Assignment 7

Dipesh Poudel

12/28/2021

**Instructions:**

Use the attached Nepal COVID-19 data extracted from Wikipedia to fit the following models with daily deaths as dependent variable and time as independent variable.

First plot the daily deaths by time and distribute the three outliers (added deaths around timeline of 400) before fitting the following models in the outlier adjusted data on training and testing datasets:

### Loading the excel data

library(readxl)  
covid\_tbl<-read\_excel('covid\_tbl\_final.xlsx')

str(covid\_tbl)

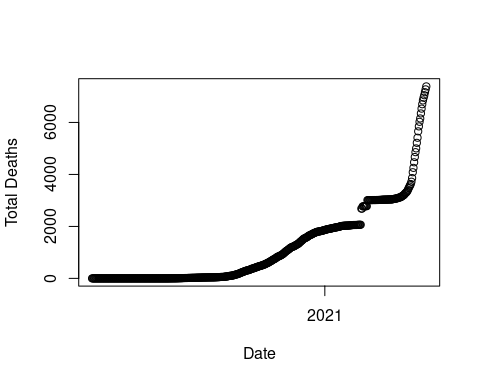
## tibble [495 × 14] (S3: tbl\_df/tbl/data.frame)  
## $ SN : num [1:495] 1 2 3 4 5 6 7 8 9 10 ...  
## $ Date : POSIXct[1:495], format: "2020-01-23" "2020-01-24" ...  
## $ Confirmed\_cases\_total : num [1:495] 1 1 1 1 1 1 1 1 1 1 ...  
## $ Confirmed\_cases\_new : num [1:495] 1 0 0 0 0 0 0 0 0 0 ...  
## $ Confirmed \_cases\_active: num [1:495] 1 1 1 1 1 1 0 0 0 0 ...  
## $ Recoveries\_total : num [1:495] 0 0 0 0 0 0 1 1 1 1 ...  
## $ Recoveries\_daily : num [1:495] 0 0 0 0 0 0 1 0 0 0 ...  
## $ Deaths\_total : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Deaths\_daily : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...  
## $ RT-PCR\_tests\_total : num [1:495] NA NA NA NA NA 3 4 5 5 NA ...  
## $ RT-PCR\_tests\_daily : num [1:495] NA NA NA NA NA NA 1 1 0 NA ...  
## $ Test\_positivity\_rate : num [1:495] NA NA NA NA NA ...  
## $ Recovery\_rate : num [1:495] 0 0 0 0 0 0 100 100 100 100 ...  
## $ Case\_fatality\_rate : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...

covid\_tbl$Date<-as.Date(as.POSIXct(covid\_tbl$Date))

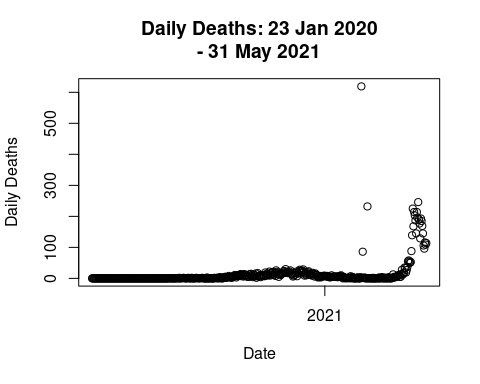
str(covid\_tbl)

## tibble [495 × 14] (S3: tbl\_df/tbl/data.frame)  
## $ SN : num [1:495] 1 2 3 4 5 6 7 8 9 10 ...  
## $ Date : Date[1:495], format: "2020-01-23" "2020-01-24" ...  
## $ Confirmed\_cases\_total : num [1:495] 1 1 1 1 1 1 1 1 1 1 ...  
## $ Confirmed\_cases\_new : num [1:495] 1 0 0 0 0 0 0 0 0 0 ...  
## $ Confirmed \_cases\_active: num [1:495] 1 1 1 1 1 1 0 0 0 0 ...  
## $ Recoveries\_total : num [1:495] 0 0 0 0 0 0 1 1 1 1 ...  
## $ Recoveries\_daily : num [1:495] 0 0 0 0 0 0 1 0 0 0 ...  
## $ Deaths\_total : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Deaths\_daily : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...  
## $ RT-PCR\_tests\_total : num [1:495] NA NA NA NA NA 3 4 5 5 NA ...  
## $ RT-PCR\_tests\_daily : num [1:495] NA NA NA NA NA NA 1 1 0 NA ...  
## $ Test\_positivity\_rate : num [1:495] NA NA NA NA NA ...  
## $ Recovery\_rate : num [1:495] 0 0 0 0 0 0 100 100 100 100 ...  
## $ Case\_fatality\_rate : num [1:495] 0 0 0 0 0 0 0 0 0 0 ...

