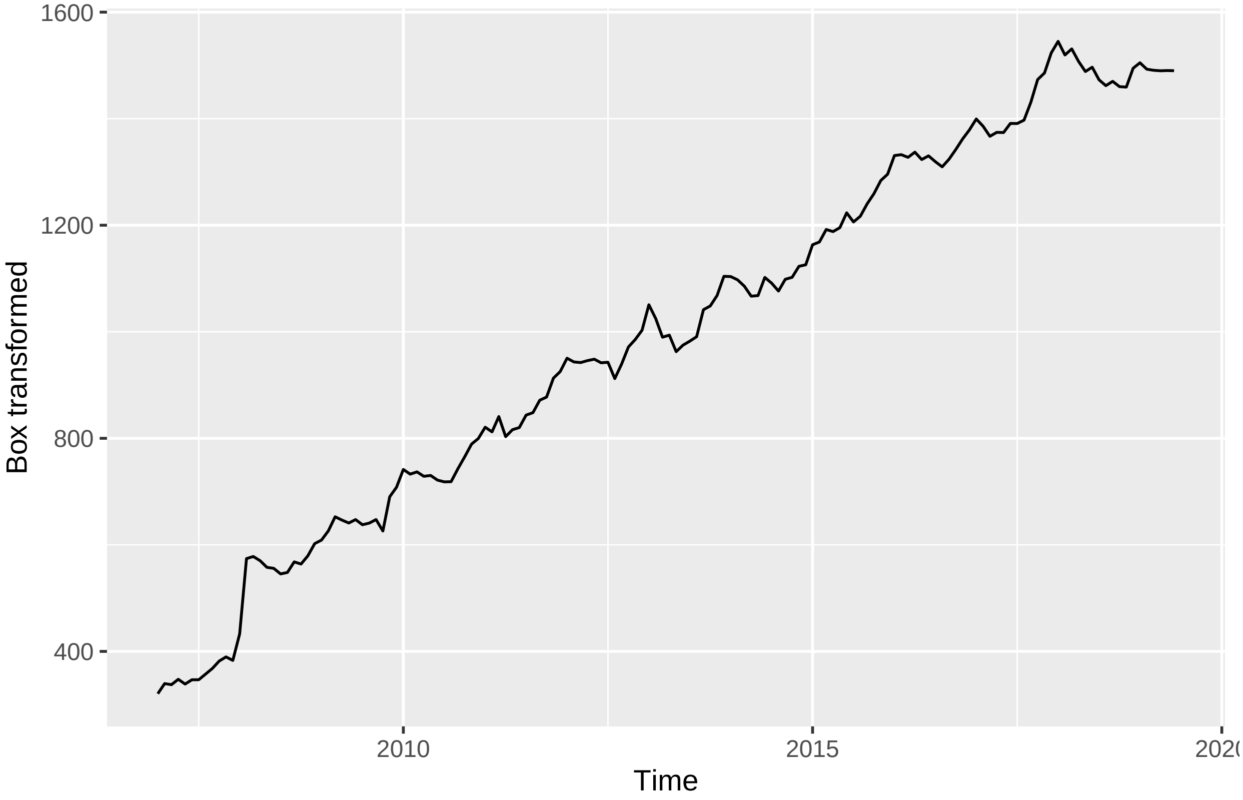
The full time series plot is presented here:



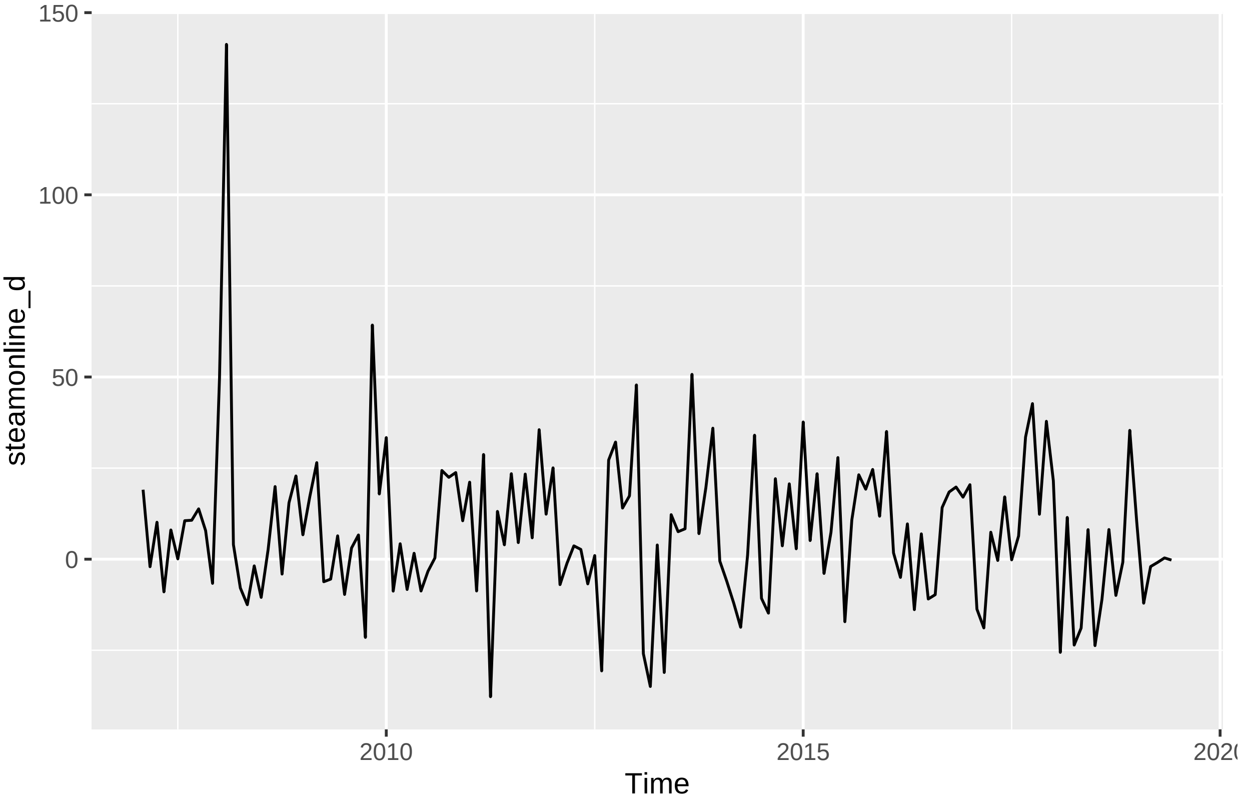
The steam online players is continuously increasing since 2007. And it's increasing faster. There is also some decrease in some years, but not significant compared to the increase. From this plot I can't see obvious seasonal effect. So I will first check if I need to apply Box-Cox transformation.



