Machine Learning Classification of E. coli Protein Localization Sites

Srikant Vasudevan

11/4/2019

ABSTRACT

Nakai and Kanehisa (1992) Nakai and Kanehisa (1991) Asuncion and Newman (2007) Horton et al. (2007) Deng et al. (2016) Min et al. (2009)

Enterohemorrhagic Escherichia coli, better known as E. coli, is a worldwide foodborne pathogen. E. coli is known for surviving well within healthy cattle, fresh produce and in the intestines of many animals. An important, beneficial process that E. coli undergoes is called protein localization. Protein localization, given simply, is the accumulation of a certain protein at a specific site in order for subsequent subcellular processes to be carried out in cellular regions.

Signal sequence recognition is a method of determing the signal peptides (signal sequences) involved in the transport of proteins through different cellular compartments. Signal sequences contain several different structural components, each of which play a key part in how it transports proteins effectively.

INTRODUCTION

Using data analysis tools such as R and R Studio, a machine learning classifier will be trained on a multivariate dataset (obtained from the UCI Machine Learning Repository), which includes E. coli attributes and several protein localization sites. This classifier will be created with the incorporation of multiple R libraries and external algorithms, most notably being a "knn" algorithm (k-nearest neighbor algorithm). The classifier will produce a prediction algorithm, from a training set to predict the protein localization sites of a testing set (both the training and testing sets are obtained from the multivariable dataset from the UCI ML Repository).

K-nearest neighbor algorithms are widely used in classification and regression tasks. Knn is known as "instance-based learning" because the function is based locally and the calculations are only performed in the classification process. This method assigns components called "weights" to the contribution of the neighbors (data points) to the overall mean of a certain attribute. Any set of data points that has known values for the classification property can be used as the "training set" for this method.

The dataset contains 8 variables and 336 observations and was donated to the UCI Machine Learning Repository in 1996 by Kenta Nakai of the Institute of Molecular and Cellular Biology.

The dataset variables are:

1. Sequence Name: Accession number for the SWISS-PROT database (factor)
2. mcg: McGeoch's method for signal sequence recognition. (double)
3. gvh: von Heijne's method for signal sequence recognition. (double)
4. lip: von Heijne's Signal Peptidase II consensus sequence score. (boolean)
5. chg: Presence of charge on N-terminus of predicted lipoproteins. (boolean)
6. aac: score of discriminant analysis of the amino acid content of the outer membrane and periplasmic proteins.(double)
7. alm1: score of the ALOM membrane-spanning region prediction program.(double)
8. alm2: score of ALOM program after excluding putative cleavable signal regions from the sequence.(double)
9. pls: protein localization site (factor)

For the instance of this classifier, a distinct discrete variable will assigned to each given protein localization site:

1. cp
2. im
3. imL
4. imS
5. imU
6. om
7. omL
8. pp

METHODS

In this classification report, multiple graphical visualizations are used to aid in the process of preliminary analysis and lead us to our final knn prediction.

1. The dataset used in this analysis was downloaded from the UCI Machine Learning repository in the form of a .data file. The .data file was converted into a .csv file and headers were added manually
2. To analyze the .csv file, R (an open-source statistical programming language) and RStudio (a free IDE for the R language) are used, these tools read the contents of the .csv file and format them into a data frame.
3. Within R, we will be using a process called "data munging", this process is used to clean and format the data to be free of errors and easily understood. This process includes replacing or eliminating missing variables, renaming variable headers to either be more understandable or more concise and logically organizing the data so it is in a sensical order, prepped for analysis. The dataset used for this classifier has minimal "dirty data" and is a fairly clean dataset, therefore this step will be less extensive.
4. Additionally, a multitude of R packages will be used in the analysis process, these include class, ggvis, gmodels, tidyverse, caret, GGally, gridExtra.
5. To analyze our data, multitudes of visual and numerical representations are used. Ggplot (R library) pairwise plots are used to effectively visualize the different protein localization sites; values such as correlation coefficients, distribution statistics and linear regression models (with localization sites color-coded) are presented to aid in such analysis.

RESULTS

Setup

To prepare the data frame, a working directory needs to be set. This directory will allow for any files (in this case the .csv file) to be accessed from the same local directory. In addition to this, all libraries should be loaded and the .csv file should be read to a dataset.

```
#Working Directory
setwd("C:/Users/Srikant/Desktop/Data Science/Week 10")
#Warnings OFF
options(warn=-1)
#Loading libraries
library(class)
library(ggvis)
library(gmodels)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------------
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(caret)
```

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

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

##
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':
##
##     nasa
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
#Read dataset
dataset <- read.csv("./ecoli.csv")
```

Descriptive Statistics

Now, since the dataset is set up, the display of several different descriptive statistics, including summary(), dim(), str(), glimpse() and head() will allow for a better understanding of the dataset. The goal is to use these descriptive statistics to not only interpret information about the dataset (basic dimensions and setup) as well as to assess the cleanliness of the dataset (lack of missing values, acceptable variable names, etc.).

