

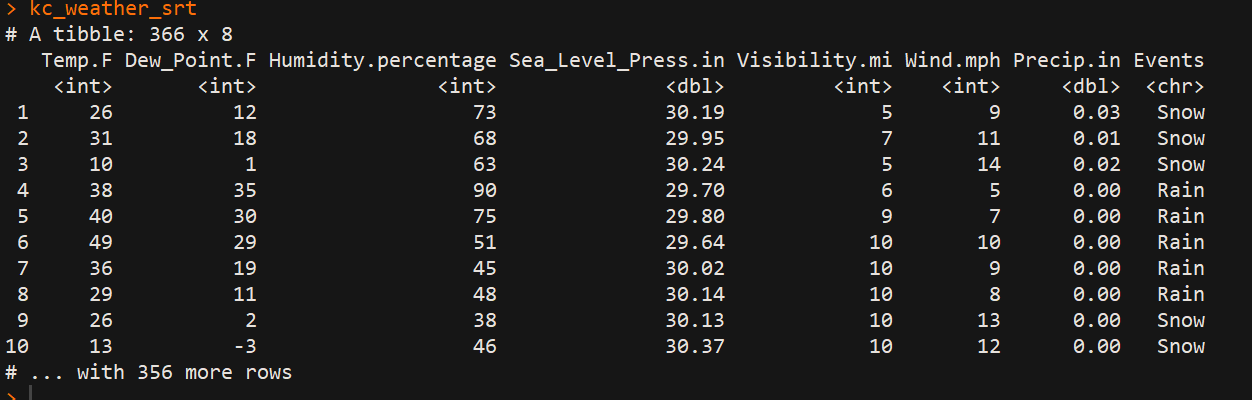
**PROGRAM**

*#import the dataset and make some changes*

library(readr)

kc\_weather\_srt <- read\_csv("C:/Users/bvkka/Desktop/ISL-Deep Medhi/kc\_weather\_srt.csv")

kc\_weather\_srt=kc\_weather\_srt[,2:9]



*#first make the response column to 0-snow, 1-rain and 2-rain\_thunderstorm*

*#install.packages("plyr")*

library(plyr)

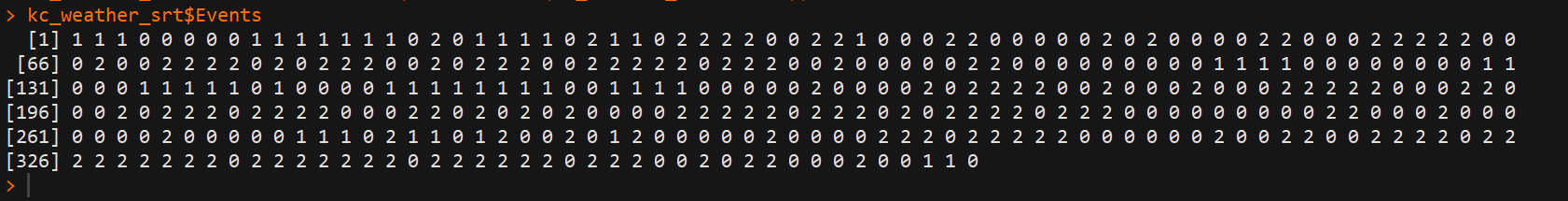
kc\_weather\_srt$Events <- revalue(kc\_weather\_srt$Events,c("Snow"=1))

kc\_weather\_srt$Events <- revalue(kc\_weather\_srt$Events,c("Rain"=0))

kc\_weather\_srt$Events <- revalue(kc\_weather\_srt$Events,c("Rain\_Thunderstorm"=2))

*#small changes to Events column , making it to numeric from character*

kc\_weather\_srt$Events<-as.numeric(as.character(kc\_weather\_srt$Events))



*#replications*

rep=100

*# newly added*

accuracy1=dim(rep)

precision\_snow1=dim(rep)

precision\_rain1=dim(rep)

precision\_rainThunderstorm1=dim(rep)

recall\_snow1=dim(rep)

recall\_rain1=dim(rep)

recall\_rainThunderstorm1=dim(rep)

*#splitting the dataset into training and test sets, also install caTools packages*

*#install.packages('caTools')*

library(caTools)

set.seed(123)

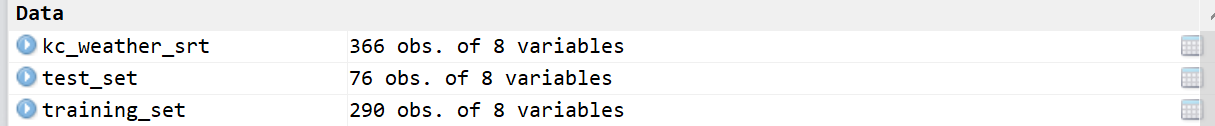
for(k in 1:rep)

{

split=sample.split(kc\_weather\_srt$Events,SplitRatio = 0.7923)

training\_set=subset(kc\_weather\_srt,split==TRUE)

test\_set=subset(kc\_weather\_srt,split==FALSE)