plot(covid\_tbl$Date,covid\_tbl$Deaths\_total,xlab = "Date",ylab = "Total Deaths")



plot(covid\_tbl$Date,  
covid\_tbl$Deaths\_daily,  
main = "Daily Deaths: 23 Jan 2020  
- 31 May 2021",  
xlab = "Date",  
ylab = "Daily Deaths")



summary(covid\_tbl$Deaths\_daily)

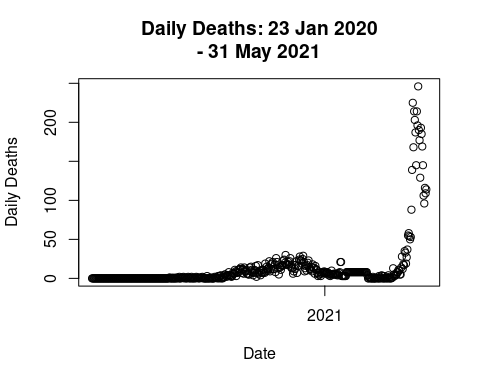
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00 0.00 2.00 14.92 11.00 619.00

library(dplyr)  
filter(covid\_tbl,Deaths\_daily>=50&Date<=as.Date("2021-03-05"))

## # A tibble: 3 × 14  
## SN Date Confirmed\_cases\_total Confirmed\_cases\_new `Confirmed \_cases\_…  
## <dbl> <date> <dbl> <dbl> <dbl>  
## 1 399 2021-02-24 273760 94 937  
## 2 401 2021-02-26 273984 112 936  
## 3 408 2021-03-05 274608 120 832  
## # … with 9 more variables: Recoveries\_total <dbl>, Recoveries\_daily <dbl>,  
## # Deaths\_total <dbl>, Deaths\_daily <dbl>, RT-PCR\_tests\_total <dbl>,  
## # RT-PCR\_tests\_daily <dbl>, Test\_positivity\_rate <dbl>, Recovery\_rate <dbl>,  
## # Case\_fatality\_rate <dbl>

wsn<-c(399,401,408)  
for(i in 1:length(wsn)){  
  
temp\_sn = wsn[i]   
# Get the Value to be adjusted  
curr\_val<-covid\_tbl[covid\_tbl$SN==temp\_sn,"Deaths\_daily"]  
# Calculate the average daily deaths for last 30 days  
avg\_daily\_deaths<-ceiling(mean(covid\_tbl[covid\_tbl$SN %in% c((temp\_sn-1):(temp\_sn-1-30)),]$Deaths\_daily))  
  
# Change the Value for given SN  
covid\_tbl[covid\_tbl$SN==temp\_sn,"Deaths\_daily"]=avg\_daily\_deaths  
# Change values for last 30 days  
covid\_tbl[covid\_tbl$SN %in% c((temp\_sn-1):(temp\_sn-1-30)),]$Deaths\_daily=as.integer( round(curr\_val/30))  
}

plot(covid\_tbl$Date,  
covid\_tbl$Deaths\_daily,  
main = "Daily Deaths: 23 Jan 2020  
- 31 May 2021",  
xlab = "Date",  
ylab = "Daily Deaths")



### Splitting the data into training and testing set

set.seed(1234)  
ind<-sample(2,nrow(covid\_tbl),replace=T,prob = c(0.7,0.3))  
train\_data<-covid\_tbl[ind==1,]  
test\_data<-covid\_tbl[ind==2,]

