Exercise 3 - Team 3

2024-01-29

Let's start by loading our dataset from the last assignment.

```
data_path =
"/Users/sheidamajidi/Desktop/Winter2024/COURSES/ORGB671/Exercise3/app_data.fe
ather"
options(repos = c(CRAN = "https://cran.rstudio.com"))
install.packages("arrow")
```

```
## Installing package into '/Users/sheidamajidi/Library/R/arm64/4.3/library'
## (as 'lib' is unspecified)

##
## The downloaded binary packages are in
##
/var/folders/zh/7hbjyl3x1y953yvj5t_7dbbw0000gn/T//Rtmpz8tLIl/downloaded_packa
ges
```

```
library(arrow)
```

```
## Warning: package 'arrow' was built under R version 4.3.1

##
## Attaching package: 'arrow'

## The following object is masked from 'package:utils':
##
##     timestamp
```

```
applications <- read_feather(data_path)
```

Now that we have our data, we can run a logistic regression on examiner mobility with AU indicator as our target variable, also known as our y.

We need to add the column for AU_move_indicator from last session. Since we're having trouble running the entire code, we've chosen to rewrite our own pre-processing here to create a lighter code file. However, the code to create the AU indicator creates a column that is holds true or false. As such, we need to turn the true/false into 1/0.

```
## Installing package into '/Users/sheidamajidi/Library/R/arm64/4.3/library'
## (as 'lib' is unspecified)

##
## The downloaded binary packages are in
##
/var/folders/zh/7hbjyl3x1y953yvj5t_7dbbw0000gn/T//Rtmpz8tLIl/downloaded_packa
ges
```

```
## Warning: package 'dplyr' was built under R version 4.3.1

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

We want to ensure that there's no null values in the variables that we're going to use for our analysis. We can use median or mode imputation to simplify this process for the sake of getting a result for our prediction, but the best scenario would be to have used a processed dataset from assignment 2.

```r
# Checking for null values in each categorical variable
sum(is.na(applications$disposal_type))

## [1] 0

sum(is.na(applications$gender))

## [1] 303859

sum(is.na(applications$race))

## [1] 0
```

Since we only have missing values for gender, we should perform imputation on that variable. However, if we use mode imputation on gender, all the remaining null values will be filled with either one or the other gender that is more prominent in the dataset, which can further skew the results. As such, we will try to use the code from assignment 2 to use the first name as a tell for gender.

The code above from the second assignment cannot be run, since it crashes our R studios when reaching the left join code.

As such, we will use mode even though we know it will skew our data.

```r
# Function to calculate mode, handling NA values
getMode <- function(v) {
   # Removing NA values
   v <- na.omit(v)

   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

# Mode imputation for 'gender'
```

```
if(sum(is.na(applications$gender.x)) > 0) {
  mode_gender <- getMode(applications$gender.x)
  applications$gender.x[is.na(applications$gender.x)] <- mode_gender
}
```

Now we can re-check to make sure there’s no null values left.

```
sum(is.na(applications$gender.x))

## [1] 0
```

Given that we have our binary target variable and that our data is ready, we can run a multiple logistic regression to be able to predict if someone will move art units or not.

```
set.seed(123)  # for reproducibility
applications_subset <- applications[sample(nrow(applications), 10000), ]
mlogit <- glm(AU_move_indicator ~ filing_date + examiner_art_unit +
uspc_class + disposal_type + race + tenure_days,
              data = applications_subset,
              family = "binomial")

summary(mlogit)

