US Covid-19 Cases Time Series Analysis

Stefan Maciolek

5/28/2020

#importing Covid Data from Johns Hopkins   
uscases <- read.csv(url("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse\_covid\_19\_data/csse\_covid\_19\_time\_series/time\_series\_covid19\_confirmed\_US.csv"))  
usdeaths <- read.csv(url("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse\_covid\_19\_data/csse\_covid\_19\_time\_series/time\_series\_covid19\_deaths\_US.csv"))  
uscases <- uscases[,-c(1:5,8:11)]  
usdeaths <- usdeaths[,-c(1:5,8:12)]  
n <- ncol(uscases)-2  
date <- 1:n  
date <- as.Date(date,origin = "2020-01-21")  
format(date,format = "%b %d %y")

## [1] "Jan 22 20" "Jan 23 20" "Jan 24 20" "Jan 25 20" "Jan 26 20" "Jan 27 20"  
## [7] "Jan 28 20" "Jan 29 20" "Jan 30 20" "Jan 31 20" "Feb 01 20" "Feb 02 20"  
## [13] "Feb 03 20" "Feb 04 20" "Feb 05 20" "Feb 06 20" "Feb 07 20" "Feb 08 20"  
## [19] "Feb 09 20" "Feb 10 20" "Feb 11 20" "Feb 12 20" "Feb 13 20" "Feb 14 20"  
## [25] "Feb 15 20" "Feb 16 20" "Feb 17 20" "Feb 18 20" "Feb 19 20" "Feb 20 20"  
## [31] "Feb 21 20" "Feb 22 20" "Feb 23 20" "Feb 24 20" "Feb 25 20" "Feb 26 20"  
## [37] "Feb 27 20" "Feb 28 20" "Feb 29 20" "Mar 01 20" "Mar 02 20" "Mar 03 20"  
## [43] "Mar 04 20" "Mar 05 20" "Mar 06 20" "Mar 07 20" "Mar 08 20" "Mar 09 20"  
## [49] "Mar 10 20" "Mar 11 20" "Mar 12 20" "Mar 13 20" "Mar 14 20" "Mar 15 20"  
## [55] "Mar 16 20" "Mar 17 20" "Mar 18 20" "Mar 19 20" "Mar 20 20" "Mar 21 20"  
## [61] "Mar 22 20" "Mar 23 20" "Mar 24 20" "Mar 25 20" "Mar 26 20" "Mar 27 20"  
## [67] "Mar 28 20" "Mar 29 20" "Mar 30 20" "Mar 31 20" "Apr 01 20" "Apr 02 20"  
## [73] "Apr 03 20" "Apr 04 20" "Apr 05 20" "Apr 06 20" "Apr 07 20" "Apr 08 20"  
## [79] "Apr 09 20" "Apr 10 20" "Apr 11 20" "Apr 12 20" "Apr 13 20" "Apr 14 20"  
## [85] "Apr 15 20" "Apr 16 20" "Apr 17 20" "Apr 18 20" "Apr 19 20" "Apr 20 20"  
## [91] "Apr 21 20" "Apr 22 20" "Apr 23 20" "Apr 24 20" "Apr 25 20" "Apr 26 20"  
## [97] "Apr 27 20" "Apr 28 20" "Apr 29 20" "Apr 30 20" "May 01 20" "May 02 20"  
## [103] "May 03 20" "May 04 20" "May 05 20" "May 06 20" "May 07 20" "May 08 20"  
## [109] "May 09 20" "May 10 20" "May 11 20" "May 12 20" "May 13 20" "May 14 20"  
## [115] "May 15 20" "May 16 20" "May 17 20" "May 18 20" "May 19 20" "May 20 20"  
## [121] "May 21 20" "May 22 20" "May 23 20" "May 24 20" "May 25 20" "May 26 20"  
## [127] "May 27 20"

The data were imported from Johns Hopkins, cleaned up, and a date variable was created for plotting purposes. For this analysis the case data wasn’t used, just the case numbers