## 1. Linear regression model

library(caret)

lm1<-train(Deaths\_daily~SN,data=train\_data,method="lm")  
predict1<-predict(lm1,newdata = test\_data)

summary(lm1)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -35.658 -11.892 -2.591 4.622 205.538   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -13.50449 3.28569 -4.110 4.95e-05 \*\*\*  
## SN 0.11173 0.01169 9.561 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 31.51 on 345 degrees of freedom  
## Multiple R-squared: 0.2094, Adjusted R-squared: 0.2072   
## F-statistic: 91.4 on 1 and 345 DF, p-value: < 2.2e-16

predict\_eval<-function(predicted\_values){  
 return(data.frame(  
 R2=R2(predicted\_values,test\_data$Deaths\_daily),  
RMSE = RMSE(predicted\_values,test\_data$Deaths\_daily),  
MAE = MAE(predicted\_values,test\_data$Deaths\_daily)  
 ))  
}

predict\_eval(predict1)

## R2 RMSE MAE  
## 1 0.1887896 32.1613 17.61361

In this model, the value of R-square decreased in the test data i. e in the training data it was 0.20 and in test data it is 0.188

## 2. Quadratic linear regression model

lm2<-train(Deaths\_daily~poly(SN,2),data=train\_data,method="lm")  
summary(lm2)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -48.544 -10.617 1.553 6.616 181.775   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.427 1.558 8.618 2.52e-16 \*\*\*  
## `poly(SN, 2)1` 301.229 29.022 10.379 < 2e-16 \*\*\*  
## `poly(SN, 2)2` 229.647 29.022 7.913 3.48e-14 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 29.02 on 344 degrees of freedom  
## Multiple R-squared: 0.3312, Adjusted R-squared: 0.3273   
## F-statistic: 85.17 on 2 and 344 DF, p-value: < 2.2e-16

predict2<-predict(lm2,newdata = test\_data)  
predict\_eval(predict2)

## R2 RMSE MAE  
## 1 0.3143297 29.52953 18.11123

In the quadratic linear regression model the value of R-squared has increased compared to the simple linear regression model. In this case the R-2 value has decreased in the test data. ## 3. Cubic linear regression model

lm3<-train(Deaths\_daily~poly(SN,3),data = train\_data,method="lm")  
summary(lm3)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -56.401 -9.822 -2.567 10.088 157.909   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.427 1.404 9.566 <2e-16 \*\*\*  
## `poly(SN, 3)1` 301.229 26.145 11.522 <2e-16 \*\*\*  
## `poly(SN, 3)2` 229.647 26.145 8.784 <2e-16 \*\*\*  
## `poly(SN, 3)3` 235.151 26.145 8.994 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 26.14 on 343 degrees of freedom  
## Multiple R-squared: 0.4588, Adjusted R-squared: 0.4541   
## F-statistic: 96.93 on 3 and 343 DF, p-value: < 2.2e-16

predict3<-predict(lm3,newdata = test\_data)  
predict\_eval(predict3)

## R2 RMSE MAE  
## 1 0.4823308 25.6787 16.66555

In this model the R-squared in the both train and test data has increased compared to previous two models. Also, the R2 value for test has also increased compared to the train.

## 4. Double quadratic linear regression model

lm4<-train(Deaths\_daily~poly(SN,4),data = train\_data,method="lm")  
summary(lm4)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -53.511 -9.839 1.374 8.894 133.202   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.427 1.166 11.52 <2e-16 \*\*\*  
## `poly(SN, 4)1` 301.229 21.720 13.87 <2e-16 \*\*\*  
## `poly(SN, 4)2` 229.647 21.720 10.57 <2e-16 \*\*\*  
## `poly(SN, 4)3` 235.151 21.720 10.83 <2e-16 \*\*\*  
## `poly(SN, 4)4` 270.390 21.720 12.45 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 21.72 on 342 degrees of freedom  
## Multiple R-squared: 0.6276, Adjusted R-squared: 0.6232   
## F-statistic: 144.1 on 4 and 342 DF, p-value: < 2.2e-16

predict4<-predict(lm4,newdata = test\_data)  
predict\_eval(predict4)

## R2 RMSE MAE  
## 1 0.6857402 19.98498 14.03474

In this model the R-squared in the both train and test data has increased compared to previous two models. Also, the R2 value for test has also increased compared to the train.

