

Question 2.1

#I work as an International Education Consultant and we help many students study abroad. We utilize multiple information to classify our students.

#1. Country that they want to study abroad at

#2. degree that they are seeking to achieve

#3. Specific schools that they have in mind

#4. Student education background

#5. Foreign language ability

Question 2.2.1

```
library(kernlab) #import library ksvm
```

```
#read data
```

```
data <- read.table("/Users/xiaofanjiao/Desktop/credit_card_data.txt",  
header= F, stringsAsFactors = F)
```

```
#quick look of the data
```

```
head(data)
```

```
tail(data)
```

```
# utilize ksvm model
```

```
model <- ksvm (as.matrix(data[,1:10]), as.matrix (data[,11]),  
              type="C-svc",  
              kernel ="vanilladot",  
              C= 100, scaled= TRUE)
```

```
model
```

```
#calculating the coefficients (a1..am)
```

```
a <- colSums(model@xmatrix[[1]] * model@coef[[1]])
```

```
a
```

```
# calculate a0
```

```
a0<- -model@b
```

```
a0
```

```
# see what the model predictions are
```

```
pred <- predict(model,data[,1:10])
```

```
pred
```

```
# see what fraction of the model's predictions match the actual  
classification
```

```
sum(pred == data$V11) / nrow(data)
```

```
## Output: 0.8639144
```

Quesiton 2.2.3

```
# Load Libray
```

```
library(kknn)
```

```
#read data
```

```
data <- read.table("/Users/xiaofanjiao/Desktop/credit_card_data.txt",  
header= F, stringsAsFactors = F)
```

```
n<-nrow(data)
```

```
#use 20 zeroes vector for accuracy test
```

```

for (k_value in 1:20){
  results<-rep(0,n)
  for (i in 1:n){
    model <- kknn(V11~V1+V2+V3+V4+V5+V6+V7+V8+V9+V10, data[-i,],data[i,],
k= k_value, scale= TRUE)
    results[i] = as.integer(fitted(model)+0.5) #round the result
  }
  #Calculate correct prediction
  s<- sum(results == data[1:n,11])
  #match percentage
  m = s/n
  print(c(k_value,m))
}

```

```

##[1] 1.0000000 0.8149847
##[1] 2.0000000 0.8149847
##[1] 3.0000000 0.8149847
##[1] 4.0000000 0.8149847
##[1] 5.0000000 0.851682
##[1] 6.0000000 0.8455657
##[1] 7.0000000 0.8470948
##[1] 8.0000000 0.8486239
##[1] 9.0000000 0.8470948
##[1] 10.0000000 0.8501529
##[1] 11.0000000 0.851682
##[1] 12.0000000 0.853211
##[1] 13.0000000 0.851682
##[1] 14.0000000 0.851682
##[1] 15.0000000 0.853211
##[1] 16.0000000 0.851682
##[1] 17.0000000 0.851682
##[1] 18.0000000 0.851682
##[1] 19.0000000 0.8501529
##[1] 20.0000000 0.8501529

```