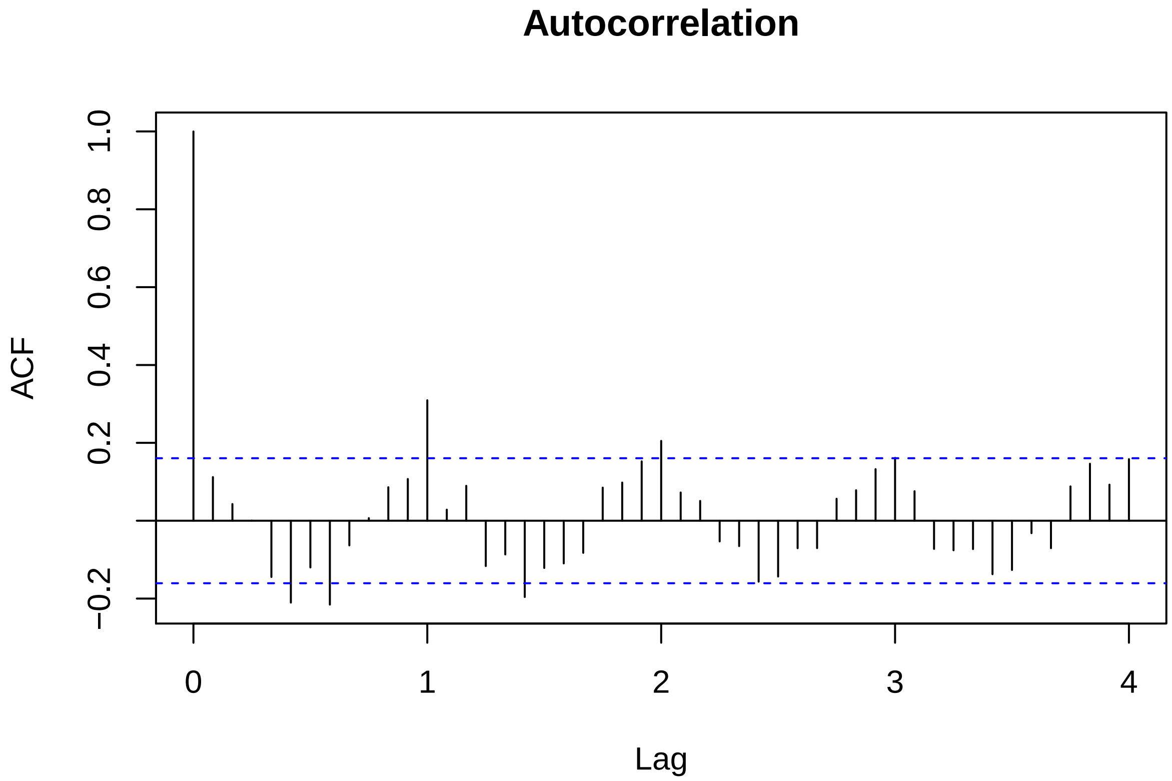
From the result plot, the 95% confidence interval does not contain 1, so I apply the Box-cox transformation on the data to make the variance equal and meet the assumption of ARIMAmodel. Now I check the transformed data:

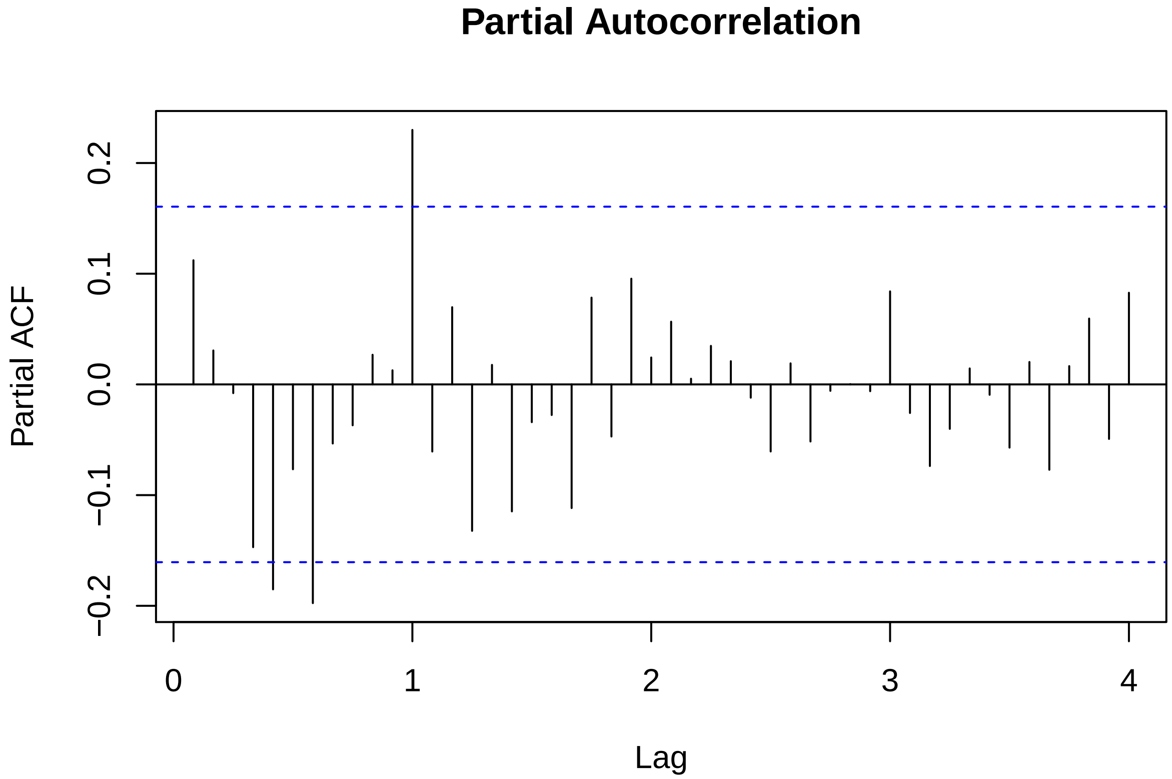


Then I apply an 1-lag difference:



Except some huge spike in 2008 and 2020 (due to the financial crisis and COVID-19 outbreak, so maybe more people are staying at home playing video games?), the series seems stationary. So now I plot the ACF and PACF of the differenced data:


Autocorrelation


Partial Autocorrelation

The ACF tails off (or cut off at 5 or 7), and the PACF cut off at lag 5 or 7 in the first 12 circle. This sign suggest the ARIMA part MA(0) or MA(5), MA(7) from the ACF, AR(0) or AR(5) or AR(7) from the PACF. ARIMA(7,1,7) is not fitted because R can't fit the model with so many parameters for this data. For the seasonal part, we have marginally significant spikes at lag 12, so consider SARIMA part (1,0,0), (1,0,1) or (0,0,1), (1,0,2), (0,0,2). So I proceed to fit the models and calculate the AICc to do the model selection. AICC = 5678 means can't fit in R.

Table 1: The AICc of different candidate models

	(1,0,0)	(1,0,1)	(0,0,1)	(1,0,2)	(0,0,2)
SARIMA(0,1,0)	1328.858	1319.449	1335.570	1321.532	1331.488
SARIMA(5,1,0)	1334.325	1328.665	1338.545	1330.813	1337.163
SARIMA(7,1,0)	1338.199	1331.631	1342.442	1333.890	1341.101
SARIMA(0,1,5)	1334.665	1328.803	1338.869	1330.952	1337.416
SARIMA(5,1,5)	1336.391	1335.673	1343.981	1337.609	1340.116
SARIMA(7,1,5)	1342.512	1340.779	1350.157	1343.098	1350.088
SARIMA(0,1,7)	1337.455	1331.775	1341.534	1333.919	1340.463
SARIMA(5,1,7)	1341.663	1338.532	1350.956	5678.000	1347.620

So from the result I conclude that the best candidate model is the $SARIMA(0,1,0) \times (1,0,1)_{12}$, and the fitted result is:

```
## Series: steamonline_1
## ARIMA(0,1,0)(1,0,1)[12]
##
## Coefficients:
##         sar1     sma1
##       0.9190  -0.6932
## s.e.  0.0602   0.1195
##
## sigma^2 estimated as 383.8:  log likelihood=-656.64
## AIC=1319.28   AICc=1319.45   BIC=1328.3
##
## Training set error measures:
##                    ME     RMSE      MAE      MPE     MAPE      MASE       ACF1
## Training set 2.260424 19.39454 13.39084 0.389759 1.593934 0.1327508 0.04068039
```

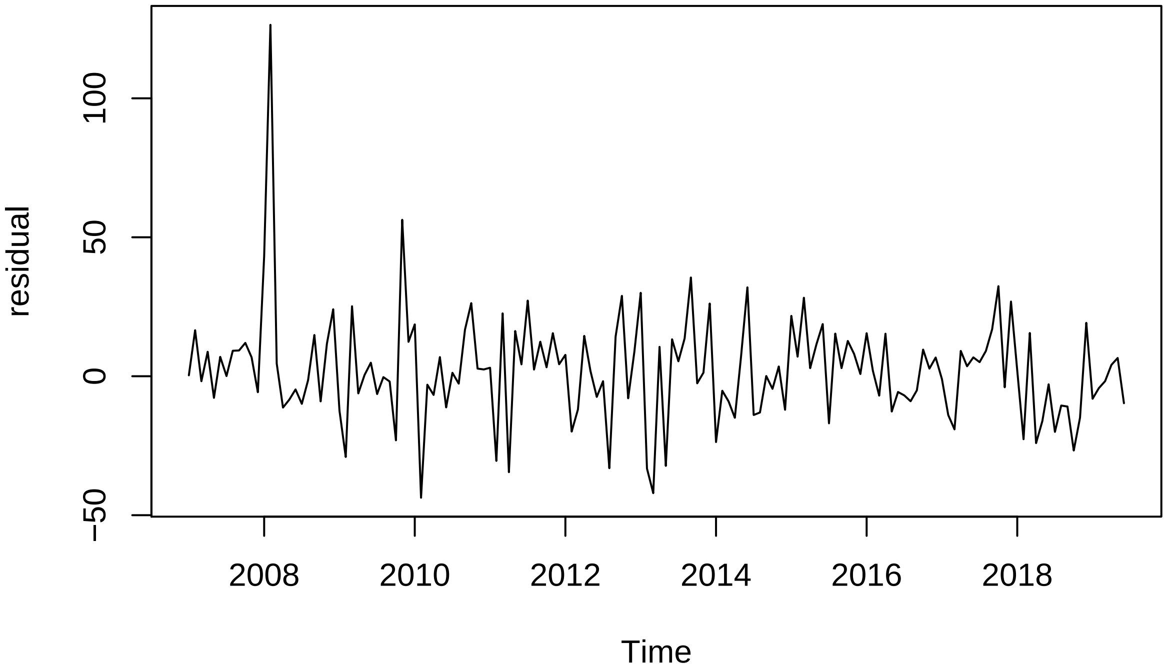
So the formula of the model (after box cox transformation) is:

$$X_n(1-B)(1-0.9190B^{12}) = Z_n(1-0.6932B^{12})$$

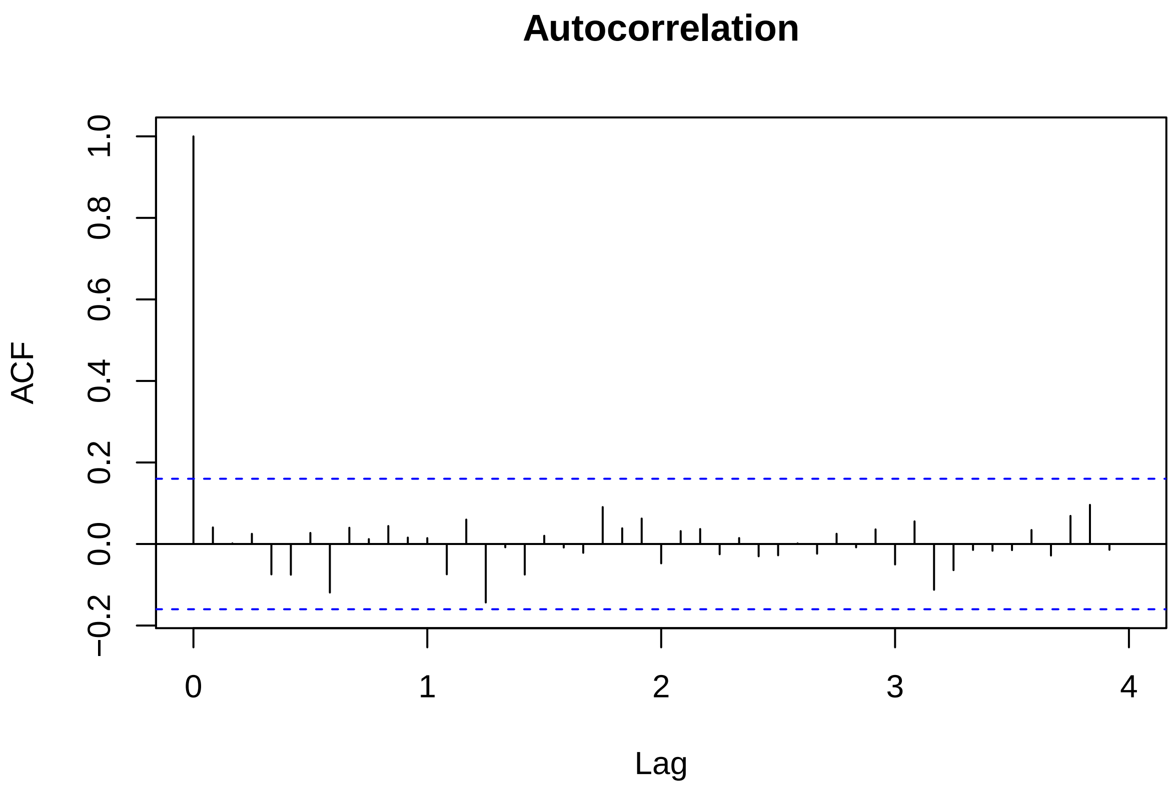
$$X_n - X_{n-1} - 0.919X_{n-12} + 0.919X_{n-13} = Z_n - 0.6932Z_{n-12}, Z_n \overset{iid}{\sim} N(0, 383.8)$$

