Assignment 3

Be sure to mark each problem # properly and your student ID (last 4 digits) shows up - no names, remember to number your pages. The submitted file should be PDF (preferably typed). Name your file with the last four digits of your student id followed by '-A3'. For example, if the last four digits of your ID are 1234, then the file name should be the following: 1234-A3.pdf

3_1 [30 points]. This assignment extends from Assignment-2. Q-1, which is reproduced below. Your assignment is to extend Part-b and use SVM, and provide a comparison with a discussion (note - need to do it only for 2.1 part-b).

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]
```

```
> kc_weather_srt
# A tibble: 366 x 8
   Temp.F Dew_Point.F Humidity.percentage Sea_Level_Press.in Visibility.mi Wind.mph Precip.in Events
    <int>       <int>               <int>              <dbl>         <int>    <int>     <dbl>  <chr>
 1     26          12                  73              30.19             5        9      0.03   Snow
 2     31          18                  68              29.95             7       11      0.01   Snow
 3     10           1                  63              30.24             5       14      0.02   Snow
 4     38          35                  90              29.70             6        5      0.00   Rain
 5     40          30                  75              29.80             9        7      0.00   Rain
 6     49          29                  51              29.64            10       10      0.00   Rain
 7     36          19                  45              30.02            10        9      0.00   Rain
 8     29          11                  48              30.14            10        8      0.00   Rain
 9     26           2                  38              30.13            10       13      0.00   Snow
10     13          -3                  46              30.37            10       12      0.00   Snow
# ... with 356 more rows
>
```

```
#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))
```

```
> kc_weather_srt$Events
  [1] 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 0 2 0 1 1 1 1 0 2 1 1 0 2 2 2 2 0 0 2 2 1 0 0 0 2 2 0 0 0 0 0 2 0 2 0 0 0 0 2 2 0 0 0 2 2 2 2 2 0 0
 [66] 0 2 0 0 2 2 2 2 0 2 0 2 2 2 0 0 2 0 2 2 2 0 0 2 2 2 2 2 0 2 2 2 0 0 2 0 0 0 0 0 2 2 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 1
[131] 0 0 0 1 1 1 1 1 0 1 0 0 0 0 1 1 1 1 1 1 1 1 0 0 1 1 1 1 0 0 0 0 0 2 0 0 0 0 2 0 2 2 2 2 0 0 2 0 0 0 2 0 0 0 2 2 2 2 2 0 0 0 2 2 0
[196] 0 0 2 0 2 2 2 0 2 2 2 2 0 0 0 2 2 0 2 0 2 0 2 0 0 0 0 2 2 2 2 2 0 2 2 2 0 2 0 2 2 2 2 2 0 2 2 2 0 0 0 0 0 0 0 0 0 2 2 0 0 0 2 0 0 0
[261] 0 0 0 0 2 0 0 0 0 0 1 1 1 0 2 1 1 0 1 2 0 0 2 0 1 2 0 0 0 0 0 2 0 0 0 0 2 2 2 0 2 2 2 2 2 2 0 0 0 0 0 2 0 0 2 2 0 0 2 2 2 2 0 2 2
[326] 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 2 0 2 2 2 2 2 2 2 0 2 2 2 2 0 0 2 0 2 2 0 0 0 2 0 0 1 1 0
>
```

```
#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)
```

Data	
kc_weather_srt	366 obs. of 8 variables
test_set	76 obs. of 8 variables
training_set	290 obs. of 8 variables

```
#*****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)
```

```
> classifier

Call:
svm(formula = Events ~ ., data = training_set, type = "C-classification", kernel = "radial", cost = 1, gamma = 0.04545455,
    coef.0 = 0, epsilon = 0.1)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.04545455

Number of Support Vectors:  195
```

```r
y_pred1=predict(classifier,newdata = test_set[-8])

#making the confusion matrix

cm1=table(test_set$Events,y_pred1)
```

```
> cm1
   y_pred1
      0  1  2
  0 24  0 13
  1  1  9  0
  2  7  0 22
>
```

```r
#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

```
> mean(accuracy1)
[1] 0.7736842
> mean(precision_rain1)
[1] 0.7938852
> mean(precision_rainThunderstorm1)
[1] 0.7231856
> mean(precision_snow1)
[1] 0.8981612
> mean(recall_rain1)
[1] 0.7278378
> mean(recall_rainThunderstorm1)
[1] 0.7931034
> mean(recall_snow1)
[1] 0.887
>
```

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

```
.6973684
 [64] 0.7631579 0.7631579 0.8157895 0.7631579 0.7763158 0.7763158 0
.6973684
 [71] 0.7763158 0.8026316 0.7631579 0.7631579 0.7631579 0.7500000 0
.8157895
 [78] 0.7631579 0.7763158 0.7631579 0.8421053 0.8684211 0.7500000 0
.7763158
 [85] 0.7631579 0.7763158 0.6973684 0.7763158 0.7763158 0.6710526 0
.7894737
 [92] 0.7894737 0.7763158 0.7236842 0.7368421 0.7763158 0.8157895 0
.7763158
 [99] 0.7368421 0.7368421
> mean(accuracy1)
[1] 0.7638158
> mean(precision_rain1)
[1] 0.7856708
> mean(precision_rainThunderstorm1)
[1] 0.7071105
> mean(precision_snow1)
[1] 0.9012634
> mean(recall_rain1)
[1] 0.7132432
> mean(recall_rainThunderstorm1)
[1] 0.7817241
> mean(recall_snow1)
[1] 0.899
>
```

SVM RADIAL WITH GAMMA =0 AND COST =1 RESULTS

```
+   precision_snow1[k]=precision1[2]
+   precision_rain1[k]=precision1[1]
+
+
+   recall1=recall1<-diag(cm1/rowSums(cm1))
+   recall_rainThunderstorm1[k]=recall1[3]
+   recall_snow1[k]=recall1[2]
+   recall_rain1[k]=recall1[1]
+
+
+
+ }
> mean(accuracy1)
[1] 0.7765789
> mean(precision_rain1)
[1] 0.7504617
> mean(precision_rainThunderstorm1)
[1] 0.7881519
> mean(precision_snow1)
[1] 0.8908958
> mean(recall_rain1)
[1] 0.8143243
> mean(recall_rainThunderstorm1)
[1] 0.7182759
> mean(recall_snow1)
[1] 0.806
>
```

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.