```
summary(dataset)
```

```
##          seq           mcg              gvh             lip
##  AAS_ECOLI :  1   Min.   :0.0000   Min.   :0.16   Min.   :0.4800
##  AAT_ECOLI :  1   1st Qu.:0.3400   1st Qu.:0.40   1st Qu.:0.4800
##  ACEA_ECOLI:  1   Median :0.5000   Median :0.47   Median :0.4800
##  ACEK_ECOLI:  1   Mean   :0.5001   Mean   :0.50   Mean   :0.4955
##  ACKA_ECOLI:  1   3rd Qu.:0.6625   3rd Qu.:0.57   3rd Qu.:0.4800
##  ADI_ECOLI :  1   Max.   :0.8900   Max.   :1.00   Max.   :1.0000
##  (Other)   :330
##       chg              aac             alm1             alm2
##  Min.   :0.5000   Min.   :0.000   Min.   :0.0300   Min.   :0.0000
##  1st Qu.:0.5000   1st Qu.:0.420   1st Qu.:0.3300   1st Qu.:0.3500
##  Median :0.5000   Median :0.495   Median :0.4550   Median :0.4300
##  Mean   :0.5015   Mean   :0.500   Mean   :0.5002   Mean   :0.4997
##  3rd Qu.:0.5000   3rd Qu.:0.570   3rd Qu.:0.7100   3rd Qu.:0.7100
##  Max.   :1.0000   Max.   :0.880   Max.   :1.0000   Max.   :0.9900
##
##       pls
##  cp     :143
##  im     : 77
```

```
##  pp     : 52
##  imU    : 35
##  om     : 20
##  omL    :  5
##  (Other):  4
```

```
head(dataset)
```

```
##          seq  mcg  gvh  lip chg  aac alm1 alm2 pls
## 1  AAT_ECOLI 0.49 0.29 0.48 0.5 0.56 0.24 0.35  cp
## 2 ACEA_ECOLI 0.07 0.40 0.48 0.5 0.54 0.35 0.44  cp
## 3 ACEK_ECOLI 0.56 0.40 0.48 0.5 0.49 0.37 0.46  cp
## 4 ACKA_ECOLI 0.59 0.49 0.48 0.5 0.52 0.45 0.36  cp
## 5  ADI_ECOLI 0.23 0.32 0.48 0.5 0.55 0.25 0.35  cp
## 6 ALKH_ECOLI 0.67 0.39 0.48 0.5 0.36 0.38 0.46  cp
```

```
dim(dataset)
```

```
## [1] 336   9
```

```
glimpse(dataset)
```

```
## Observations: 336
## Variables: 9
## $ seq  <fct> AAT_ECOLI, ACEA_ECOLI, ACEK_ECOLI, ACKA_ECOLI, ADI_ECOLI,...
## $ mcg  <dbl> 0.49, 0.07, 0.56, 0.59, 0.23, 0.67, 0.29, 0.21, 0.20, 0.4...
## $ gvh  <dbl> 0.29, 0.40, 0.40, 0.49, 0.32, 0.39, 0.28, 0.34, 0.44, 0.4...
## $ lip  <dbl> 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.4...
## $ chg  <dbl> 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0....
## $ aac  <dbl> 0.56, 0.54, 0.49, 0.52, 0.55, 0.36, 0.44, 0.51, 0.46, 0.5...
## $ alm1 <dbl> 0.24, 0.35, 0.37, 0.45, 0.25, 0.38, 0.23, 0.28, 0.51, 0.1...
## $ alm2 <dbl> 0.35, 0.44, 0.46, 0.36, 0.35, 0.46, 0.34, 0.39, 0.57, 0.3...
## $ pls  <fct> cp, cp, cp, cp, cp, cp, cp, cp, cp, cp, cp, cp, cp, cp, c...
```