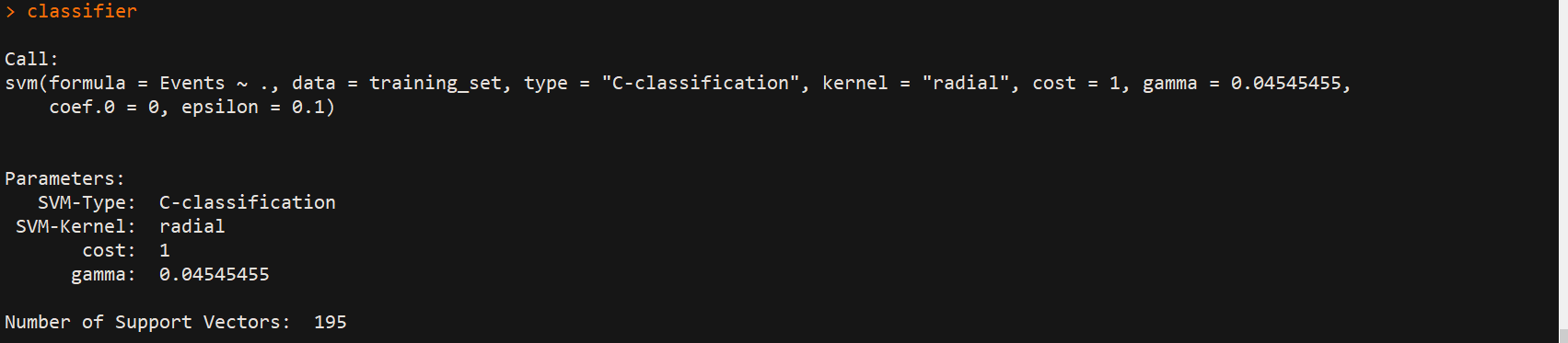
*#\*\*\*\*\*SVM\*\*\*\*\*#*

*#fitting SVM to the training set*

*#install.packages('e1071')*

library(e1071)

classifier=svm(Events~.,data=training\_set,type='C-classification',kernel="radial",cost=1,gamma=0.04545455,coef.0=0,epsilon=0.1)



y\_pred1=predict(classifier,newdata = test\_set[-8])

*#making the confusion matrix*

cm1=table(test\_set$Events,y\_pred1)



*#calculating the accuracy*

accuracy1[k]=mean(y\_pred1==test\_set$Events)

*#Precision of rain, rain\_thunderstorm and snow results*

precision1=precision1<-diag(cm1)/colSums(cm1)

precision\_rainThunderstorm1[k]=precision1[3]

precision\_snow1[k]=precision1[2]

precision\_rain1[k]=precision1[1]

*#Recall of rain, rain\_thunderstorm and snow results*

recall1=recall1<-diag(cm1/rowSums(cm1))

recall\_rainThunderstorm1[k]=recall1[3]

recall\_snow1[k]=recall1[2]

recall\_rain1[k]=recall1[1]

}

*#Calculating the end results using mean*

mean(accuracy1)

mean(precision\_rain1)

mean(precision\_rainThunderstorm1)

mean(precision\_snow1)

mean(recall\_rain1)

mean(recall\_rainThunderstorm1)

mean(recall\_snow1)

