Project 3

Cameron Farrugia

5/6/2020

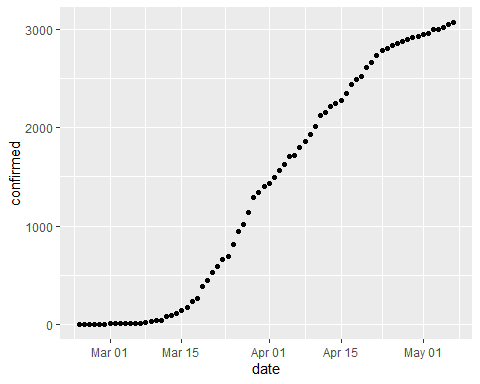
## Project 3

I left off project 2 with some simple analysis of the data. Looking at the data again I think it would be most benefical to try to implement time series analysis to best predict the trend of covid in Italy.

covid=covid19("ITA", level = 2)  
str(covid)

## Classes 'grouped\_df', 'tbl\_df', 'tbl' and 'data.frame': 1554 obs. of 42 variables:  
## $ id : chr "ITA, Abruzzo" "ITA, Abruzzo" "ITA, Abruzzo" "ITA, Abruzzo" ...  
## $ date : Date, format: "2020-02-24" "2020-02-25" ...  
## $ deaths : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ confirmed : num 0 0 0 1 1 2 5 5 6 7 ...  
## $ tests : num 5 5 13 33 33 43 52 52 52 85 ...  
## $ recovered : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ hosp : num 0 0 0 1 1 2 3 3 5 7 ...  
## $ icu : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ vent : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ school\_closing : num 3 3 3 3 3 3 3 3 3 3 ...  
## $ workplace\_closing : num 3 3 3 3 3 3 3 3 3 3 ...  
## $ cancel\_events : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ gatherings\_restrictions : num 4 4 4 4 4 4 4 4 4 4 ...  
## $ transport\_closing : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ stay\_home\_restrictions : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ internal\_movement\_restrictions : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ international\_movement\_restrictions: num 3 3 3 3 3 3 3 3 3 3 ...  
## $ information\_campaigns : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ testing\_policy : num 1 1 2 2 2 2 2 2 2 2 ...  
## $ contact\_tracing : num 2 2 2 2 2 2 2 2 2 2 ...  
## $ stringency\_index : num 64.4 64.4 64.4 64.4 64.4 ...  
## $ mkt\_close : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ mkt\_volume : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ country : chr "Italy" "Italy" "Italy" "Italy" ...  
## $ state : chr "Abruzzo" "Abruzzo" "Abruzzo" "Abruzzo" ...  
## $ city : chr NA NA NA NA ...  
## $ lat : num 42.4 42.4 42.4 42.4 42.4 ...  
## $ lng : num 13.4 13.4 13.4 13.4 13.4 ...  
## $ pop : int 1315196 1315196 1315196 1315196 1315196 1315196 1315196 1315196 1315196 1315196 ...  
## $ pop\_female : logi NA NA NA NA NA NA ...  
## $ pop\_14 : num 0.126 0.126 0.126 0.126 0.126 0.126 0.126 0.126 0.126 0.126 ...  
## $ pop\_15\_64 : num 0.639 0.639 0.639 0.639 0.639 0.639 0.639 0.639 0.639 0.639 ...  
## $ pop\_65 : num 0.236 0.236 0.236 0.236 0.236 0.236 0.236 0.236 0.236 0.236 ...  
## $ pop\_age : num 45.9 45.9 45.9 45.9 45.9 45.9 45.9 45.9 45.9 45.9 ...  
## $ pop\_density : num 121 121 121 121 121 ...  
## $ pop\_death\_rate : num 0.0112 0.0112 0.0112 0.0112 0.0112 0.0112 0.0112 0.0112 0.0112 0.0112 ...  
## $ hosp\_beds : logi NA NA NA NA NA NA ...  
## $ smoking\_male : logi NA NA NA NA NA NA ...  
## $ smoking\_female : logi NA NA NA NA NA NA ...  
## $ gdp : logi NA NA NA NA NA NA ...  
## $ health\_exp : logi NA NA NA NA NA NA ...  
## $ health\_exp\_oop : logi NA NA NA NA NA NA ...  
## - attr(\*, "groups")=Classes 'tbl\_df', 'tbl' and 'data.frame': 21 obs. of 2 variables:  
## ..$ id : chr "ITA, Abruzzo" "ITA, Basilicata" "ITA, Calabria" "ITA, Campania" ...  
## ..$ .rows:List of 21  
## .. ..$ : int 1 2 3 4 5 6 7 8 9 10 ...  
## .. ..$ : int 75 76 77 78 79 80 81 82 83 84 ...  
## .. ..$ : int 149 150 151 152 153 154 155 156 157 158 ...  
## .. ..$ : int 223 224 225 226 227 228 229 230 231 232 ...  
## .. ..$ : int 297 298 299 300 301 302 303 304 305 306 ...  
## .. ..$ : int 371 372 373 374 375 376 377 378 379 380 ...  
## .. ..$ : int 445 446 447 448 449 450 451 452 453 454 ...  
## .. ..$ : int 519 520 521 522 523 524 525 526 527 528 ...  
## .. ..$ : int 593 594 595 596 597 598 599 600 601 602 ...  
## .. ..$ : int 667 668 669 670 671 672 673 674 675 676 ...  
## .. ..$ : int 741 742 743 744 745 746 747 748 749 750 ...  
## .. ..$ : int 815 816 817 818 819 820 821 822 823 824 ...  
## .. ..$ : int 889 890 891 892 893 894 895 896 897 898 ...  
## .. ..$ : int 963 964 965 966 967 968 969 970 971 972 ...  
## .. ..$ : int 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 ...  
## .. ..$ : int 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 ...  
## .. ..$ : int 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194 ...  
## .. ..$ : int 1259 1260 1261 1262 1263 1264 1265 1266 1267 1268 ...  
## .. ..$ : int 1333 1334 1335 1336 1337 1338 1339 1340 1341 1342 ...  
## .. ..$ : int 1407 1408 1409 1410 1411 1412 1413 1414 1415 1416 ...  
## .. ..$ : int 1481 1482 1483 1484 1485 1486 1487 1488 1489 1490 ...  
## ..- attr(\*, ".drop")= logi TRUE