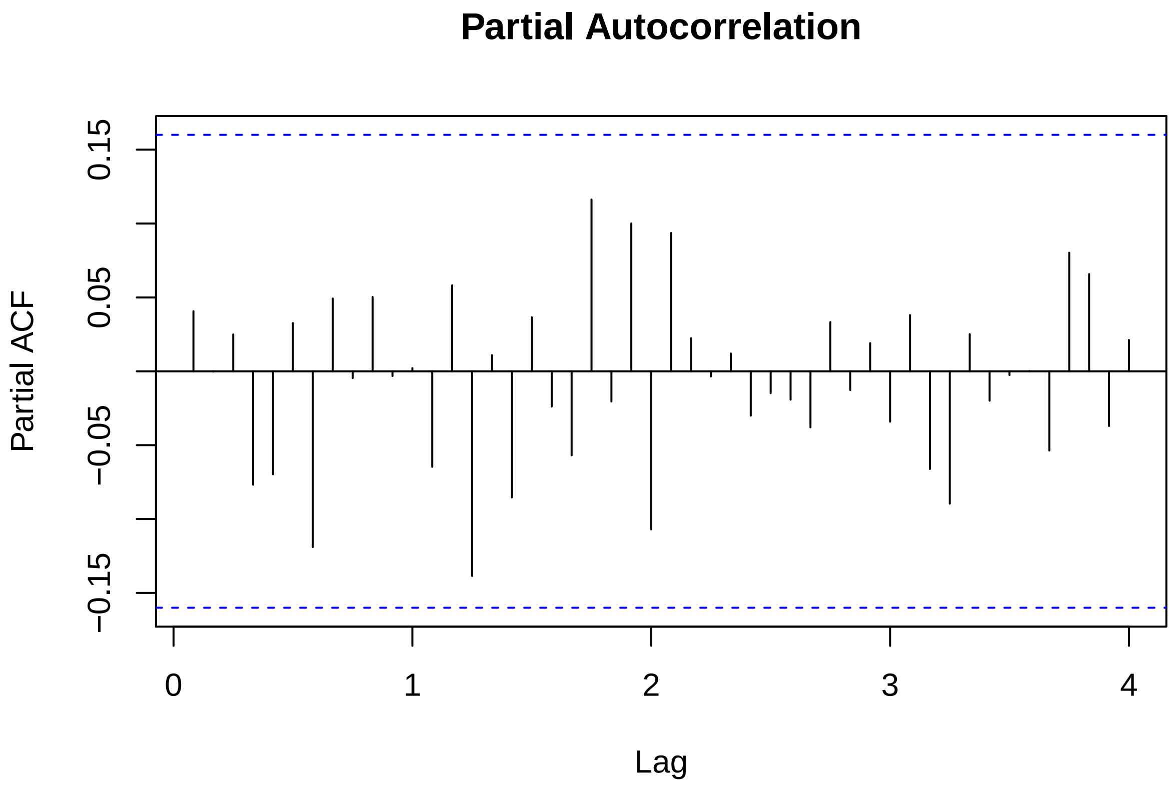
Model diagnostic

First I plot the residual as below:



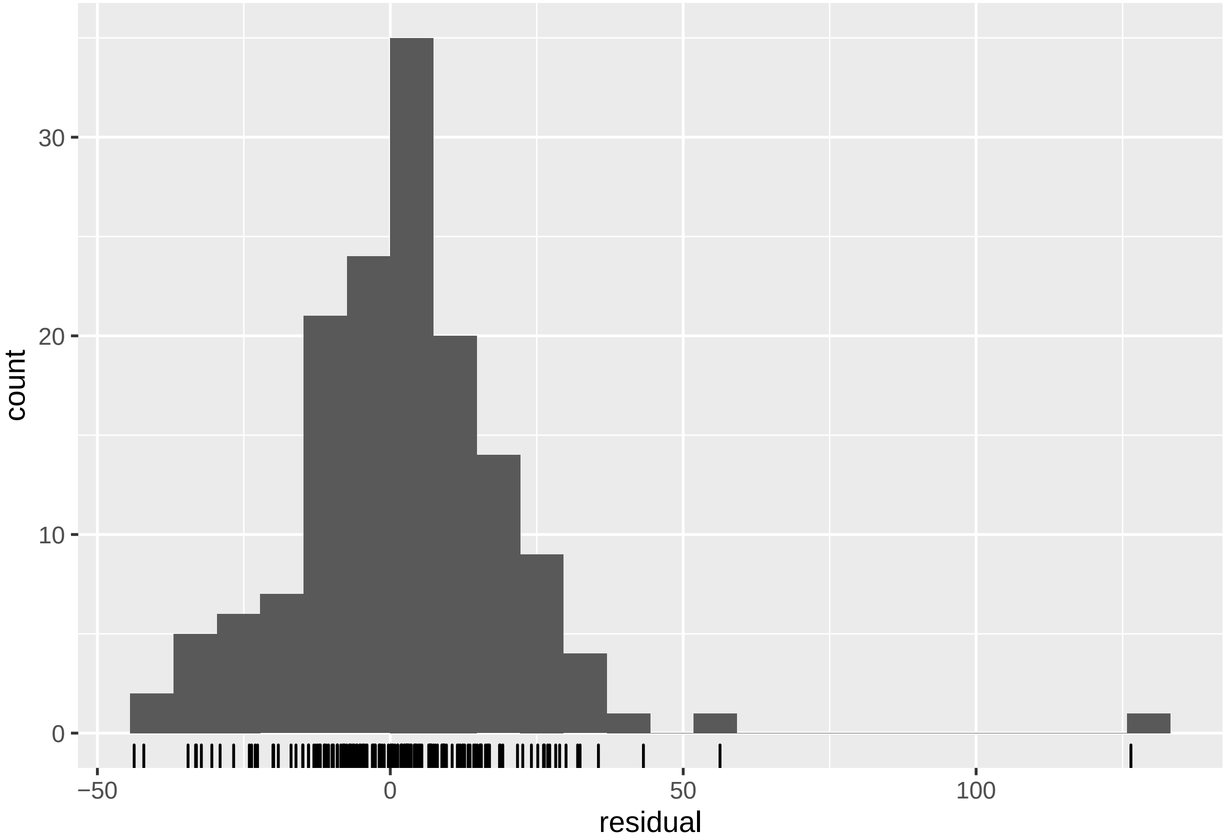
And I plot its sample ACF And PACF:

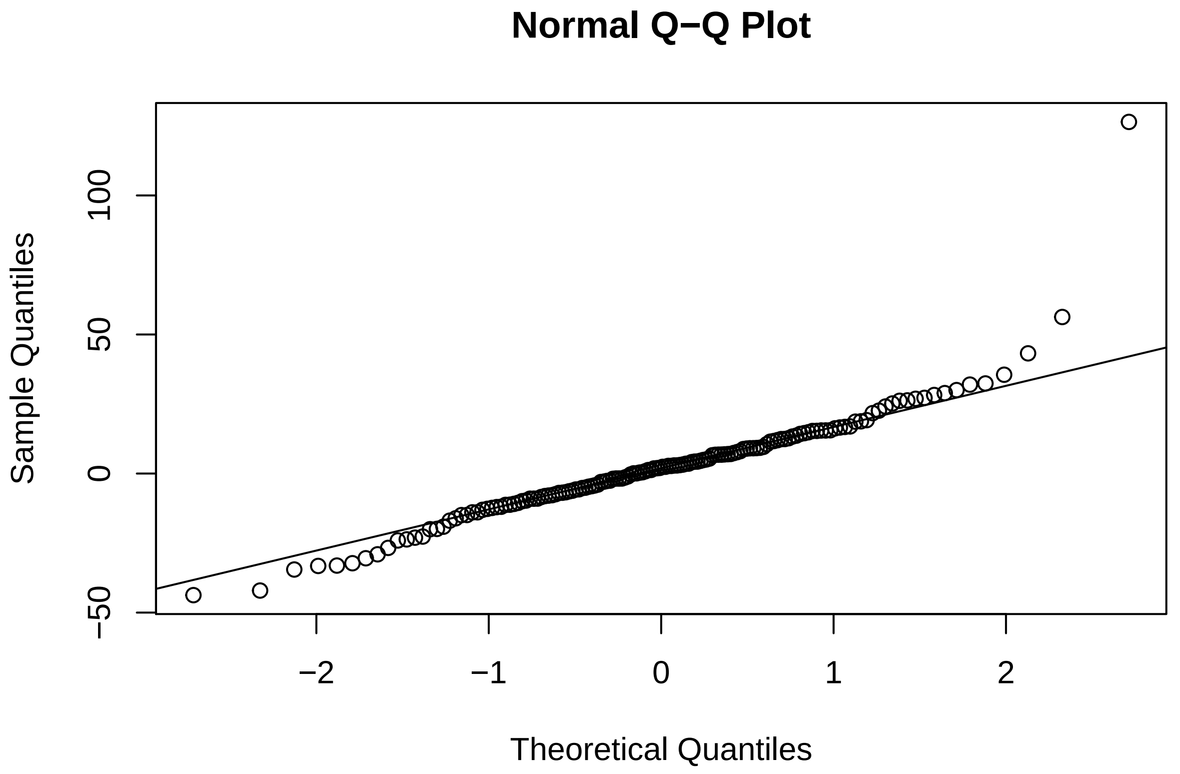

Autocorrelation
ACF
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0
1
2
3
4
Lag


Partial Autocorrelation
Partial ACF
0.15
0.05
−0.05
−0.15
0
1
2
3
4
Lag

From the sample ACF and PACF plots, there is no significant spikes. So there is no sign for further SARIMA parts to consider.

Then I check the normality of the residual:

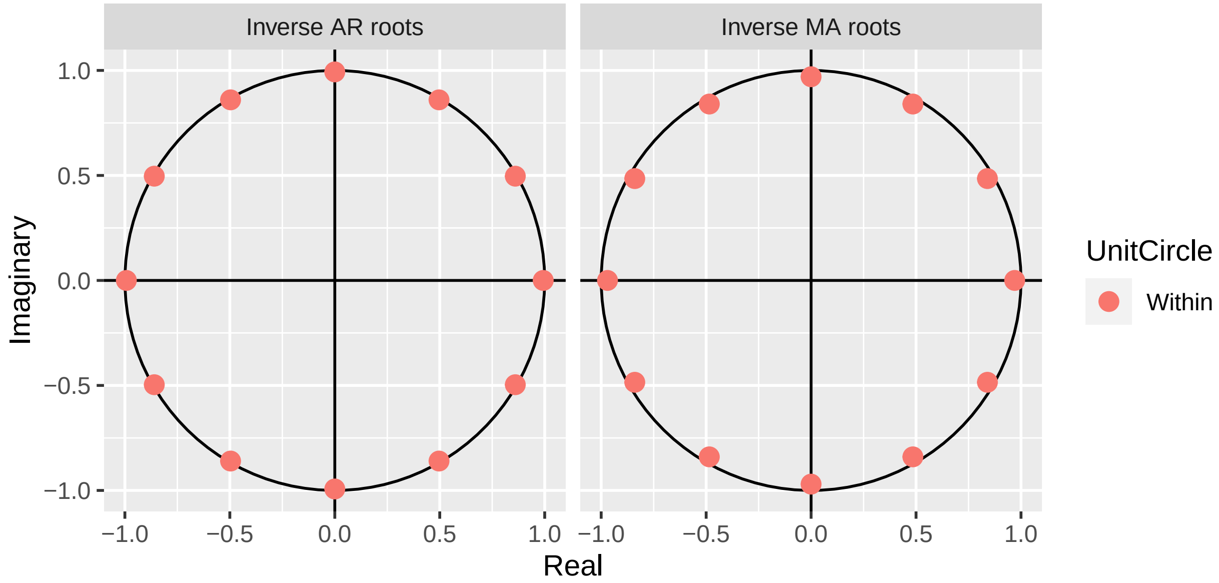




```
##
##  Shapiro-Wilk normality test
##
## data:  residual
## W = 0.88965, p-value = 3.588e-09
```

The Shapiro-Wilk test was rejected, so there is some non normality in the residual. The histogram indicates that there is some residual quite big to make the distribution unnormal. The QQ plot also suggest this, and if we subtract those outliers, other points fit the line quite well. So I think except for the outliers, the residual is normal.

Then I check the characteristic root.



As all of the inverse unit roots are inside the unit circle, the model is stationary plus invertible. Those rootes are from the seasonal part (order 12 means 12 roots.)

Then I check the residual dependence, the data has 161 observations, so the degree of freedom of the Ljung Box and Box pierce are $13 - 2 = 10$, and the degree of freedom of the Mcleod Li test is 13.