```
tail(dataset)
```

```
##            seq  mcg  gvh  lip chg  aac alm1 alm2 pls
## 331 TORA_ECOLI 0.43 0.59 0.48 0.5 0.52 0.49 0.56  pp
## 332 TREA_ECOLI 0.74 0.56 0.48 0.5 0.47 0.68 0.30  pp
## 333 UGPB_ECOLI 0.71 0.57 0.48 0.5 0.48 0.35 0.32  pp
## 334 USHA_ECOLI 0.61 0.60 0.48 0.5 0.44 0.39 0.38  pp
## 335 XYLF_ECOLI 0.59 0.61 0.48 0.5 0.42 0.42 0.37  pp
## 336 YTFQ_ECOLI 0.74 0.74 0.48 0.5 0.31 0.53 0.52  pp
```

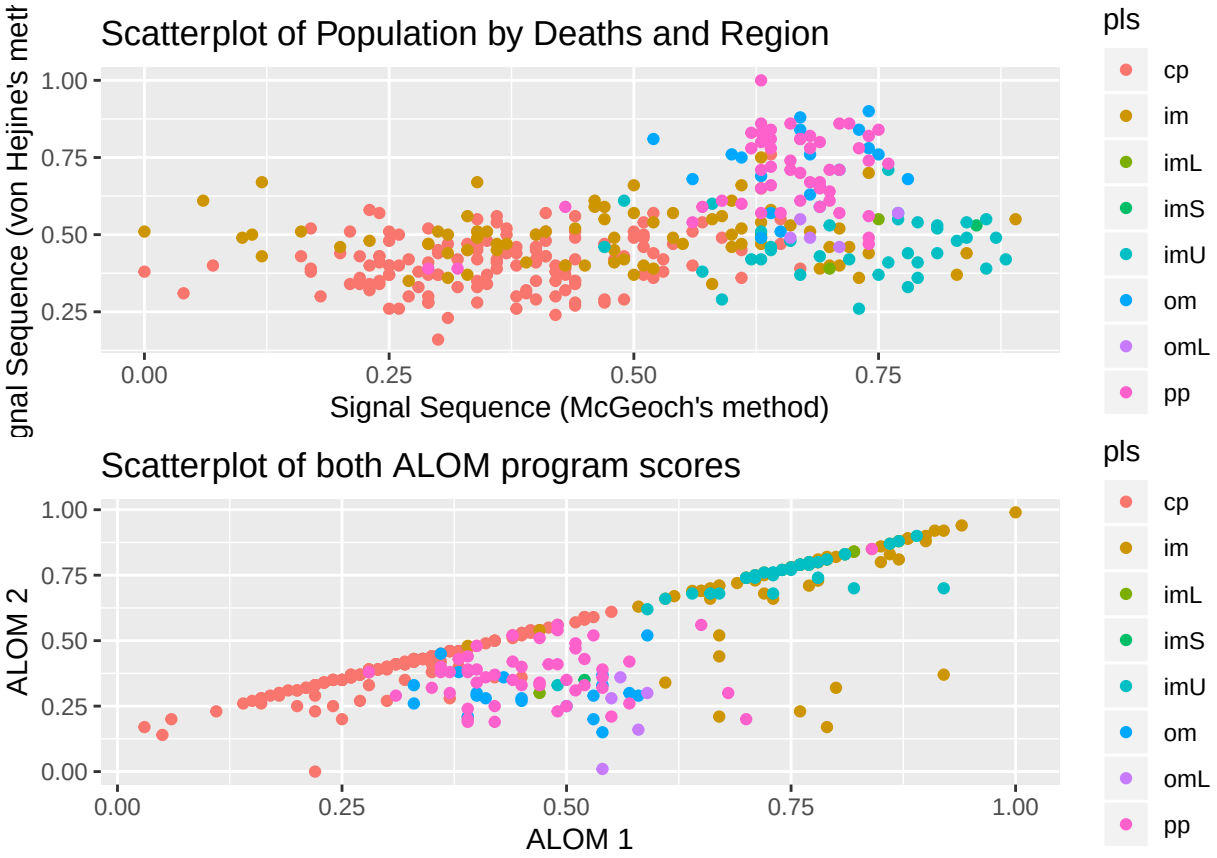
Through a quick glimpse of the descriptive statistics, it is seen that there are 336 observations with 9 variables within our dataset. Also by looking at the summary of the dataset, it is seen that there are no mising values in the dataset and that the variable names are acceptable for our analysis.

Graphical Visualization

Aside from the descriptive statistics of the dataset, different graphical visualizations can help in grasping a better understanding of the data. The two that will be used in this classification are a "gg" pairwise plot and a "gg" scatterplot, both with color-coding.