##
## Call:
## glm(formula = AU_move_indicator ~ filing_date + examiner_art_unit +
##     uspc_class + disposal_type + race + tenure_days, family = "binomial",
##     data = applications_subset)
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)         2.906e+00  1.381e+00   2.104 0.035350 *
## filing_date        -1.397e-04  1.856e-05  -7.530 5.06e-14 ***
## examiner_art_unit   1.159e-03  5.276e-04   2.197 0.028047 *
## uspc_class015      -2.010e+01  6.406e+02  -0.031 0.974971
## uspc_class023       1.353e+01  1.694e+03   0.008 0.993626
## uspc_class029       1.364e+01  8.890e+02   0.015 0.987762
## uspc_class034       1.357e+01  2.400e+03   0.006 0.995488
## uspc_class043       1.330e+01  2.400e+03   0.006 0.995579
## uspc_class044      -8.355e-02  1.450e+00  -0.058 0.954044
## uspc_class048      -1.491e+00  1.285e+00  -1.160 0.246008
## uspc_class051       1.366e+01  8.357e+02   0.016 0.986956
## uspc_class052       1.306e+01  2.400e+03   0.005 0.995658
## uspc_class055       2.081e-01  1.445e+00   0.144 0.885457
## uspc_class056      -2.018e+01  2.400e+03  -0.008 0.993289
## uspc_class065      -1.261e+00  1.197e+00  -1.054 0.292075
## uspc_class068      -2.054e+00  1.223e+00  -1.680 0.092974 .
## uspc_class071       1.369e+01  9.014e+02   0.015 0.987880
## uspc_class073       1.333e+01  1.686e+03   0.008 0.993695
## uspc_class074      -2.058e+01  2.400e+03  -0.009 0.993158
## uspc_class075      -1.031e+00  1.273e+00  -0.810 0.418063
## uspc_class082      1.257e+01  2.400e+03   0.005 0.995822    
## uspc_class095     -1.647e+00  1.136e+00  -1.450 0.147022    
## uspc_class096     -1.114e+00  1.195e+00  -0.933 0.351006    
## uspc_class099     -2.010e+01  9.791e+02  -0.021 0.983625    
## uspc_class106      8.099e-01  1.437e+00   0.563 0.573115    
## uspc_class111     -1.990e+01  2.400e+03  -0.008 0.993382    
## uspc_class117     -1.643e+00  1.122e+00  -1.464 0.143084    
## uspc_class118     -1.375e+00  1.092e+00  -1.260 0.207847    
## uspc_class127      1.396e+01  1.380e+03   0.010 0.991930    
## uspc_class131      1.352e+01  7.481e+02   0.018 0.985581    
## uspc_class134     -2.362e+00  1.055e+00  -2.240 0.025107 *  
## uspc_class136     -1.895e+00  1.054e+00  -1.798 0.072113 .  
## uspc_class137      1.323e+01  2.400e+03   0.006 0.995601    
## uspc_class148     -4.632e-01  1.147e+00  -0.404 0.686377    
## uspc_class149      1.309e+01  1.381e+03   0.009 0.992439    
## uspc_class152      1.351e+01  5.431e+02   0.025 0.980156    
## uspc_class156     -7.217e-01  1.063e+00  -0.679 0.496976    
## uspc_class162     -9.549e-01  1.268e+00  -0.753 0.451552    
## uspc_class164      1.336e+01  5.295e+02   0.025 0.979875    
## uspc_class174     -1.923e+01  2.400e+03  -0.008 0.993606    
## uspc_class180     -2.070e+01  2.400e+03  -0.009 0.993117    
## uspc_class196      1.343e+01  2.400e+03   0.006 0.995534    
## uspc_class201      1.324e+01  1.383e+03   0.010 0.992359    
## uspc_class202      1.333e+01  2.400e+03   0.006 0.995568    
## uspc_class203     -8.923e-01  1.479e+00  -0.603 0.546380    
## uspc_class204     -8.235e-01  1.081e+00  -0.762 0.446151    
## uspc_class205     -4.858e-01  1.187e+00  -0.409 0.682346    
## uspc_class206     -1.946e+01  2.400e+03  -0.008 0.993530    
## uspc_class208     -1.220e+00  1.196e+00  -1.020 0.307705    
## uspc_class209      1.333e+01  1.686e+03   0.008 0.993692    
## uspc_class210     -1.540e+00  1.042e+00  -1.478 0.139436    
## uspc_class216     -2.317e+00  1.077e+00  -2.150 0.031535 *  
## uspc_class219      1.305e+01  5.500e+02   0.024 0.981066    
## uspc_class222      1.329e+01  1.380e+03   0.010 0.992315    
## uspc_class228      1.333e+01  4.140e+02   0.032 0.974320    
## uspc_class249     -3.217e+00  1.745e+00  -1.843 0.065267 .  
