DATA583_Proj

Jasmine Chen

2020-04-12

```
library(dplyr)
library(ggplot2)
library(forecast)
library(tseries)
library(timelineR)
library(dplyr)
```

```
confirmed_global <- read.csv(url("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_
dd=subset(confirmed_global, Country.Region=='Korea, South')
write.csv(dd,"korea_covid19_confirmed.csv", row.names = FALSE)
```

```
df=subset(confirmed_global, Country.Region=='Korea, South', select=-c(Province.State, Lat, Long))
confirmed=as.data.frame(colSums(df[,-1]))
names(confirmed)[1] <- 'cumConfirmed'
```

```
confirmed <- cbind(Date = rownames(confirmed), confirmed)
rownames(confirmed) <- 1:nrow(confirmed)
confirmed$Date=substring(confirmed$Date,2)
confirmed$Date <- as.Date(confirmed$Date, "%m.%d.%y")
confirmed$dailyConfirmed=c(1,diff(confirmed$cumConfirmed))
```

```
death_global <- read.csv(url("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covi
dd2=subset(death_global, Country.Region=='Korea, South')
write.csv(dd2,"korea_covid19_death.csv", row.names = FALSE)
```

```
df1=subset(death_global, Country.Region=='Korea, South', select=-c(Province.State, Lat, Long))
death=as.data.frame(colSums(df1[,-1]))
names(death)[1] <- 'cumDeath'
```

```
death <- cbind(Date = rownames(death), death)
rownames(death) <- 1:nrow(death)
death$Date=substring(death$Date,2)
death$Date <- as.Date(death$Date, "%m.%d.%y")
death$dailyDeath=c(0,diff(death$cumDeath))
```

```
test <- read.csv('full-list-cumulative-total-tests-per-thousand.csv')
df2=subset(test, Entity=='South Korea',select=-c(Entity, Code))
```

```
df2$Total.tests=round(df2$Total.tests.per.thousand/1000*51269000,0)
df2$dailyTests=c(0,diff(df2$Total.tests))
df2$Date <- as.Date(df2$Date , "%b %d,%Y")
```

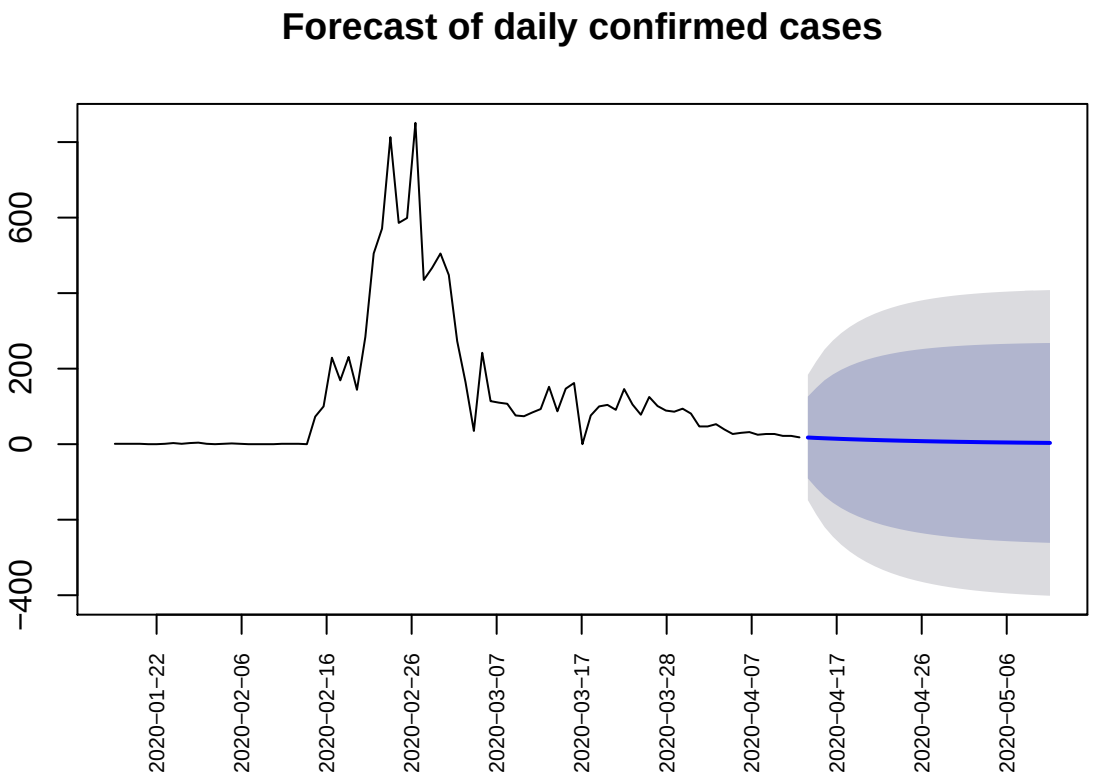
```r
korea=inner_join(inner_join(confirmed, death,by = "Date"),df2, by = "Date")
```