covid1<- covid %>% filter(state=="Abruzzo")  
ggplot(covid1, aes(x=date, y=confirmed))+geom\_point()

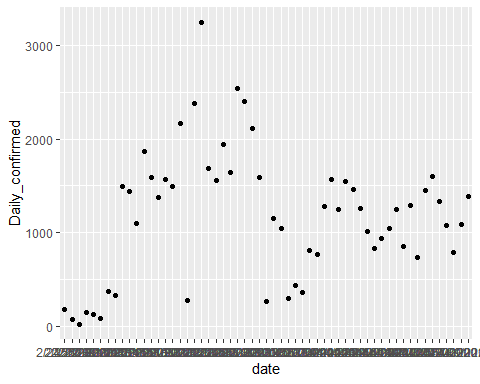


This graph can be misleading because this is Cumulative data, so I decided to break it down into daily confirmed cases. I also decided to change my focus to the state of Lombardia because it had the largest amount of cases.

It was easiest to create a new column in Excel to make a daily confirmed cases column.

covid<-read.csv("covidOG.csv")  
covid1<- covid %>% filter(state=="Lombardia")  
ggplot(covid1, aes(x=date, y=Daily\_confirmed))+geom\_point()+geom\_smooth()

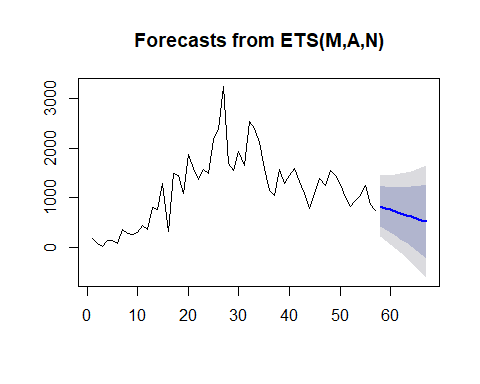
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



It looks like there is a spike in March and it drops off more in April and is somewhat steady. Let’s see if I can produce a time series model that can predict future daily confirmed correctly.

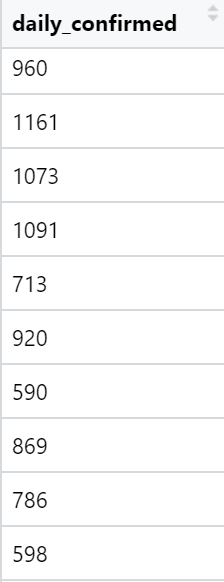
daily\_confirmed\_ts<-ts(covid1$Daily\_confirmed, frequency = 7)

fit <- ets(covid1$Daily\_confirmed)  
fc <- forecast(fit)  
plot(fc)



summary(fc)