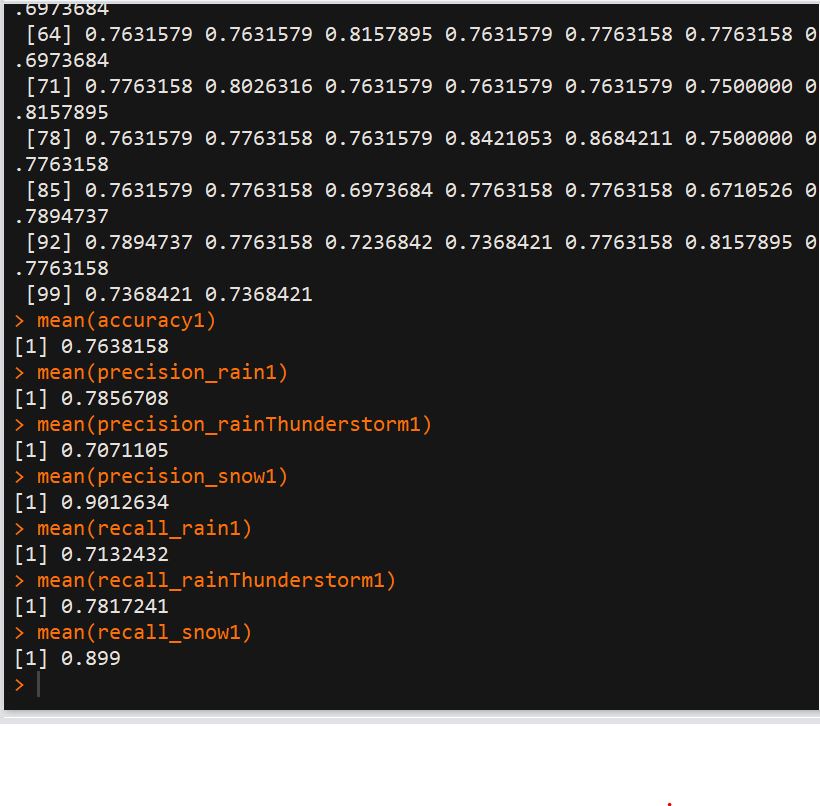
SVM Radial Kernel Results



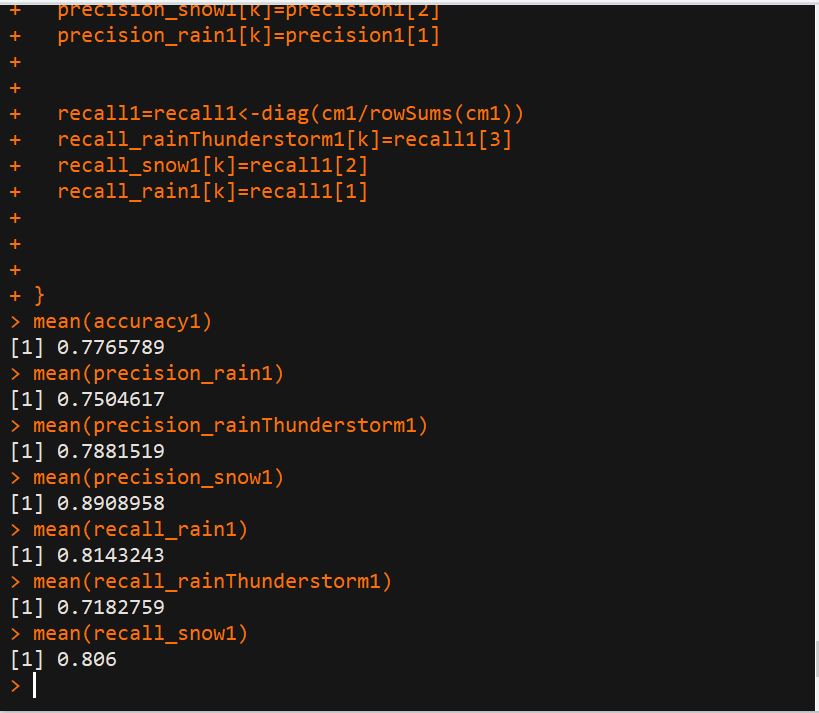
RESULTS:

**I also changed the tuning parameters under SVM tuning to see the best results. I have used Kernels like linear, radial and sigmoid with different cost and gamma parameters. We see some differences.**

**SVM Linear Results**



SVM RADIAL WITH GAMMA =0 AND COST =1 RESULTS



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Tuning Parameters | Accuracy | Precision Snow | Precision Rain | Precision Rain Thunderstorm | Recall Snow | Recall Rain | Recall ThunderStorm |
| SVM | kernel="radial”,cost=1, gamma=0.04545455,coef.0=0,epsilon=0.1 | 0.7736842 | 0.8981612 | 0.7938852 | 0.7231856 | 0.887 | 0.7278378 | 0.7931034 |
| SVM | kernel="linear" | 0.7638158 | 0.9012634 | 0.7856708 | 0.7071105 | 0.899 | 0.7132432 | 0.7817241 |
| SVM | kernel="radial",cost=1,gamma=0 | 0.7765789 | 0.8908958 | 0.7504617 | 0.7881519 | 0.806 | 0.8143243 | 0.7182759 |
| SVM | kernel="sigmoid”,cost=1, gamma=0.04545455,coef.0=0,epsilon=0.1 | 0.7515789 | 0.8825927 | 0.7829367 | 0.6924296 | 0.868 | 0.68 | 0.8027586 |

**Comparing to the other models using in Assignment 2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Model | Accuracy | Precision Snow | Precision Rain | Precision Rain Thunderstorm | Recall Snow | Recall Rain | Recall ThunderStorm |
| LDA | 0.9026316 | 0.6407459 | 0.9115871 | 0.9906168 | 0.911675 | 0.902705 | 0.9906152 |
| QDA | 0.7389474 | 0.945 | 0.697027 | 0.7213793 | 0.7950919 | 0.7514844 | 0.7115056 |
| KNN (K=5) | 0.745 | 0.895 | 0.7305405 | 0.7117241 | 0.9098042 | 0.7444542 | 0.701065 |

**Discussion Note:**

1. From Accuracy Results, we see that SVM model performs better than QDA and KNN, but LDA outperforms SVM too.
2. From Precision of Snow Results, SVM does better than LDA and KNN
3. From Precision of Rain Results, SVM does better than QDA and KNN, but less than LDA
4. From Precision of thunderstorm Results, SVM does better than QDA and KNN, but less than LDA.
5. From Recall of Snow Results, SVM does better than QDA and KNN
6. From Recall of Rain Results, SVM does better than QDA and KNN, but less than LDA
7. From Recall of thunderstorm Results, SVM does better than QDA and KNN, but less than LDA.
8. So, overall if we compare performance with respect to classifiers, LDA>SVM>KNN>QDA.