```
#Create two different scatterplots that each determine correlation between two variables
sp1 <- ggplot(data=dataset, aes(x=mcg, y=gvh))
sp1 <- sp1 + geom_point(aes(col=pls))
sp1 <- sp1 + labs(title = "Scatterplot of Population by Deaths and Region",
```

```
                  y="Signal Sequence (von Hejine's method)",
                  x="Signal Sequence (McGeoch's method)")
sp2 <- ggplot(data=dataset, aes(x=alm1, y=alm2))
sp2 <- sp2 + geom_point(aes(col=pls))
sp2 <- sp2 + labs(title = "Scatterplot of both ALOM program scores",
                  y="ALOM 2", x="ALOM 1")
grid.arrange(sp1, sp2)
```

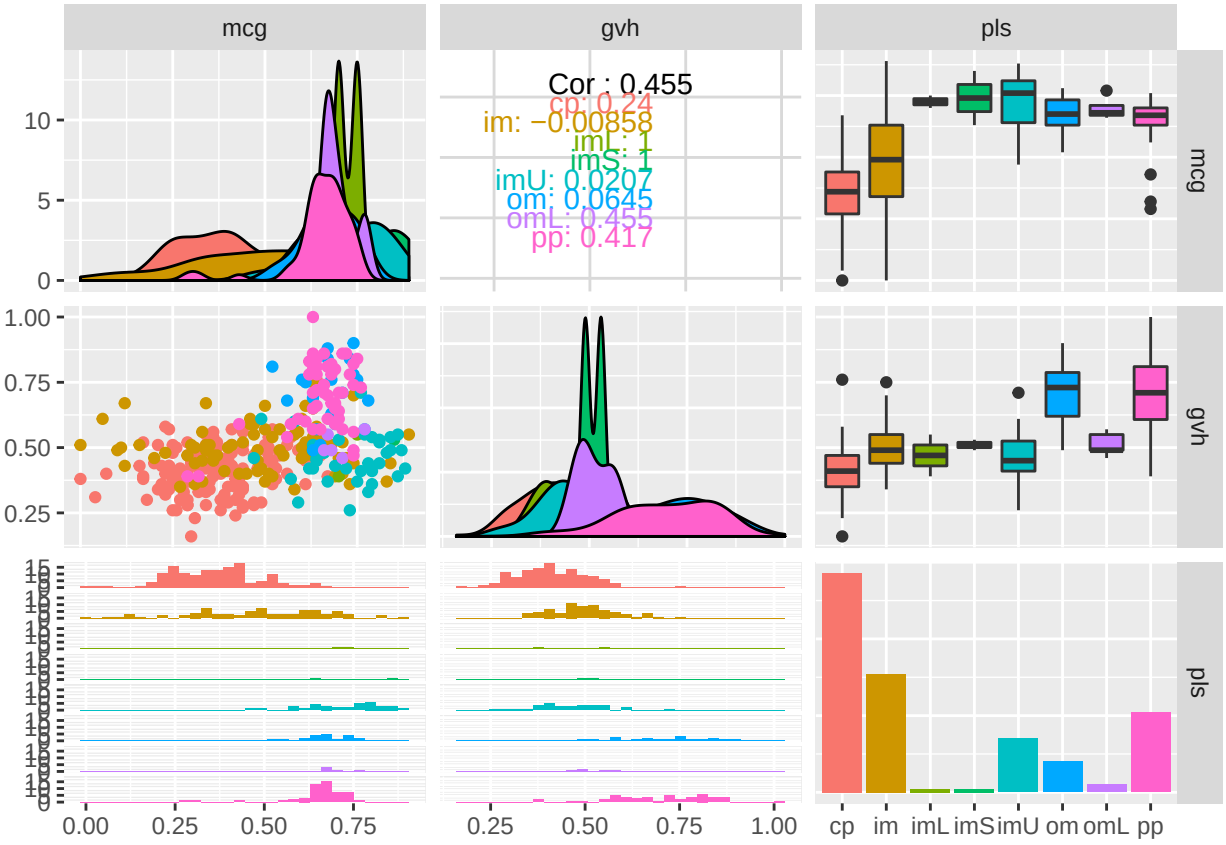


In these two scatterplots, a few things can be noticed, most notably, there is an apparent positive correleation between the signal sequences using von Hejine's method and McGeoch's method, which is expected. There is also an apparent positive correlation, though less strong, between the two ALOM scores.