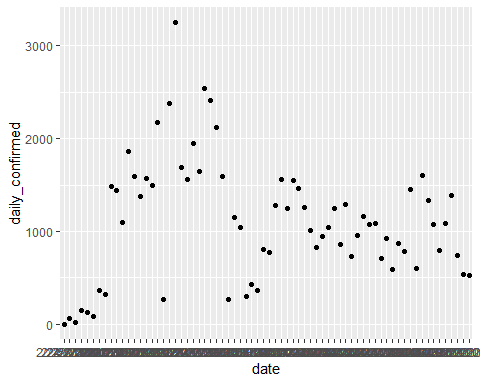
##   
## Forecast method: ETS(M,A,N)  
##   
## Model Information:  
## ETS(M,A,N)   
##   
## Call:  
## ets(y = covid1$Daily\_confirmed)   
##   
## Smoothing parameters:  
## alpha = 0.3902   
## beta = 0.0372   
##   
## Initial states:  
## l = 71.1455   
## b = 57.9801   
##   
## sigma: 0.3795  
##   
## AIC AICc BIC   
## 908.6587 909.8352 918.8740   
##   
## Error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -43.96754 403.531 306.0662 -34.21425 52.7828 1.000275  
## ACF1  
## Training set 0.1617949  
##   
## Forecasts:  
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## 58 827.3849 424.94637 1229.823 211.90829 1442.861  
## 59 792.2554 365.24823 1219.263 139.20429 1445.307  
## 60 757.1259 301.58823 1212.664 60.44114 1453.811  
## 61 721.9964 234.25590 1209.737 -23.93835 1467.931  
## 62 686.8670 163.52846 1210.205 -113.51021 1487.244  
## 63 651.7375 89.65154 1213.823 -207.89878 1511.374  
## 64 616.6080 12.82977 1220.386 -306.79113 1540.007  
## 65 581.4785 -66.77603 1229.733 -409.94127 1572.898  
## 66 546.3490 -149.04773 1241.746 -517.16855 1609.867  
## 67 511.2196 -233.90693 1256.346 -628.35307 1650.792

 These are the actual numbers from the new data. These are what I compared to the point forecast directly above.

The forecast predicts the next 10 days of the daily confirmed cases. Luckily this data set is updated daily and I am able to compare results by looking at the new data. I did this by getting the new data set and modifying the Excel file to have the daily confirmed column as I did with the original.

covid2<-read.csv("covidNew.csv")  
covid3<- covid2 %>% filter(state=="Lombardia")  
ggplot(covid3, aes(x=date, y=daily\_confirmed))+geom\_point()+geom\_smooth()

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'



newCovid<-ggplot(covid3, aes(x=date, y=daily\_confirmed))+geom\_point()+geom\_smooth()  
daily\_confirmed\_ts\_new<-ts(covid3$daily\_confirmed, frequency = 7)

Since I am not entirely sure how to compare the forecasted and the actual data, I manually looked at the predicted and the real. Although the numbers were a bit off the numbers were within the confidence intervals which is good to see.

My hope was to try to find out MSE which is very high for this. So either I wasn’t quite able to get it correctly or it is just very unpredictable. This MSE is very close to the forecasted model as well though, so that is good to see that they are close.

daily\_confirmed\_ts %>% tsCV(forecastfunction=rwf, drift=TRUE, h=1) -> e  
e^2 %>% mean(na.rm=TRUE) %>% sqrt()

## [1] 444.8277

sqrt(mean(residuals(rwf(daily\_confirmed\_ts, drift=TRUE))^2, na.rm=TRUE))

## [1] 430.535

# Figuring out test error

My original thought was to just compare the new and old data, but it occured to me that I may know how to find test error for linear models, but I don’t know how for time series. If I found a way to do it I would have used the old covid dataset as the training set and used the new one from the day after the old one ended to test it. The manual assessment was able to tell me that it was within the ranges predicted, but not quite accurate to the actual values predicted.

# Subsetting time series

My thought on how to subset the data is in the code below.

traints<-subset(daily\_confirmed\_ts, start=length(daily\_confirmed\_ts))  
testts<-subset(daily\_confirmed\_ts\_new, start=length(daily\_confirmed\_ts\_new)-length(daily\_confirmed\_ts))

# Conclusion

I decided to use a time series approach to this problem because we are trying to predict future events based on past information. I did not want to use a linear model for this problem for many reasons. One reason was because trying a linear model produced a not accurate result which makes sense because pandemics rise and fall in more of a curved pattern. That is why I decided time series was the best approach. Time series forecasting is a good apporach because as was shown in a graph eariler it shows confidence intervals. The reason this is important is because it is clear to see that the trend of the virus is very hard to predict and although the time series model isn’t perfect at predicting the actual amounts the actual was still within the confidence intervals. It is more realistic to know the upper and lower bounds of the possible outcomes than it is to know the exact statistics with something as unpredictable as this. Since the dataset was designed the way it was where their wasn’t many good predictor canidates, I think time series was the best option. My analysis explored just the trend of daily cases and tried to predict roughly the amount of daily cases going forward. I think that considering the dataset and the approach I took that the model is decent at showing at least the trend of potential future results. I really liked the forecasting graph that was produced showing the predicted line and confidence intervals, time series was brand new for me and I had to teach myself as I went. I know I didn’t quite get everything I was looking for done, but it is a tool I hope to try to perfect in the future.