ARIMA daily confirmed

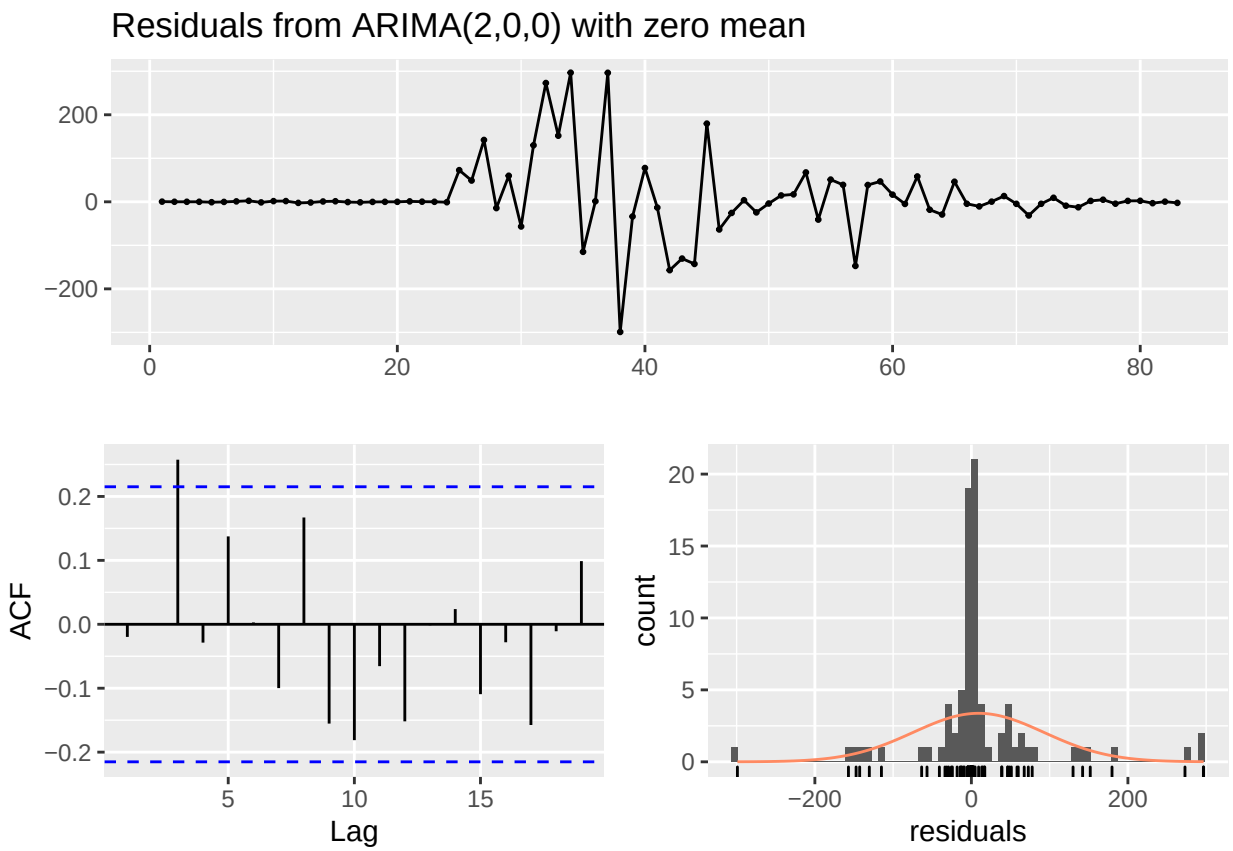
```r
fit_confirmed <- auto.arima(korea$dailyConfirmed, seasonal = FALSE)
fit_confirmed
```

```
## Series: korea$dailyConfirmed
## ARIMA(2,0,0) with zero mean
##
## Coefficients:
##          ar1     ar2
##       0.7003  0.2304
## s.e.  0.1056  0.1055
##
## sigma^2 estimated as 7160:  log likelihood=-486.06
## AIC=978.12   AICc=978.42   BIC=985.38
```

```r
predictionTime=30
preds <- forecast(fit_confirmed, h=predictionTime)
dates<-c(as.Date(korea$Date),as.Date(seq(as.Date(korea$Date[nrow(korea)]),by="day",length.out = predicti
plot(preds,xaxt='n',main=("Forecast of daily confirmed cases"))
at  <-  seq(1,nrow(korea)+predictionTime,length.out=12)
axis(1, at = at+5, labels = dates[at],cex.axis = .7,las=3)
```



```
checkresiduals(fit_confirmed)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,0,0) with zero mean
## Q* = 16.69, df = 8, p-value = 0.0335
##
## Model df: 2.   Total lags used: 10
```

```
# Evaluating predictive performance using the last 5 points
(pred5=forecast(auto.arima(korea$dailyConfirmed[1:77], seasonal = FALSE),h=5))
```

```
##    Point Forecast      Lo 80    Hi 80     Lo 95    Hi 95
## 78       29.29957  -83.39877 141.9979 -143.0577 201.6568
## 79       27.86988 -109.68726 165.4270 -182.5056 238.2454
## 80       26.24785 -133.45511 185.9508 -217.9967 270.4924
## 81       24.78363 -151.62600 201.1933 -245.0116 294.5789
## 82       23.38560 -166.81934 213.5905 -267.5078 314.2790
```

```
accuracy(korea$dailyConfirmed[78:82], pred5$mean)
```

```
##                ME     RMSE      MAE      MPE     MAPE       ACF1 Theil's U
## Test set 1.717306 2.428052 2.018166 6.417383 7.563611 -0.2092487  1.151091
```