Aside from the correlations, another key piece of information can be obtained from these graphs: there are clear distinctions between different protein localization sites (different colors on the graphs), leading us to believe that the graphed values contribute to the different classifications.

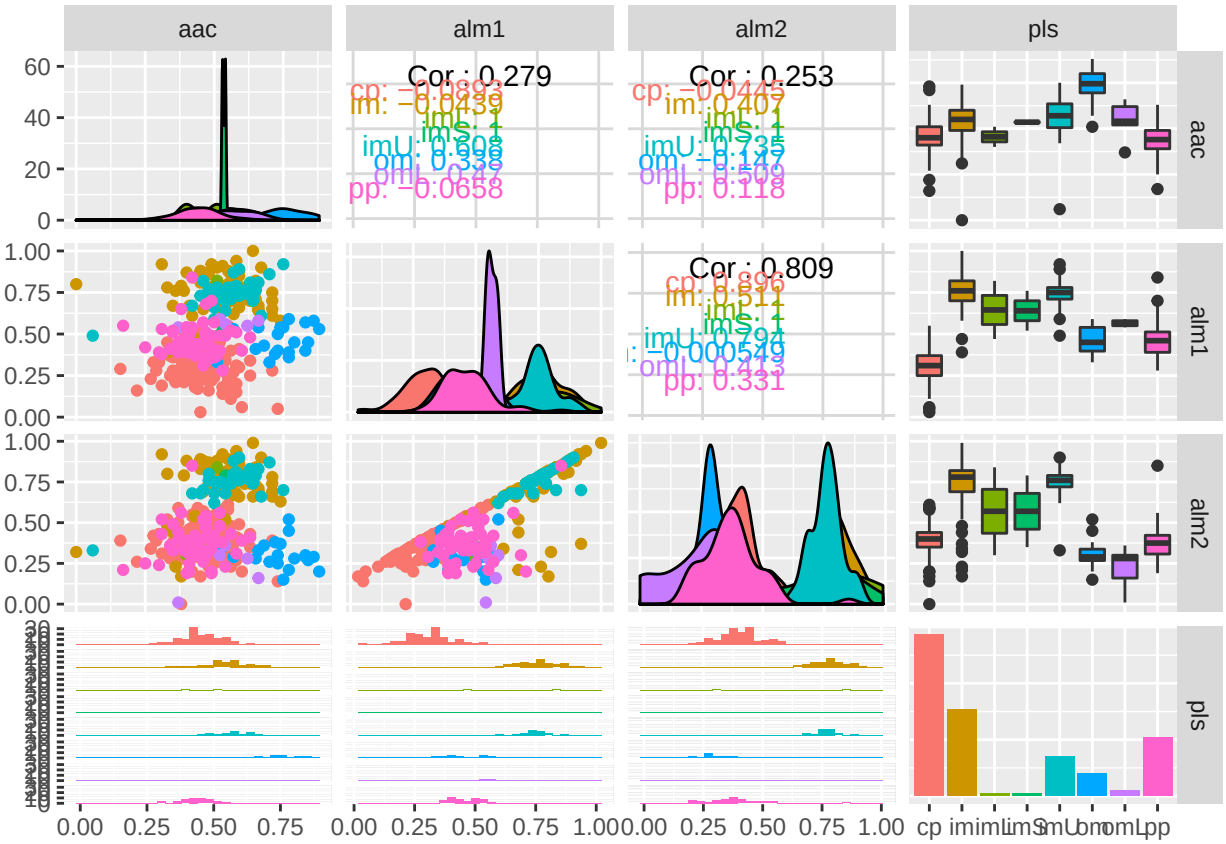
```
#Separate the dataset in to smaller sets for better visualization
#Exclude boolean values
short <- dataset[, c(2:3, 9)]
short1 <- dataset[, c(6:8, 9)]
plot1 <- ggpairs(short, aes(color=(pls)), size = 8)
plot2 <- ggpairs(short1, aes(color=(pls)), size = 8)
plot1
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
plot2
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