#finding total cases and turning them into a time series object  
library(TSA)

##   
## Attaching package: 'TSA'

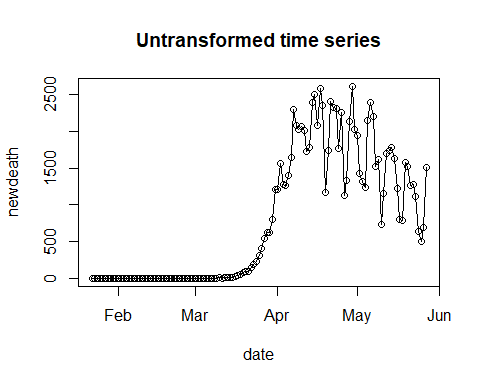
## The following objects are masked from 'package:stats':  
##   
## acf, arima

## The following object is masked from 'package:utils':  
##   
## tar

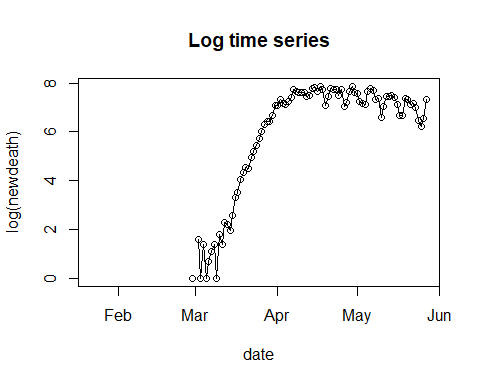
totdeath <- colSums(usdeaths[,3:(n+2)])  
newdeath <- rep(0,n)  
newdeath[1] <- totdeath[1]  
newdeath[2:n] <- diff(totdeath)  
newdeath <- ts(data=newdeath,start=c(2020,01,22),frequency = 365)

A the data were turned into a time series object

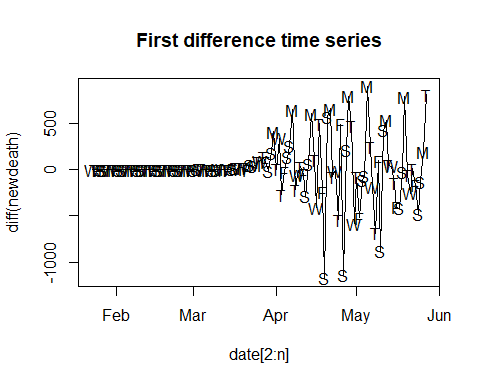
#trying various transformations  
wkday <- c("W","T","F","S","S","M","T")  
plot(date,newdeath,type="o",main="Untransformed time series")



plot(date,log(newdeath),type="o",main="Log time series")

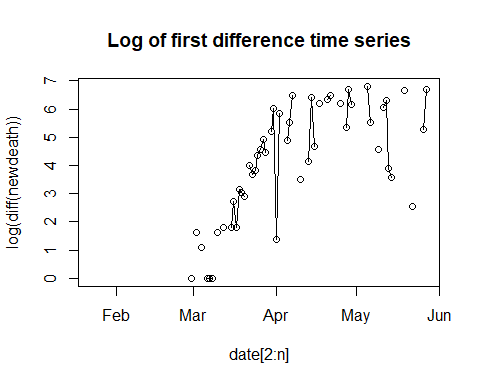


plot(date[2:n],diff(newdeath),type="o",main="First difference time series",pch=wkday)

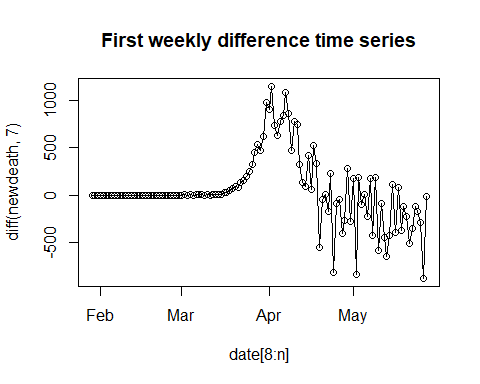


plot(date[2:n],log(diff(newdeath)),type="o",main="Log of first difference time series")