## 5. Fifth order polynomial regression model

lm5<-train(Deaths\_daily~poly(SN,5),data = train\_data,method="lm")  
summary(lm5)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -70.136 -4.962 -0.243 4.227 127.848   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.427 1.003 13.39 <2e-16 \*\*\*  
## `poly(SN, 5)1` 301.229 18.681 16.12 <2e-16 \*\*\*  
## `poly(SN, 5)2` 229.647 18.681 12.29 <2e-16 \*\*\*  
## `poly(SN, 5)3` 235.151 18.681 12.59 <2e-16 \*\*\*  
## `poly(SN, 5)4` 270.390 18.681 14.47 <2e-16 \*\*\*  
## `poly(SN, 5)5` 205.774 18.681 11.02 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.68 on 341 degrees of freedom  
## Multiple R-squared: 0.7253, Adjusted R-squared: 0.7213   
## F-statistic: 180.1 on 5 and 341 DF, p-value: < 2.2e-16

predict5<-predict(lm5,newdata = test\_data)  
predict\_eval(predict5)

## R2 RMSE MAE  
## 1 0.8005885 15.90596 8.879888

This model has the high value of R2 0.72 in the training and 0.80 in the test. Since the value of R2 has increased in the test we can say that our model is a good model.  
## 6. KNN regression model

library(caret)  
knnmodel<-train(Deaths\_daily~SN,data = train\_data,method="knn")  
summary(knnmodel)

## Length Class Mode   
## learn 2 -none- list   
## k 1 -none- numeric   
## theDots 0 -none- list   
## xNames 1 -none- character  
## problemType 1 -none- character  
## tuneValue 1 data.frame list   
## obsLevels 1 -none- logical   
## param 0 -none- list

predict6<-predict(knnmodel,newdata = test\_data)  
predict\_eval(predict6)

## R2 RMSE MAE  
## 1 0.9777022 5.806763 2.827703

Since KNN is a lazy algorithm there are no interpret able summary in the KNN model. In test data we got the R2 of 0.97 which is highest till now. ## 7. ANN-MLP regression model with 2 hidden layers with 3 and neurons

library(neuralnet)  
nn<-neuralnet(Deaths\_daily~SN,data = train\_data,hidden = c(3,2),linear.output = F)  
plot(nn,main="The Architecture of the Neural Network")  
summary(nn)

## Length Class Mode   
## call 5 -none- call   
## response 347 -none- numeric   
## covariate 347 -none- numeric   
## model.list 2 -none- list   
## err.fct 1 -none- function  
## act.fct 1 -none- function  
## linear.output 1 -none- logical   
## data 14 data.frame list   
## exclude 0 -none- NULL   
## net.result 1 -none- list   
## weights 1 -none- list   
## generalized.weights 1 -none- list   
## startweights 1 -none- list   
## result.matrix 20 -none- numeric

predict7<-predict(nn,newdata = test\_data)  
predict\_eval(predict7)

## R2 RMSE MAE  
## 1 0.03179269 37.80605 13.32432

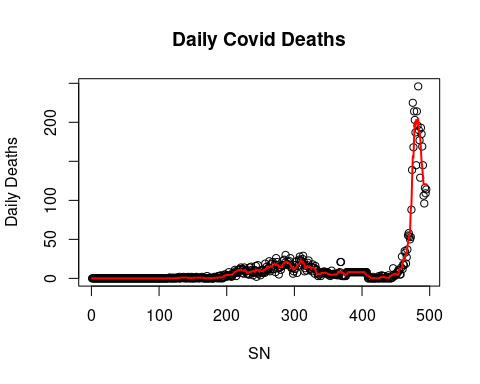
The neural network has the lowest value of R2.

## 8. Select the best model with lowest RMSE on the test data

The best model is the one with highest value of R2 and smallest value of RMSE(Error). In our case KNN model gave the highest value of R2. Therefore we can say that KNN is the best model in this case.

## 9. Write a summary and recommendation for Ministry of Health, Nepal

#Plot with linear model  
plot(covid\_tbl$SN, covid\_tbl$Deaths\_daily,  
main = "Daily Covid Deaths",  
xlab = "SN",  
ylab = "Daily Deaths")  
lines(predict(knnmodel,newdata = covid\_tbl), col = "red", lwd=2)

 The model shows that the number of deaths will increase, reach a peak and go down. So, I would recommend that the vaccine to be provided to as many people as possible as fast as possible and ease the lock down with great care.