Evaluating predictive performance: Compute the accuracy measures of the last 5 points against forecast, 'Point Forecast' has the lowest ME, RMSE, MAPE, ACF1 and Theil's U.

ARIMA daily confirmed/ daily tests

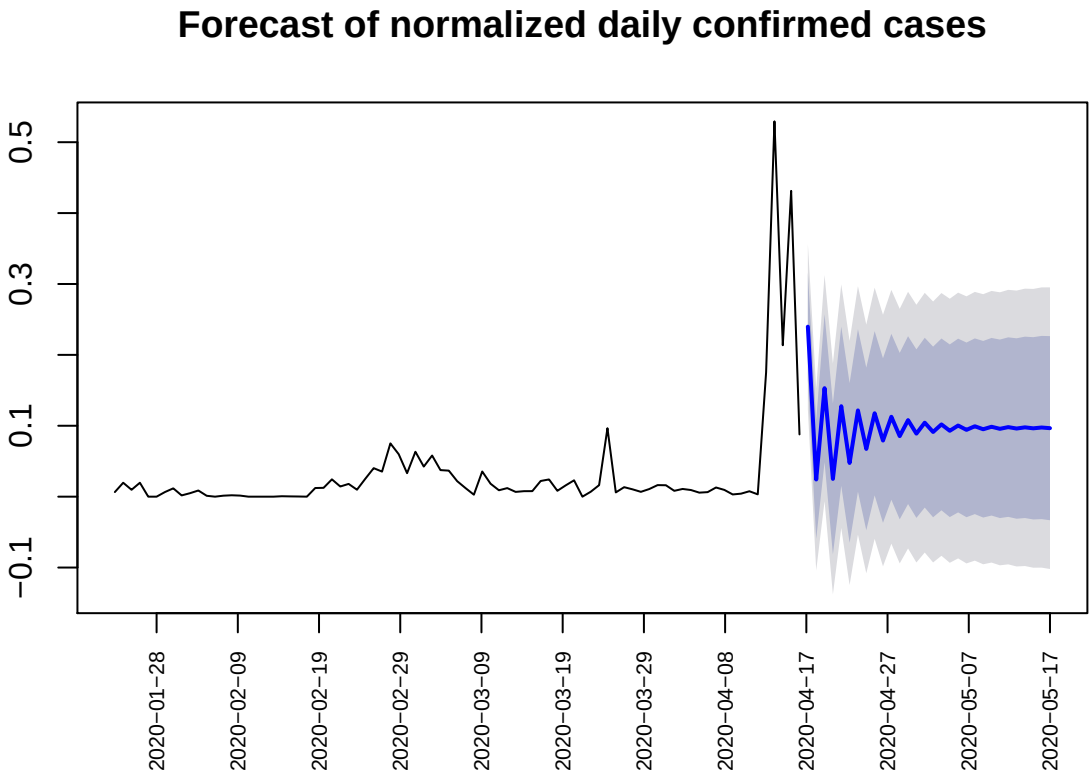
```
fit_confirmed_n <- auto.arima(korea$dailyConfirmed/korea$dailyTests[5:82], seasonal = FALSE)
```

```
## Warning in korea$dailyConfirmed/korea$dailyTests[5:82]: longer object length is
## not a multiple of shorter object length
```