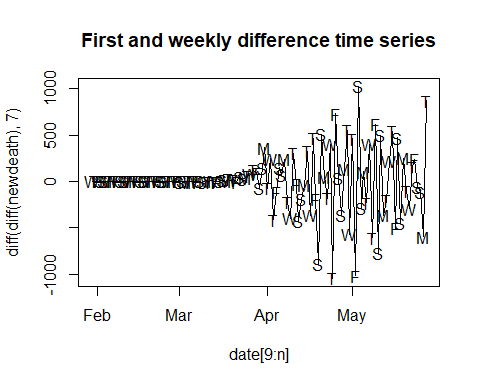
## Warning in log(diff(newdeath)): NaNs produced



plot(date[8:n],diff(newdeath,7),type="o",main="First weekly difference time series")

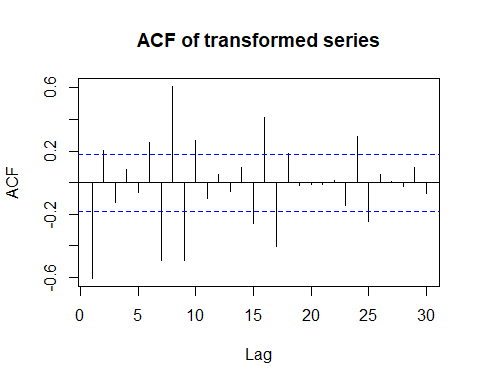


plot(date[9:n],diff(diff(newdeath),7),type="o",main="First and weekly difference time series",pch=wkday)

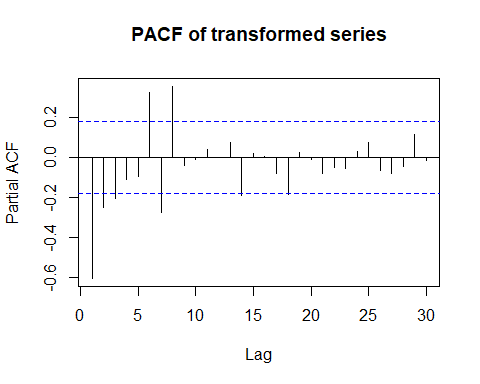


Several transformations and combinations of transformations were tested to see if they made the data random. In the end, a first and first seasonal difference with a weekly period were chosen.

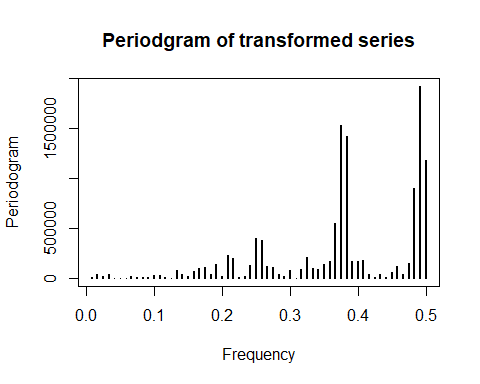
#starting with an ARIMA model with both a first and first weekly difference  
#preliminary analysis  
transdeath <- diff(diff(newdeath),7)  
acf(as.vector(transdeath),main="ACF of transformed series",lag.max=30)



pacf(as.vector(transdeath),main="PACF of transformed series",lag.max=30)



periodogram(transdeath,main="Periodgram of transformed series")



The ACF and PACF of the series suggest a starting model with at least 1 AR and 1 seasonal AR component.