- Box - Pierce test

```
## [1] 0.8914165
```

- Ljung - Box test

```
## [1] 0.8676882
```

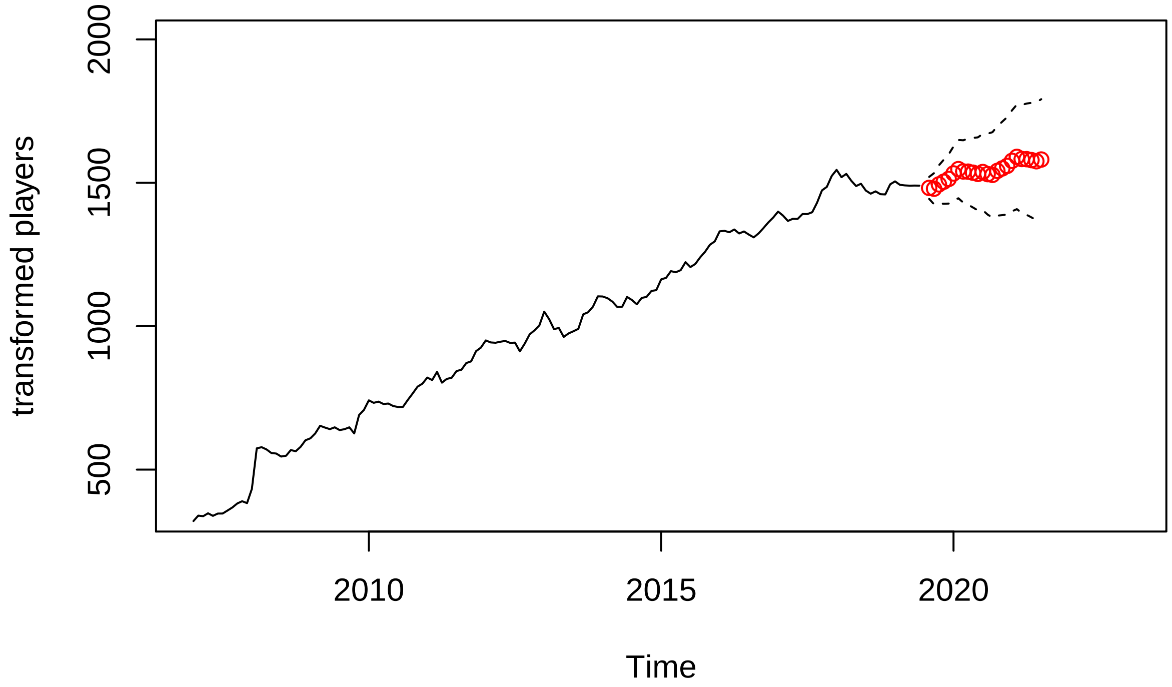
- McLeod - Li test

```
## [1] 0.9996716
```

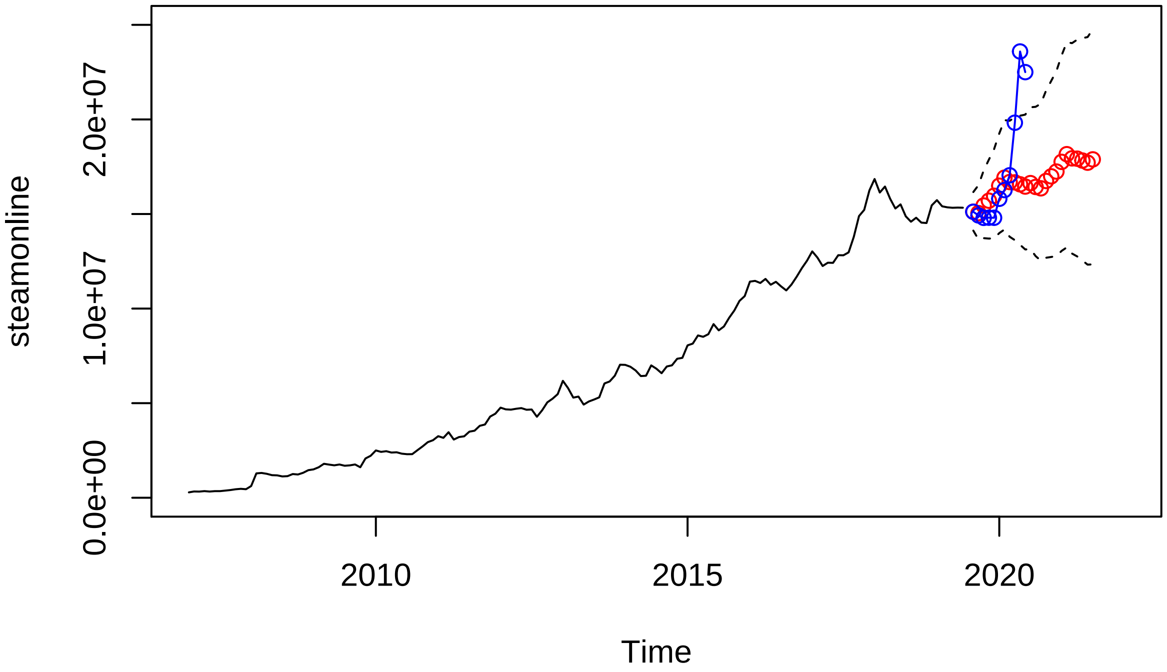
All of the independence tests are passed. But I also note that the Mcleod Li test has a much smaller p value then the other two, that might resulted from the outliers in 2008 or 2020.

Forecasting the future players number

Now as all of the tests are passed, I can proceed to forecast the future data. I will forecast the following two years. First I obtain the forecast and the interval after the box transformation:



Then I get the plot before the box cox transformation:



It seems the true online players number is inside the 95% confidence interval in the prediction of 2019, but not after 2020 Feb. I think the prediction not in the CI in 2020 is not a sign of failure of the model. That is more because of the COVID-19. The 2019 observations are still inside the 95% confidence interval, which means the model is good. The last three has a huge boost, because the COVID-19 outbreak, people stay at home and the popular of Steam is not a surprise. So I think that is not the failure of the model, it is because of the outbreak of COVID.

Conclusion

In this project I analyzed the Steam online gamers (Monthly average onlines) from the Steam database. I find the proper model

$$X_n - X_{n-1} - 0.919X_{n-12} + 0.919X_{n-13} = Z_n - 0.6932Z_{n-12}, Z_n \overset{iid}{\sim} N(0, 383.8)$$

To fit the players data, and forecast the future online players for the next two years.