These "gg" pairwise plots contain a plethora of information, correlation between different values, distribution graphs of different variables and comparitive box and bar plots representing different two-variable distributions. The most important data from these plots is the distribution of different variables and correlation between different values. Though touched on in the scatterplots, the pairwise plots qualify the expectation that the alm1 and alm2 values are closely related. The pairwise plots also provide vital information on individual distributions of variables, which seem to be approximately normal or bimodal for all of the protein localization sites, further qualifying the expectation that there is a distinction in most variables for each protein localization site.

Knn Classification

To start the knn classification process, there is quite a lot of preparation that needs to be done to the dataset. A randomization seed needs to be set to allow for randomization during the sampling stage. A new column will be mutated to the dataset with values representing protein localization sites, but instead of factor values, they will be numeric values (numeric assignments shown in introduction). Using this new column and the randomization seed, an index of 1s and 2s (training set and testing set respectively) will be sampled. The probability for this sample will be 65% 1s (training set) and 35% 2 (testing set).

```
#Set seed for randomization
set.seed(12345)
#New column with values 1-8 which will allow us to numerically identify the "pls" values
dataset <- mutate(dataset, variable_class = as.numeric(dataset$pls))

summary(dataset)
```

```
##          seq           mcg               gvh             lip
##  AAS_ECOLI :  1   Min.   :0.0000   Min.   :0.16   Min.   :0.4800
##  AAT_ECOLI :  1   1st Qu.:0.3400   1st Qu.:0.40   1st Qu.:0.4800
##  ACEA_ECOLI:  1   Median :0.5000   Median :0.47   Median :0.4800
##  ACEK_ECOLI:  1   Mean   :0.5001   Mean   :0.50   Mean   :0.4955
##  ACKA_ECOLI:  1   3rd Qu.:0.6625   3rd Qu.:0.57   3rd Qu.:0.4800
##  ADI_ECOLI :  1   Max.   :0.8900   Max.   :1.00   Max.   :1.0000
##  (Other)   :330
##       chg              aac             alm1             alm2
##  Min.   :0.5000   Min.   :0.000   Min.   :0.0300   Min.   :0.0000
##  1st Qu.:0.5000   1st Qu.:0.420   1st Qu.:0.3300   1st Qu.:0.3500
##  Median :0.5000   Median :0.495   Median :0.4550   Median :0.4300
##  Mean   :0.5015   Mean   :0.500   Mean   :0.5002   Mean   :0.4997
##  3rd Qu.:0.5000   3rd Qu.:0.570   3rd Qu.:0.7100   3rd Qu.:0.7100
##  Max.   :1.0000   Max.   :0.880   Max.   :1.0000   Max.   :0.9900
##
##       pls      variable_class
##  cp     :143   Min.   :1.000
##  im     : 77   1st Qu.:1.000
##  pp     : 52   Median :2.000
##  imU    : 35   Mean   :3.146
##  om     : 20   3rd Qu.:5.000
##  omL    :  5   Max.   :8.000
##  (Other):  4
```

```
#Split the dataset into 1s and 2s (indexing)
ind <- sample(2, nrow(dataset), replace=TRUE, prob=c(.65, .35))
ind
```

```
##   [1] 2 2 2 2 1 1 1 1 2 2 1 1 2 1 1 1 1 1 1 2 1 1 2 2 1 1 2 1 1 1 2 1 1 2 1
##  [36] 1 2 2 1 1 2 1 2 2 1 1 1 1 1 1 2 2 1 1 2 1 2 1 1 1 2 1 2 2 2 1 2 1 1 2
##  [71] 2 1 1 1 1 1 2 2 1 1 2 1 1 1 1 1 2 1 2 1 2 2 1 1 2 1 2 2 1 1 1 1 2 1 1
## [106] 2 1 2 1 1 1 1 2 1 2 2 2 1 1 1 1 1 1 2 2 2 2 1 1 2 1 2 2 2 2 2 2 1 1 2
## [141] 1 2 1 1 1 2 2 1 2 1 2 1 2 1 1 2 2 1 2 1 2 1 2 2 1 1 2 1 2 1 2 1 2 1 1
## [176] 2 2 2 2 2 1 1 1 1 2 2 1 2 1 1 2 2 1 1 2 1 1 2 1 1 1 2 1 1 2 2 1 1 1 2
## [211] 1 2 1 1 2 1 1 1 1 1 1 1 1 1 2 2 2 1 1 1 1 1 1 2 2 1 2 2 1 1 2 2 2 2 1
## [246] 2 2 1 1 2 1 2 1 1 2 2 1 1 1 1 2 2 2 2 2 1 1 2 2 2 1 1 1 2 2 1 1 1 1 2
```

```
## [281] 1 2 1 1 1 1 1 2 1 1 2 2 1 1 2 1 2 1 1 1 1 1 2 2 1 2 1 1 1 1 1 1 2 1 1
## [316] 2 1 1 1 2 1 2 1 1 1 2 1 1 1 1 1 2 2 2 1 1
```

In the display of the index, it is seen that after the random sampling, approximately 65% of the dataset is the training set and approximately 35% of the dataset is the testing set.