#model fitting:  
#looking at the periodograms, it appears a ARIMA(1,1,0)x(1,1,0)\_7 is justified to start  
tsmodel <- function(ar,ma,ars,mas){  
 mod <- arima(newdeath,order=c(ar,1,ma),seasonal=list(order=c(ars,1,mas),period=7))  
 return(mod)  
}  
tsmodel(1,0,1,0)

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 sar1  
## -0.5466 -0.2685  
## s.e. 0.0906 0.1036  
##   
## sigma^2 estimated as 57435: log likelihood = -821.32, aic = 1646.64

tsmodel(2,0,1,0)

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 ar2 sar1  
## -0.6753 -0.2538 -0.2696  
## s.e. 0.1081 0.0912 0.1144  
##   
## sigma^2 estimated as 53876: log likelihood = -817.58, aic = 1641.16

tsmodel(3,0,1,0)

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 ar2 ar3 sar1  
## -0.7311 -0.3754 -0.1811 -0.2368  
## s.e. 0.1097 0.1113 0.0942 0.1126  
##   
## sigma^2 estimated as 52266: log likelihood = -815.77, aic = 1639.53

tsmodel(3,0,2,0) #not significant

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 ar2 ar3 sar1 sar2  
## -0.7192 -0.3509 -0.2095 -0.2779 -0.1595  
## s.e. 0.1151 0.1128 0.0945 0.1214 0.1000  
##   
## sigma^2 estimated as 51011: log likelihood = -814.52, aic = 1639.04

tsmodel(3,1,1,0) #not significant

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 ar2 ar3 ma1 sar1  
## -0.4764 -0.2002 -0.1231 -0.2702 -0.2286  
## s.e. 0.2999 0.2281 0.1276 0.2892 0.1140  
##   
## sigma^2 estimated as 51999: log likelihood = -815.46, aic = 1640.92

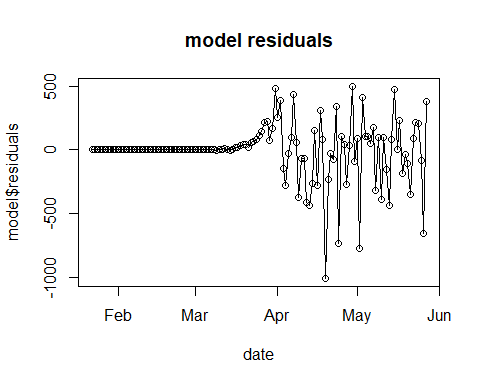
tsmodel(3,0,1,1) #not stationary

##   
## Call:  
## arima(x = newdeath, order = c(ar, 1, ma), seasonal = list(order = c(ars, 1,   
## mas), period = 7))  
##   
## Coefficients:  
## ar1 ar2 ar3 sar1 sma1  
## -0.7197 -0.3460 -0.1837 0.3006 -0.5915  
## s.e. 0.1150 0.1142 0.0934 0.3352 0.2677  
##   
## sigma^2 estimated as 50672: log likelihood = -814.19, aic = 1638.37

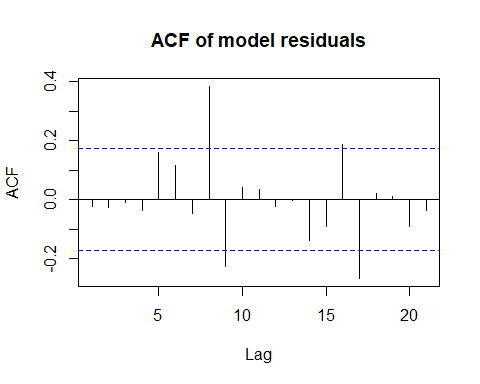
#the final model is an ARIMA(3,1,0)X(1,1,0)\_7

The order of the model was raised one component at a time until adding further components were no longer significant. The final model was an ARIMA(3,1,0)X(1,1,0)\_7

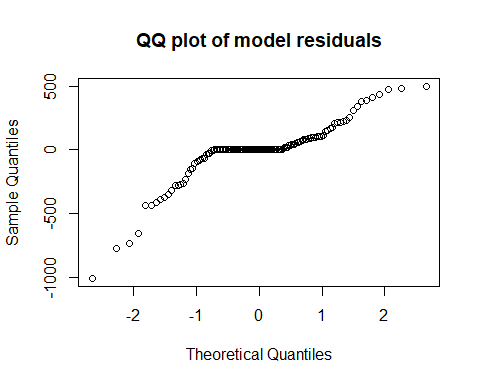
#diagnostics:  
model<-tsmodel(3,0,1,0)  
plot(date,model$residuals,main="model residuals",type="o")