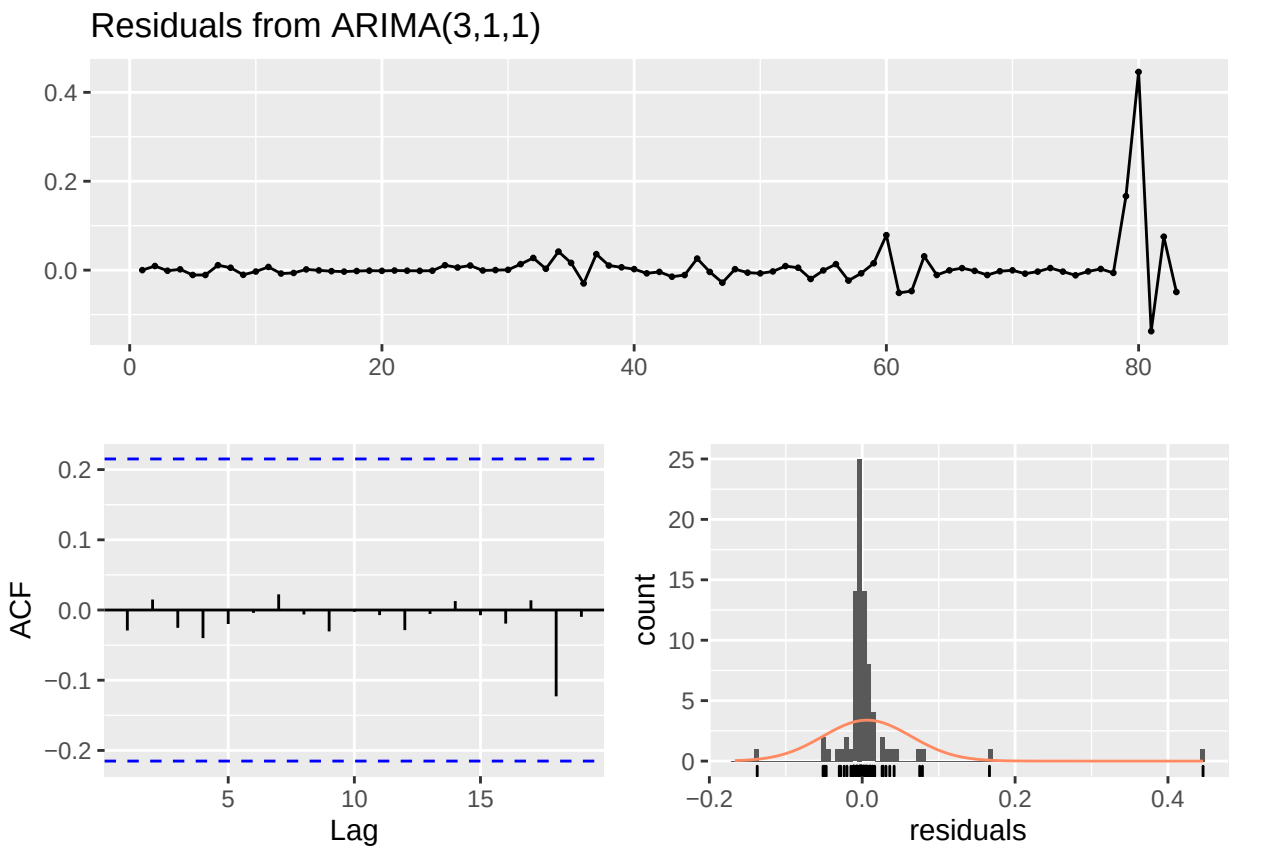
```
fit_confirmed_n
```

```
## Series: korea$dailyConfirmed/korea$dailyTests[5:82]
## ARIMA(3,1,1)
##
## Coefficients:
##          ar1     ar2      ar3      ma1
##       0.3870  0.5658  -0.3674  -0.9293
## s.e.  0.1568  0.1364   0.1282   0.1232
##
## sigma^2 estimated as 0.003538:  log likelihood=116.34
## AIC=-222.67   AICc=-221.88   BIC=-210.64
```

```
predictionTime=30
preds_n <- forecast(fit_confirmed_n, h=predictionTime)
dates<-c(as.Date(korea$Date[5:82]),as.Date(seq(as.Date(korea$Date[nrow(korea)]),by="day",length.out = pr
plot(preds_n,xaxt='n',main=("Forecast of normalized daily confirmed cases"))
at  <-  seq(1,nrow(korea[5:82,])+predictionTime,length.out=12)
axis(1, at = at+5, labels = dates[at],cex.axis = .7,las=3)
```




```
checkresiduals(fit_confirmed_n)
```



