Question 2

Summary:

1. The breakpoints I am considering are 4.
   Temp.F – Based on range of numeric i.e., 0-80 I am taking as (0,30,60,90) .
   Humidity.percentage – Humidity ranges from 0-90 so break points are (0,30,60,90).
   Precip.in - Precip ranges from 0-2 so break points are (0,1,2,3)
2. When we consider only single predictor Temp.F the accuracy is less than the 3 predictors considered together.
3. Temp.F has low precision than Precip.in.
4. Precip.in has highest precision rate. So, the precision will be decreased when we eliminate this predictor.
5. Compared to LDA, QDA, KNN the naïve Bayes gives the moderately good accuracy with least relatively low error rate.
6. Both qualitative and quantitative predictors give almost the same accuracy, precision, and recall.
7. Temp.F, Humidity.percentage together gives higher accuracy than Temp.F, Humidity.percentage,Precip.in together.

Qualitative Predictors	Accuracy	Precision	Recall
Temp.F c(0,30,60,90) Humidity.percentage, c(0,30,60,90) Precip.in c(0,1,2,3)	0.746	0.767	0.9934
Temp.F c(0,30,60,90)	0.723	0.685	0.991
Temp.F c(0,33,66,99)	0.730	0.754	0.989
Temp.F c(0,50,100,150)	0.518	0.472	0.944
Humidity c(0,30,60,90)	0.523	0.350	1
Humidity c(30,50,70,90)	0.566	0.62	1
Precip c(0,1,2,3)	0.489	0.99	1
Precip c(0,0.5,1,1.5)	0.53	0.962	1
Temp.F c(0,33,66,99) Humidity.percentage, c(30,50,70,90) Precip.in c(0,0.5,1,1.5)	0.71	0.80	0.99

Quantitative Predictors	Accuracy	Precision	Recall
Temp.F, Humidity.percentage, Precip.in	0.747	0.84	0.9388
Temp.F	0.728	0.737	0.982
Humidity.percentage	0.575	0.604	1
Precip.in	0.381	0.94	0.54
Temp.F, Humidity.percentage	0.753	0.775	0.985
Temp.F, Precip.in	0.710	0.886	0.9396
Temp.F,Dew_Point.F ,Humidity.percentage ,SeaLevelPress.in,Visibility.mi,Wind.mph, Precip.in,	0.754	0.777	0.96

8.For quantitative predictors the naïve Bayes model with all predictors considered has higher accuracy but lower precision and recall.
9. We get very good recall when we consider the qualitative data.
10. Naïve Bayes classification considers all features are independent on class, so we find all the accuracy are almost same.

Now consider, from the KC weather data set, just the predictors: Temp.F, Humidity.percentage, Precip.in. Categorize these three data sets into qualitative predictors. It is up to you to decide on the break points, but you must discuss a rationale for your breakpoints. Now apply, naive Bayes Classifier on the entire data set (with these three qualitative predictors), using 290 of them as a training data set randomly (and the rest as the test data set), over 100 replications. Report on accuracy, precision, and recall.

Qualitative Temp.F, Humidity. Percentage, Percip.in Naïve Bayes:

```
> data=read.csv("kc_weather_srt.csv")
> data$CatTemp<-cut(data$Temp.F,breaks=c(0,30,60,90),labels = c("Low","Med","High"))
> data$CatHumid<-cut(data$Humidity.percentage,breaks = c(0,30,60,90),labels = c("H_Low","H_Med","H_High"))
> data$CatPrecip.in<-cut(data$Precip.in, breaks=c(0,1,2,3),labels = c("P_Low","P_Med","P_High"))
> data
        Date Temp.F Dew_Point.F Humidity.percentage Sea_Level_Press.in Visibility.mi Wind.mph Precip.in
1   2014-1-1     26          12                  73              30.19             5        9      0.03
2   2014-1-4     31          18                  68              29.95             7       11      0.01
3   2014-1-5     10           1                  63              30.24             5       14      0.02
4  2014-1-10     38          35                  90              29.70             6        5      0.00
5  2014-1-11     40          30                  75              29.80             9        7      0.00
6  2014-1-12     49          29                  51              29.64            10       10      0.00
```

```
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+   train=createDataPartition(data$Events,p=.80,list=FALSE)
+   dataTrain=data[train,]
+   dataTest=data[-train,]
+   e1071Model=naiveBayes(Events~CatTemp+CatHumid+CatPrecip.in,data=dataTrain)
+   e1071Predictions=predict(e1071Model,dataTest)
+   cm=confusionMatrix(e1071Predictions,dataTest$Events)
+   Accuracy[k]=cm$overall[1]
+   Precision[k]=cm$byClass[5]
+   Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7464384
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.7673333
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9934921
```

```
> data=read.csv("kc_weather_srt.csv")
> data$CatTemp<-cut(data$Temp.F,breaks=c(0,33,66,99),labels = c("Low","Med","High"))
> data$CatHumid<-cut(data$Humidity.percentage,breaks = c(30,50,70,90),labels = c("H_Low","H_Med","H_High"))
> data$CatPrecip.in<-cut(data$Precip.in, breaks=c(0,0.5,1,1.5),labels = c("P_Low","P_Med","P_High"))
> data
       Date Temp.F Dew_Point.F Humidity.percentage
1   2014-1-1     26          12                  73
2   2014-1-4     31          18                  68
3   2014-1-5     10           1                  63
4  2014-1-10     38          35                  90
```

```
> for(k in 1:100){
+
+   train=createDataPartition(data$Events,p=.80,list=FALSE)
+   dataTrain=data[train,]
+   dataTest=data[-train,]
+   e1071Model=naiveBayes(Events~CatTemp+CatHumid+CatPrecip.in,data=dataTrain)
+   e1071Predictions=predict(e1071Model,dataTest)
+   cm=confusionMatrix(e1071Predictions,dataTest$Events)
+   Accuracy[k]=cm$overall[1]
+   Precision[k]=cm$byClass[5]
+   Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7127397
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.8057778
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9909524
```