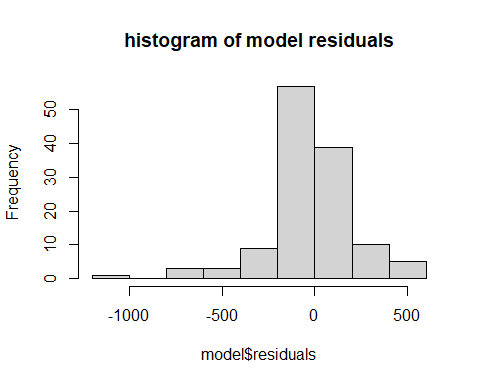
acf(as.vector(model$residuals),main="ACF of model residuals")



qqnorm(model$residuals,main="QQ plot of model residuals")



hist(model$residuals,main="histogram of model residuals")

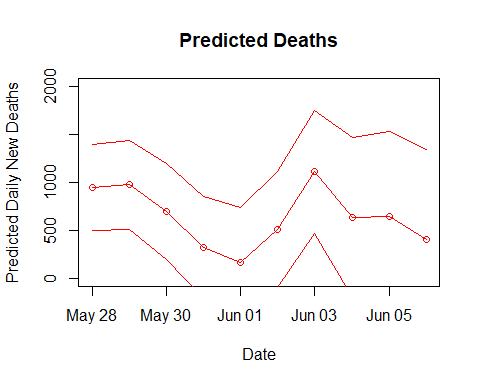


The residuals were analyzed and diagnostics produced. The residuals are more or less normal, and although they have a high autocorrelation at lag 8, it doesn’t appear large enough to justify a more complex model

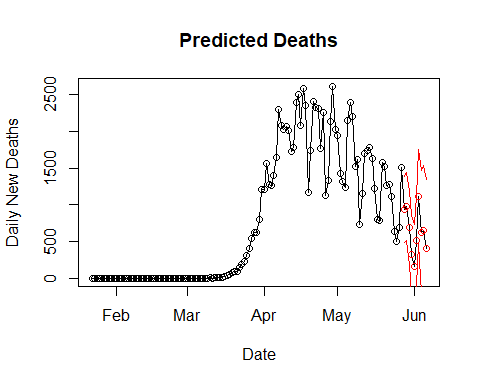
#predictions  
predictions <- predict(model,n.ahead=10)  
dateahead<- seq(from=(n+1),to=(n+10))  
dateahead <- as.Date(dateahead,origin = "2020-01-21")  
format(dateahead,format = "%b %d %y")

## [1] "May 28 20" "May 29 20" "May 30 20" "May 31 20" "Jun 01 20" "Jun 02 20"  
## [7] "Jun 03 20" "Jun 04 20" "Jun 05 20" "Jun 06 20"

plot(dateahead,predictions$pred,main="Predicted Deaths",type="o",ylim=c(0,2000),ylab="Predicted Daily New Deaths",xlab="Date",col="red")  
lines(dateahead,(predictions$pred - 1.96\*predictions$se),type="l",col="red")  
lines(dateahead,(predictions$pred + 1.96\*predictions$se),type="l",col="red")



datenew <- c(date,dateahead)  
total <- c(newdeath,predictions$pred)  
plot(datenew,total,col=c(rep("black",n),rep("red",10)),main="Predicted Deaths",type="o",ylab="Daily New Deaths",xlab="Date")  
lines(dateahead,(predictions$pred - 1.96\*predictions$se),type="l",col="red")  
lines(dateahead,(predictions$pred + 1.96\*predictions$se),type="l",col="red")



The model was used to create predictions for 10 days out.