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(3,1,1)
## Q* = 0.46806, df = 6, p-value = 0.9982
##
## Model df: 4.   Total lags used: 10
```

```
# Evaluating predictive performance using the last 5 points
(pred5d=forecast(auto.arima(korea$dailyConfirmed[5:77]/korea$dailyTests[5:77], seasonal = FALSE),h=5))
```

```
##    Point Forecast        Lo 80      Hi 80       Lo 95      Hi 95
## 74    0.009062931 -0.009872609 0.02799847 -0.01989648 0.03802234
## 75    0.010244347 -0.010261310 0.03075000 -0.02111635 0.04160504
## 76    0.011238218 -0.010309599 0.03278604 -0.02171632 0.04419276
## 77    0.012074316 -0.010181577 0.03433021 -0.02196313 0.04611177
## 78    0.012777686 -0.009966000 0.03552137 -0.02200578 0.04756115
```

```
accuracy(korea$dailyConfirmed[78:82]/korea$dailyTests[78:82], pred5d$mean)
```

```
##                    ME        RMSE         MAE      MPE     MAPE        ACF1
## Test set 0.006851223 0.007203728 0.006851223 60.49638 60.49638 -0.03853435
##          Theil's U
## Test set  7.865711
```

ARIMA daily death

```
fit_death <- auto.arima(korea$dailyDeath, seasonal = FALSE)

preds_d <- forecast(fit_death, h=30)
plot(preds_d)

# Evaluating predictive performance using the last 5 points
(pred5d=forecast(auto.arima(korea$dailyDeath[1:77], seasonal = FALSE),h=5))
accuracy(korea$dailyDeath[78:82], pred5d$mean)
```

ARIMA daily death/ daily tests

```
fit_death_n <- auto.arima(korea$dailyDeath/korea$dailyTests, seasonal = FALSE)
fit_death_n

preds_dn <- forecast(fit_death_n, h=30)
plot(preds_dn)
```

```r
# Evaluating predictive performance using the last 5 points
(pred5dn=forecast(auto.arima(korea$dailyDeath[1:77]/korea$dailyTests[1:77], seasonal = FALSE),h=5))
accuracy(korea$dailyDeath/korea$dailyTests[78:82], pred5dn$mean)
```