From the exploratory analysis, I apply the box cox transformation and then I apply a lag 1 difference to make the data meet the assumption of the SARIMA model. Then I would use the sample ACF and PACF plot to find the candidate models. After calculate those models using MLE and obtain their AICc, I find the best model, and then diagnostic the residual for its ACF,PACF, normality, independence and check the characteristic root of the model. The residual pass all of the tests except for the Shapiro because of the outliers. So I conclude it's a god model and proceed to the forecast. The forecast result shows that the average players will increase over the next two years. Though the true value is not in the forecast interval, that is because of the outbreak of COVID 19 and more people rely Steam to stay at home for entertainment.

References

1. - [Steam (service) From Wikipedia, the free encyclopedia] (https://en.wikipedia.org/wiki/Steam_(service))
2. - [Steam Database] (https://steamdb.info/app/753/graphs/)
3. PSTAT 174 Spring 2020 Lecture slides, Labs

Appendix: Rcode

```r
knitr::opts_chunk$set(
    echo = FALSE,
    error = FALSE,
    message = FALSE,
    warning = FALSE
)
library(forecast)
library(MASS)
library(car)
library(tseries)
data = read.table("steamonline.csv",header = TRUE)
steamonline = ts(data[1:150,1],frequency=12,start = c(2007,1))
head(steamonline)
steamonline_test = data[151:161,1]
autoplot(steamonline,ylab = "Steam day Peak number")
#Box cox transformation
lamb = boxcox(steamonline~1)
optlamb = lamb$x[which.max(lamb$y)]
steamonline_1 = (steamonline^optlamb - 1)/optlamb
autoplot(steamonline_1,ylab = "Box transformed")
steamonline_d = diff(steamonline_1,lag=1)
autoplot(steamonline_d)
par(mfrow = c(2,1))
# acf
acf(steamonline_d,main = "Autocorrelation",lag.max = 48)
# pacf
pacf(steamonline_d,main = "Partial Autocorrelation",lag.max = 48)
#model selection
AICc = 5678
arimapart = list(c(0,1,0),c(5,1,0),c(7,1,0),c(0,1,5),c(5,1,5),
                 c(7,1,5),c(0,1,7),c(5,1,7))
seasonalpart = list(c(1,0,0),c(1,0,1),c(0,0,1),c(1,0,2),c(0,0,2))
fits = list()
flag = 0
AICcs = matrix(0,length(arimapart),length(seasonalpart))
for(i in 1:length(arimapart)){
  for(j in 1:length(seasonalpart)){
    if(i == 8 & j == 4){
      AICcs[i,j] = 5678
    }
    else{
          fits[[i*length(arimapart)+j]] = Arima(steamonline_1,order=arimapart[[i]],
                    seasonal = seasonalpart[[j]],method = "ML")
    AICcs[i,j] = fits[[i*length(arimapart)+j]]$aicc
    if(fits[[i*length(arimapart)+j]]$aicc < AICc){
      AICc = fits[[i*length(arimapart)+j]]$aicc
      flag = c(i,j)
    }
    }

  }
}
rownames(AICcs) = c("SARIMA(0,1,0)","SARIMA(5,1,0)","SARIMA(7,1,0)",
                    "SARIMA(0,1,5)","SARIMA(5,1,5)","SARIMA(7,1,5)",
                    "SARIMA(0,1,7)","SARIMA(5,1,7)")
colnames(AICcs) = c("(1,0,0)","(1,0,1)","(0,0,1)","(1,0,2)","(0,0,2)")
knitr::kable(AICcs, caption = "The AICc of different candidate models")
fit = Arima(steamonline_1,order=c(0,1,0),
                   seasonal = c(1,0,1),method = "ML")
summary(fit)
#residual
residual = fit$residuals
ts.plot(residual)
par(mfrow = c(2,1))
# acf
acf(residual,main = "Autocorrelation",lag.max = 48)
# pacf
pacf(residual,main = "Partial Autocorrelation",lag.max = 48)
#normality
gghistogram(residual)
qqnorm(residual)
qqline(residual)
shapiro.test(residual)
autoplot(fit)
# independence tests
Box.test(residual, lag = 13, type = c("Box-Pierce"),fitdf = 2)$p.value
Box.test(residual, lag = 13, type = c("Ljung-Box"),fitdf = 2)$p.value
Box.test(residual^2, lag = 13, type = c("Ljung-Box"),fitdf = 0)$p.value
# Forecast part
forecasted = predict(fit, n.ahead=24)
pred = forecasted$pred
se = forecasted$se
# The forecast after box transform
ts.plot(steamonline_1, xlim = c(2007,2023), ylim = c(350,2000), ylab = "transformed players")
points(2020-6/12+(1:24)/12,pred,col = "red")
lines(2020-6/12+(1:24)/12,pred+1.96*se,lty=2)
lines(2020-6/12+(1:24)/12,pred-1.96*se,lty=2)
pred1 = exp(log(pred * optlamb + 1)/optlamb)
pred1l = exp(log((pred-1.96*se) * optlamb + 1)/optlamb)
pred1u = exp(log((pred+1.96*se) * optlamb + 1)/optlamb)
ts.plot(steamonline, xlim = c(2007,2022),ylim = c(0,2.5*10^7))
points(2020-6/12+(1:24)/12,pred1,col = "red")
lines(2020-6/12+(1:24)/12,pred1l,lty=2)
lines(2020-6/12+(1:24)/12,pred1u,lty=2)
```

```r
points(2020-6/12+(1:11)/12,steamonline_test,col = "blue")
lines(2020-6/12+(1:11)/12,steamonline_test,lty=1, col = "blue")
```