After creating the index, the specified index values need to be assigned to either the training or testing set. As seen in the chunk below, only columns 2, 3, 6, 7, and 8 are being used for the knn classification. This is because columns 4 and 5 (consensus sequence score and charge on predicted lipoproteins respectively) are boolean values, and therefore have a negligible contribution to the overall knn classification.

```
#We are omitting the intrinsically boolean values as well as the factor values
#Euclidean distance cannot be calculated with such values in place
dataset.training <- dataset[ind==1, c(2:3, 6:8, 10)]
dataset.test <- dataset[ind==2, c(2:3, 6:8, 10)]

summary(dataset.test)
```

```
##       mcg              gvh              aac              alm1
##  Min.   :0.0000   Min.   :0.1600   Min.   :0.2200   Min.   :0.0300
##  1st Qu.:0.3400   1st Qu.:0.4000   1st Qu.:0.4500   1st Qu.:0.3300
##  Median :0.5000   Median :0.4700   Median :0.5100   Median :0.4600
##  Mean   :0.4996   Mean   :0.4923   Mean   :0.5161   Mean   :0.5093
##  3rd Qu.:0.6600   3rd Qu.:0.5600   3rd Qu.:0.5800   3rd Qu.:0.7100
##  Max.   :0.8600   Max.   :0.9000   Max.   :0.8800   Max.   :0.9100
##       alm2        variable_class
##  Min.   :0.0100   Min.   :1.000
##  1st Qu.:0.3500   1st Qu.:1.000
##  Median :0.4400   Median :2.000
##  Mean   :0.5164   Mean   :3.007
##  3rd Qu.:0.7400   3rd Qu.:5.000
##  Max.   :0.9200   Max.   :8.000
```

```
summary(dataset.training)
```

```
##       mcg              gvh              aac              alm1
##  Min.   :0.0000   Min.   :0.2300   Min.   :0.000   Min.   :0.0500
##  1st Qu.:0.3400   1st Qu.:0.4000   1st Qu.:0.410   1st Qu.:0.3250
##  Median :0.5100   Median :0.4800   Median :0.480   Median :0.4500
##  Mean   :0.5004   Mean   :0.5053   Mean   :0.489   Mean   :0.4939
##  3rd Qu.:0.6700   3rd Qu.:0.5850   3rd Qu.:0.560   3rd Qu.:0.7000
##  Max.   :0.8900   Max.   :1.0000   Max.   :0.860   Max.   :1.0000
##       alm2        variable_class
##  Min.   :0.0000   Min.   :1.000
##  1st Qu.:0.3500   1st Qu.:1.000
##  Median :0.4200   Median :2.000
##  Mean   :0.4882   Mean   :3.241
##  3rd Qu.:0.6800   3rd Qu.:5.000
##  Max.   :0.9900   Max.   :8.000
```

```
dataset.trainLabels <- na.omit(dataset[ind==1, 10])
dataset.testLabels <- na.omit(dataset[ind==2, 10])
```

After full preparation of the dataset, the knn() function should be used to conduct the knn classification. The results of the knn will be read to the dataset data_pred. Two more datasets will be created, merge and final_data. The "merge" dataset will simply be a data frame containing the observed values (dataset.testLabels) and the predicted values (result of knn classification, data_pred). Names will be applied to the final dataset with the observed values being named as "Observed Class" and the predicted values being named as "Predicted Class".

```
data_pred <- knn(train = dataset.training, test = dataset.test, cl = dataset.trainLabels, k=1)
data_pred
```

```
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [36] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2
##  [71] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 5 5 5 5 5 5 5 5 5 5 5
## [106] 5 5 5 5 6 6 6 6 6 6 6 6 6 6 7 7 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
## Levels: 1 2 3 4 5 6 7 8
```

```
merge <- data.frame(dataset.testLabels, data_pred)
dim(merge)
```

```
## [1] 137   2
```

```
names <- colnames(dataset.test)
names
```

```
## [1] "mcg"            "gvh"            "aac"            "alm1"
## [5] "alm2"           "variable_class"
```

```
final_data <- cbind(dataset.test, merge)
names(final_data) <- c(names, "Observed Class", "Predicted Class")
```