Qualitative Temp.F Naïve Bayes with break points as 0,30,60,90:

```
> library(caret)
> library(e1071)
>
> data=read.csv("kc_weather_srt.csv", stringsAsFactors=TRUE)
> data$CatTemp<-cut(data$Temp.F,breaks=c(0,30,60,90),labels = c("Low","Med","High"))
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~CatTemp,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7231507
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.6857778
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9919048
```

Qualitative Temp.F Naïve Bayes with break points as 0,33,66,99:

```
> data=read.csv("kc_weather_srt.csv", stringsAsFactors=TRUE)
> data$CatTemp<-cut(data$Temp.F,breaks=c(0,33,66,99),labels = c("Low","Med","High"))
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~CatTemp,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7306849
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.7542222
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9896825
> library(caret)
> library(e1071)
```

Qualitative Temp.F Naïve Bayes with break points as 0,33,66,99:

```
> data=read.csv("kc_weather_srt.csv", stringsAsFactors=TRUE)
> data$CatTemp<-cut(data$Temp.F,breaks=c(0,50,100,150),labels = c("Low","Med","High"))
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~CatTemp,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.5182192
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.4726667
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9944444
```

Quantitative Temp.F model

```
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Temp.F,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7289041
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.7377778
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9828571
```

Quantitative Temp.F, Humidity. Percentage, Percip.in Naïve Bayes:

```
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Temp.F+Humidity.percentage+Precip.in,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7473973
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.8477778
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9388889
```

Quantitative Humidity.percentage model

```
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Humidity.percentage,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.5757534
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.6044444
> meanRecall=mean(Recall)
> meanRecall
[1] 1
```

Quantitative Precip.in model

```
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Precip.in,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.3819178
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.94
> meanRecall=mean(Recall)
> meanRecall
[1] 0.5442857
```

Quantitative Temp.F, Humidity. Percentage Naïve Bayes:

```
> Accuracy=dim(100)
> Precision=dim(100)
> Recall=dim(100)
>
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Temp.F+Humidity.percentage,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7538356
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.7753333
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9857143
```

Quantitative Temp.F,Percip.in Naïve Bayes:

```
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Temp.F+Precip.in,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.710411
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.8868889
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9396825
```

Quantitative Temp.F, Dew_Point.F, Humidity.percentage, Sea_Level_Press.in, Visibility.mi, Wind.mph, Precip.in Naïve Bayes:

```
> for(k in 1:100){
+
+ train=createDataPartition(data$Events,p=.80,list=FALSE)
+ dataTrain=data[train,]
+ dataTest=data[-train,]
+ e1071Model=naiveBayes(Events~Temp.F+Dew_Point.F+Humidity.percentage+Sea_Level_Press.in
+                   +Visibility.mi+Wind.mph+Precip.in,data=dataTrain)
+ e1071Predictions=predict(e1071Model,dataTest)
+ cm=confusionMatrix(e1071Predictions,dataTest$Events)
+ Accuracy[k]=cm$overall[1]
+ Precision[k]=cm$byClass[5]
+ Recall[k]=cm$byClass[6]
+ }
> meanAccuracy=mean(Accuracy)
> meanAccuracy
[1] 0.7549315
> meanPrecision=mean(Precision)
> meanPrecision
[1] 0.7775556
> meanRecall=mean(Recall)
> meanRecall
[1] 0.9615873
```

References:

1. https://www.rdocumentation.org/packages/klaR/versions/0.6-12/topics/plot.NaiveBayes
2. https://cran.r-project.org/web/packages/naivebayes/naivebayes.pdf