HLA Class prediction

Hem R. Gurung, Ph.D.

This project builds a simple binomial logistic regression model to predict whether a given peptide belongs to class I or class II based on its mass, length, m/z, and retention time values.

```
setwd("C:/Users/HGURUNG1/Desktop/files")

# read the dataset
dataset <- read.csv("all_pep_final.csv", header = T)

# Check dimension
dim(dataset)
```

```
## [1] 5856   14
```

```
# Check column names
names(dataset)
```

```
##  [1] "Peptide"     "X.10lgP"     "Mass"        "Length"      "ppm"        
##  [6] "m.z"         "RT"          "Area"        "Fraction"    "Scan"       
## [11] "Source.File" "X.Spec"      "Accession"   "HLA_Class"
```

```
# Glance at first 6 rows
head(dataset)
```

```
##               Peptide X.10lgP     Mass Length  ppm      m.z    RT Area
## 1    HSSTFDAGAGIALNDH   86.23 1611.728     16 -2.6 806.8690 34.57   NA
## 2     DEFKVETSNKVLDYD   82.52 1800.842     15  0.2 601.2880 40.02   NA
## 3     AGKYVPAIAHLIHSL   82.00 1588.909     15  0.3 530.6437 51.94   NA
## 4   FSDEFKVETSNKVLDYD   81.68 2034.942     17 -1.8 679.3201 43.68   NA
## 5 VDKVIQAQTAFSANPANPA   80.23 1940.995     19  0.9 648.0063 38.24   NA
## 6      LFLQFGAQGSPFLK   79.74 1551.845     14  2.7 518.2903 59.05   NA
##   Fraction Scan
## 1        8 5912
## 2        9 7017
## 3       15 5115
## 4       11 6201
## 5        9 6554
## 6        9 9862
##                                                                           Source.File
## 1 18-10-26-iRT THP1 Mac No pulse L243 FXN 19-iRT THP1 Mac No pulse L243 FXN 19.mzXML
## 2 18-10-26-iRT THP1 Mac No pulse L243 FXN 20-iRT THP1 Mac No pulse L243 FXN 20.mzXML
## 3 18-10-26-iRT THP1 Mac No pulse L243 FXN 26-iRT THP1 Mac No pulse L243 FXN 26.mzXML
## 4 18-10-26-iRT THP1 Mac No pulse L243 FXN 22-iRT THP1 Mac No pulse L243 FXN 22.mzXML
## 5 18-10-26-iRT THP1 Mac No pulse L243 FXN 20-iRT THP1 Mac No pulse L243 FXN 20.mzXML
## 6 18-10-26-iRT THP1 Mac No pulse L243 FXN 20-iRT THP1 Mac No pulse L243 FXN 20.mzXML
##   X.Spec                               Accession HLA_Class
## 1     16     P04406-2|G3P_HUMAN:P04406|G3P_HUMAN  Class II
## 2      4                      O14672|ADA10_HUMAN  Class II
## 3      3                       Q9H3G5|CPVL_HUMAN  Class II
## 4      9                      O14672|ADA10_HUMAN  Class II
## 5      5 O00560-2|SDCB1_HUMAN:O00560|SDCB1_HUMAN  Class II
## 6    279            Biognosys|iRT-Kit_peptide_11  Class II
```

```
# remove unwanted columns
drop <- c("Peptide", "X.10lgP", "ppm", "Area", "Fraction", "Scan", "Source.File", "X.Spec", "Accession"
dataset <- dataset[, !names(dataset) %in% drop]

# Check if there is any NA in the dataset
sapply(dataset, function(x) sum(is.na(x)))
```

```
##      Mass    Length       m.z        RT HLA_Class
##         0         0         0         0         0
```

```
head(dataset)
```

```
##       Mass Length      m.z    RT HLA_Class
## 1 1611.728     16 806.8690 34.57  Class II
## 2 1800.842     15 601.2880 40.02  Class II
## 3 1588.909     15 530.6437 51.94  Class II
## 4 2034.942     17 679.3201 43.68  Class II
## 5 1940.995     19 648.0063 38.24  Class II
## 6 1551.845     14 518.2903 59.05  Class II
```

```
# Rename Class I as 1 and class II as 0
dataset$HLA_Class <- ifelse(dataset$HLA_Class == "Class I", 1, 0)

# Check class bias
table(dataset$HLA_Class) # class bias with more proportion in class I data
```

```
##
##    0    1
## 1843 4013
```

```
# treat class bias and split dataset into train and validate sets
all_ones <- dataset[which(dataset$HLA_Class == 1), ]
all_zeros <- dataset[which(dataset$HLA_Class == 0), ]
dim(all_ones)
```

```
## [1] 4013    5
```

```
dim(all_zeros)
```

```
## [1] 1843    5
```

```
set.seed(123)
training_indices_ones <- sample(1:nrow(all_ones), 0.8*nrow(all_zeros))
training_indices_zeros <- sample(1:nrow(all_zeros), 0.8*nrow(all_zeros))

training_ones <- all_ones[training_indices_ones, ]
training_zeros <- all_zeros[training_indices_zeros, ]
train <- rbind(training_ones, training_zeros)
dim(train) # rows doubled
```

```
## [1] 2948    5
```

```
# Create validation dataset
validate_ones <- all_ones[-training_indices_ones, ]
validate_zeros <- all_zeros[-training_indices_zeros, ]
validate <- rbind(validate_ones, validate_zeros)
dim(validate)
```

```
## [1] 2908    5
```

```r
# Fit a binomial regression model
model <- glm(HLA_Class ~., family = binomial(link = 'logit'), data = train)

# Print summary of the model
summary(model)
```

```
##
## Call:
## glm(formula = HLA_Class ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0180  -0.4215   0.0810   0.6225   4.1588
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  8.536e+00  3.436e-01  24.845  < 2e-16 ***
## Mass        -4.807e-04  5.824e-04  -0.825    0.409
## Length      -6.180e-01  6.409e-02  -9.642  < 2e-16 ***
## m.z         -5.111e-05  7.042e-04  -0.073    0.942
## RT          -2.268e-02  5.641e-03  -4.020 5.81e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4086.8  on 2947  degrees of freedom
## Residual deviance: 2241.3  on 2943  degrees of freedom
## AIC: 2251.3
##
## Number of Fisher Scoring iterations: 6
```

```r
# Run anova to analyze the table of deviance
anova(model, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: HLA_Class
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    2947     4086.8
## Mass    1  1716.25      2946     2370.5 < 2.2e-16 ***
## Length  1   109.51      2945     2261.0 < 2.2e-16 ***
## m.z     1     3.60      2944     2257.4   0.05769 .
## RT      1    16.14      2943     2241.3 5.898e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# assess the predictive ability of the model
fitted.results <- predict(model, newdata = validate, type = "response")
fitted.results <- ifelse(fitted.results >0.5, 1, 0)
misClassificationError <- mean(fitted.results != validate$HLA_Class)
print(paste("Accuracy", 1-misClassificationError))
```

```
## [1] "Accuracy 0.934662998624484"
```

```r
# Interpolate the classification of peptides
validate$predicted_HLA_Class <- fitted.results

# Check first 6 rows
head(validate)
```

```
##          Mass Length      m.z    RT HLA_Class predicted_HLA_Class
## 1033 1386.810     14 463.2775 27.83         1                   0
## 1037 1250.725     11 417.9169 23.82         1                   1
## 1038 1129.671     10 565.8441 28.99         1                   1
## 1039 1006.606      9 504.3116 45.89         1                   1
## 1041 1094.645     10 365.8913 22.95         1                   1
## 1042 1140.687     12 571.3495 38.56         1                   0
```

```r
# plot ROC curve
library(ROCR)
```

