Homework 1 Problem 1

CS 498, Spring 2018, Xiaoming Ji

```
library(readr)
data  = read.csv("pima-indians-diabetes.data.csv", header = FALSE)
#head(data)
all_x = data[,-c(9)]
all_y = data[,9]
```

Part A

Build a simple naive Bayes classifier to classify this data set. We will use 20% of the data for evaluation and the other 80% for training. There are a total of 768 data-points.

We use a normal distribution to model each of the class-conditional distributions.

```
#Prepare data
set.seed(19720816)
train_len = round(dim(all_x)[1] * 0.8)
train_index = sample(1:dim(all_x)[1], round(dim(all_x)[1] * 0.8))
train_x = all_x[train_index,]
train_y = all_y[train_index]
eval_x = all_x[-train_index,]
eval_y = all_y[-train_index]
```

We first calculate p(y=1) and p(y=0)

```
p_y_1 = length(train_y[train_y == 1]) / train_len
p_y_0 = 1 - p_y_1
```

$$\frac{1}{\sigma_k\sqrt{2\pi}}e^{-(x-\mu_k)^2/2\sigma_k^2}$$

```
wdat<-read.csv('pima-indians-diabetes.data.csv', header=FALSE)
library(klaR)
```

```
## Loading required package: MASS
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.4.3
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
## Warning in as.POSIXlt.POSIXct(Sys.time()): unknown timezone 'default/Asia/
## Shanghai'
```

```
bigx<-wdat[,-c(9)]
bigy<-wdat[,9]
trscore<-array(dim=10)
tescore<-array(dim=10)
for (wi in 1:10)
{wtd<-createDataPartition(y=bigy, p=.8, list=FALSE)
```

```
 nbx<-bigx
 ntrbx<-nbx[wtd, ]
 ntrby<-bigy[wtd]
 trposflag<-ntrby>0
 ptregs<-ntrbx[trposflag, ]
 ntregs<-ntrbx[!trposflag,]
 ntebx<-nbx[-wtd, ]
 nteby<-bigy[-wtd]
 ptrmean<-sapply(ptregs, mean, na.rm=TRUE)
 ntrmean<-sapply(ntregs, mean, na.rm=TRUE)
 ptrsd<-sapply(ptregs, sd, na.rm=TRUE)
 ntrsd<-sapply(ntregs, sd, na.rm=TRUE)
 ptroffsets<-t(t(ntrbx)-ptrmean)
 ptrscales<-t(t(ptroffsets)/ptrsd)
 ptrlogs<--(1/2)*rowSums(apply(ptrscales,c(1, 2), function(x)x^2), na.rm=TRUE)-sum(log(ptrsd))
 ntroffsets<-t(t(ntrbx)-ntrmean)
 ntrscales<-t(t(ntroffsets)/ntrsd)
 ntrlogs<--(1/2)*rowSums(apply(ntrscales,c(1, 2), function(x)x^2), na.rm=TRUE)-sum(log(ntrsd))
 lvwtr<-ptrlogs>ntrlogs
 gotrighttr<-lvwtr==ntrby
 trscore[wi]<-sum(gotrighttr)/(sum(gotrighttr)+sum(!gotrighttr))
 pteoffsets<-t(t(ntebx)-ptrmean)
 ptescales<-t(t(pteoffsets)/ptrsd)
 ptelogs<--(1/2)*rowSums(apply(ptescales,c(1, 2), function(x)x^2), na.rm=TRUE)-sum(log(ptrsd))
 nteoffsets<-t(t(ntebx)-ntrmean)
 ntescales<-t(t(nteoffsets)/ntrsd)
 ntelogs<--(1/2)*rowSums(apply(ntescales,c(1, 2), function(x)x^2), na.rm=TRUE)-sum(log(ntrsd))
 lvwte<-ptelogs>ntelogs
 gotright<-lvwte==nteby
 tescore[wi]<-sum(gotright)/(sum(gotright)+sum(!gotright))
}
```