CONCLUSION

Crosstable Visualization

To visualize the results of the knn compared to the observed results in the dataset, we will use a cross table:

```
CrossTable(x = dataset.testLabels, y = data_pred, prop.chisq=FALSE, format="SPSS")
```

```
##
##    Cell Contents
## |-------------------------|
## |                   Count |
## |             Row Percent |
## |          Column Percent |
## |           Total Percent |
## |-------------------------|
##
## Total Observations in Table:  137
##
##                    | data_pred
## dataset.testLabels |         1  |         2  |         5  |         6  |         7  |         8  | Row Total  |
## -------------------|------------|------------|------------|------------|------------|------------|------------|
##                  1 |        59  |         0  |         0  |         0  |         0  |         0  |        59  |
##                    |   100.000% |     0.000% |     0.000% |     0.000% |     0.000% |     0.000% |    43.066% |
##                    |   100.000% |     0.000% |     0.000% |     0.000% |     0.000% |     0.000% |            |
##                    |    43.066% |     0.000% |     0.000% |     0.000% |     0.000% |     0.000% |            |
## -------------------|------------|------------|------------|------------|------------|------------|------------|
##                  2 |         0  |        33  |         0  |         0  |         0  |         0  |        33  |
##                    |     0.000% |   100.000% |     0.000% |     0.000% |     0.000% |     0.000% |    24.088% |
##                    |     0.000% |   100.000% |     0.000% |     0.000% |     0.000% |     0.000% |            |
##                    |     0.000% |    24.088% |     0.000% |     0.000% |     0.000% |     0.000% |            |
## -------------------|------------|------------|------------|------------|------------|------------|------------|
##                  5 |         0  |         0  |        17  |         0  |         0  |         0  |        17  |
```

```
##                    |    0.000% |    0.000% |  100.000% |    0.000% |    0.000% |    0.000% |   12.409% |
##                    |    0.000% |    0.000% |  100.000% |    0.000% |    0.000% |    0.000% |           |
##                    |    0.000% |    0.000% |   12.409% |    0.000% |    0.000% |    0.000% |           |
## -------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
##                  6 |        0  |        0  |        0  |       10  |        0  |        0  |       10  |
##                    |    0.000% |    0.000% |    0.000% |  100.000% |    0.000% |    0.000% |    7.299% |
##                    |    0.000% |    0.000% |    0.000% |  100.000% |    0.000% |    0.000% |           |
##                    |    0.000% |    0.000% |    0.000% |    7.299% |    0.000% |    0.000% |           |
## -------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
##                  7 |        0  |        0  |        0  |        0  |        2  |        0  |        2  |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |  100.000% |    0.000% |    1.460% |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |  100.000% |    0.000% |           |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |    1.460% |    0.000% |           |
## -------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
##                  8 |        0  |        0  |        0  |        0  |        0  |       16  |       16  |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |    0.000% |  100.000% |   11.679% |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |    0.000% |  100.000% |           |
##                    |    0.000% |    0.000% |    0.000% |    0.000% |    0.000% |   11.679% |           |
## -------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
##       Column Total |       59  |       33  |       17  |       10  |        2  |       16  |      137  |
##                    |   43.066% |   24.088% |   12.409% |    7.299% |    1.460% |   11.679% |           |
## -------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
##
##
```

In this cross table, the columns are the predicted values from the knn prediction algorithm and the rows are the observed values from the dataset. The cross table indicates that out of 59 observed data points of value "1", 59 were correctly predicted; out of 33 observed data points of value "2", 33 were correctly predicted; out of 17 observed data points of value "5", 17 were correctly predicted; out of 10 observed data points with value "6", 10 were correctly predicted; out of 2 observed data points of value "7", 2 were correctly predicted; and out of 16 observed data points of value "8", 16 were correctly predicted.

The accuracy for all of the predicted cells in the cross table is "1.000" indicating that the knn classification algorithm predicted the protein localization sites of the testing set with 100% accuracy.

REFERENCES

Asuncion, Arthur, and David Newman. 2007. "UCI Machine Learning Repository."

Deng, Zhenyun, Xiaoshu Zhu, Debo Cheng, Ming Zong, and Shichao Zhang. 2016. "Efficient kNN Classification Algorithm for Big Data." *Neurocomputing* 195: 143–48.

Horton, Paul, Keun-Joon Park, Takeshi Obayashi, Naoya Fujita, Hajime Harada, CJ Adams-Collier, and Kenta Nakai. 2007. "WoLF Psort: Protein Localization Predictor." *Nucleic Acids Research* 35 (suppl_2): W585–W587.

Min, Renqiang, David A Stanley, Zineng Yuan, Anthony Bonner, and Zhaolei Zhang. 2009. "A Deep Non-Linear Feature Mapping for Large-Margin Knn Classification." In *2009 Ninth Ieee International Conference on Data Mining*, 357–66. IEEE.

Nakai, Kenta, and Minoru Kanehisa. 1991. "Expert System for Predicting Protein Localization Sites in Gram-Negative Bacteria." *Proteins: Structure, Function, and Bioinformatics* 11 (2): 95–110.

———. 1992. "A Knowledge Base for Predicting Protein Localization Sites in Eukaryotic Cells." *Genomics* 14 (4): 897–911.