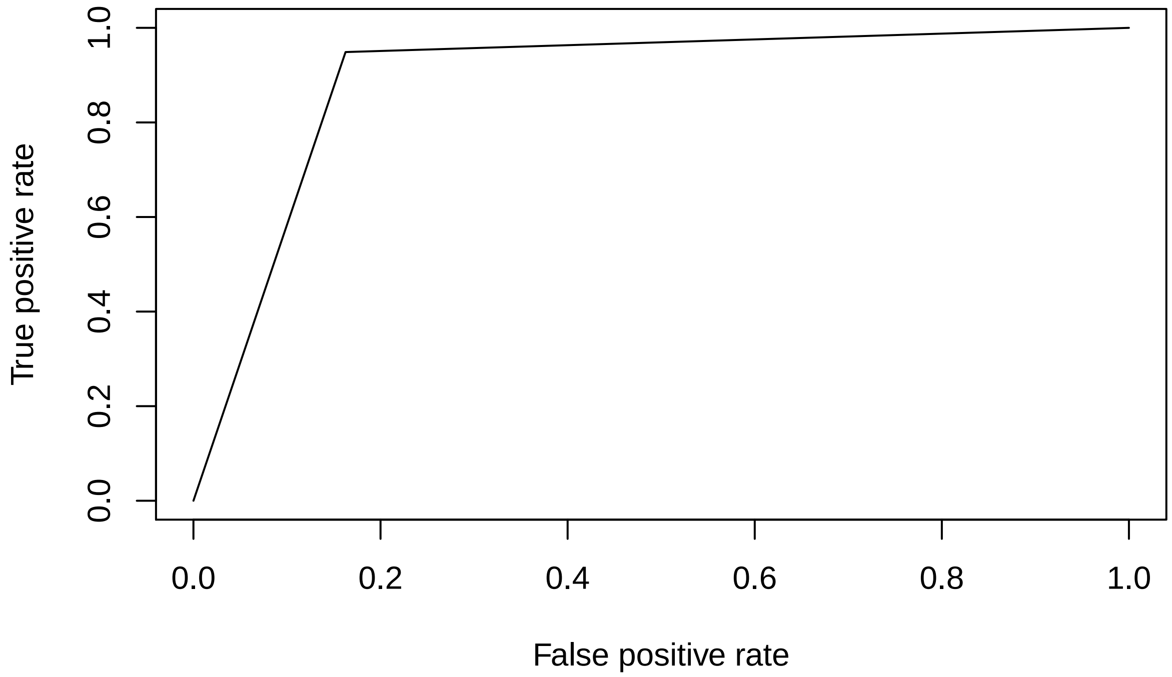
```
## Warning: package 'ROCR' was built under R version 3.4.4
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```r
pr <- prediction(fitted.results, validate$HLA_Class)
prf <- performance(pr, measure = "tpr", x.measure = "fpr")
plot(prf)
```



```
# calculate AUC
AUC <- performance(pr, measure = "auc")
AUC <- AUC@y.values[[1]]
AUC
```

```
## [1] 0.8930986
```