

# Homework1

N/A

2023-11-16

```
library(kernlab)
library(class)
```

Read the CSV file into a data frame, convert the data frame into a matrix

```
dataset = read.csv("credit_card.csv", header = TRUE)
dataset_matrix = as.matrix(dataset)
model = ksvm(dataset_matrix[, 1:10], dataset_matrix[, 11], type = "C-svc", kernel = "vanilladot", C = 200, scaled = TRUE)
```

```
## Setting default kernel parameters
```

```
#calculation of a1...am
a = colSums(model@xmatrix[[1]] * model@coef[[1]])
decimals = format(a, scientific = FALSE)
print(decimals)
```

```
##           A1           A2           A3           A8
## "-0.00035658041" "-0.00005234880" "-0.00016509945" " 0.00111617077"
##           A9           A10           A11           A12
## " 1.00758841864" "-0.00047342530" "-0.00005133491" "-0.00003788560"
##           A14           A15
## "-0.00006765545" " 0.10600825114"
```

#Equation for Classifier: #y = -0.00035658041A1 - 0.00005234880A2 - 0.00016509945 \* A3 +0.00111617077 A4 + # 1.00758841864 A5 + -0.00047342530 \* A6 + -0.00005133491 \* A7 - 0.00003788560 \* A8 # + -0.00006765545 \* A9 + 0.10600825114 \* A10"

```
a0 = model@b
print(a0)
```

```
## [1] -0.0814246
```

```
pred = predict(model,dataset_matrix[,1:10])
```

```
# see what fraction of the model's predictions match the actual classification
sum(pred == dataset_matrix[,11]) / nrow(dataset_matrix)
```

```
## [1] 0.8639144
```

```
#KNN Classifier:

# Create empty lists to store results
scaled_X_list <- list()
y_list <- list()

# Iterate through each row index

for (i in 1:nrow(dataset_matrix)) {
  # Select predictor variables for all rows except i
  X <- scale(dataset_matrix[-i, 1:10])
  scaled_X_list[[i]] <- X

  # Select response variable for all rows except i
  y <- dataset_matrix[-i, 11]
  y_list[[i]] <- y
}

knn_instance = knn(X, test = X, cl= y, k = 3) # I tried k-values of 3,5 and 7.

predictions = table(y, knn_instance)
predictions
```

```
##      knn_instance
## y      0      1
## 0 328  29
## 1  34 262
```

#Based on experimentation, the k-value that produced the best results was a k-value of 3. #I have summarized the results into a confusion matrix from the table() method. The best # knn classifier produced 329 true negative(actual 0, predicted 0), 29 false positives(actual 0, #predicted 1, 33 false negatives( actual 1, predicted 0) and 262 true positive(actual 1, predicted 1).