## uspc_class252     -1.282e+00  1.068e+00  -1.200 0.229993    
## uspc_class257      1.358e+01  1.696e+03   0.008 0.993613    
## uspc_class261      1.353e+01  7.915e+02   0.017 0.986358    
## uspc_class264     -4.597e-01  1.073e+00  -0.428 0.668353    
## uspc_class266      1.355e+01  1.058e+03   0.013 0.989784    
## uspc_class300     -2.015e+01  1.695e+03  -0.012 0.990519    
## uspc_class307      1.312e+01  2.400e+03   0.005 0.995637    
## uspc_class313     -2.795e+00  1.595e+00  -1.752 0.079706 .  
## uspc_class324      1.256e+01  1.697e+03   0.007 0.994095    
## uspc_class336      1.320e+01  2.400e+03   0.006 0.995610    
## uspc_class340     -4.491e+00  1.624e+00  -2.766 0.005682 ** 
## uspc_class343     -2.050e+01  2.400e+03  -0.009 0.993185    
## uspc_class345     -2.852e+00  1.170e+00  -2.437 0.014811 *  
## uspc_class348     -4.608e+00  1.104e+00  -4.175 2.98e-05 ***
## uspc_class351      1.346e+01  2.400e+03   0.006 0.995525
## uspc_class359      1.344e+01  2.400e+03   0.006 0.995532
## uspc_class360      1.313e+01  2.400e+03   0.005 0.995634
## uspc_class361     -3.750e+00  1.765e+00  -2.125 0.033555 *
## uspc_class362      1.322e+01  2.400e+03   0.006 0.995605
## uspc_class366     -1.552e+00  1.137e+00  -1.365 0.172313
## uspc_class370     -2.513e+00  1.091e+00  -2.304 0.021231 *
## uspc_class375     -4.171e+00  1.118e+00  -3.730 0.000191 ***
## uspc_class380     -2.528e+00  1.128e+00  -2.241 0.025022 *
## uspc_class382     -2.850e+00  1.767e+00  -1.613 0.106842
## uspc_class386     -5.801e+00  1.189e+00  -4.881 1.06e-06 ***
## uspc_class399      1.316e+01  1.694e+03   0.008 0.993800
## uspc_class403      1.291e+01  2.400e+03   0.005 0.995707
## uspc_class419     -1.755e+00  1.221e+00  -1.438 0.150553
## uspc_class420      1.344e+01  8.425e+02   0.016 0.987277
## uspc_class422     -3.429e-01  1.085e+00  -0.316 0.751971
## uspc_class423     -1.061e+00  1.081e+00  -0.981 0.326525
## uspc_class424     -3.161e+00  1.027e+00  -3.079 0.002079 **
## uspc_class425     -1.128e+00  1.111e+00  -1.015 0.309988
## uspc_class426     -4.749e-01  1.074e+00  -0.442 0.658263
## uspc_class427     -1.476e+00  1.045e+00  -1.412 0.157818
## uspc_class428     -1.535e+00  1.029e+00  -1.491 0.135894
## uspc_class429     -1.745e+00  1.034e+00  -1.688 0.091357 .
## uspc_class430      4.869e-01  1.140e+00   0.427 0.669416
## uspc_class433      1.331e+01  2.400e+03   0.006 0.995573
## uspc_class435     -2.947e+00  1.026e+00  -2.872 0.004084 **
## uspc_class436     -9.642e-01  1.076e+00  -0.896 0.370342
## uspc_class438     -1.314e+00  1.131e+00  -1.162 0.245097
## uspc_class439      1.283e+01  2.400e+03   0.005 0.995734
## uspc_class442     -7.393e-01  1.190e+00  -0.621 0.534410
## uspc_class455      1.278e+01  1.195e+03   0.011 0.991468
## uspc_class473      1.302e+01  2.400e+03   0.005 0.995669
## uspc_class474     -1.973e+01  2.400e+03  -0.008 0.993440
## uspc_class494      1.362e+01  1.648e+03   0.008 0.993406
## uspc_class501      1.339e+01  5.960e+02   0.022 0.982072
## uspc_class502     -1.047e+00  1.098e+00  -0.953 0.340643
## uspc_class503      1.321e+01  1.067e+03   0.012 0.990123
## uspc_class504     -2.933e+00  1.111e+00  -2.639 0.008315 **
## uspc_class505      1.327e+01  1.378e+03   0.010 0.992314
## uspc_class506     -2.162e+00  1.128e+00  -1.917 0.055225 .
## uspc_class507     -1.519e+00  1.212e+00  -1.253 0.210157
## uspc_class508     -5.362e-01  1.261e+00  -0.425 0.670676
## uspc_class510      5.027e-02  1.252e+00   0.040 0.967975
## uspc_class512      1.308e+01  1.199e+03   0.011 0.991296
## uspc_class514     -2.358e+00  1.027e+00  -2.297 0.021647 *
## uspc_class516      1.327e+01  1.385e+03   0.010 0.992355
## uspc_class518      1.350e+01  1.373e+03   0.010 0.992158
## uspc_class521     -1.886e+00  1.143e+00  -1.650 0.098926 .
## uspc_class522     -1.415e+00  1.275e+00  -1.110 0.266998
## uspc_class523     -1.064e+00  1.113e+00  -0.957 0.338711
## uspc_class524     -6.681e-01  1.065e+00  -0.627 0.530570
## uspc_class525     -8.660e-01  1.109e+00  -0.781 0.434738
## uspc_class526     -7.840e-01  1.149e+00  -0.683 0.494877
## uspc_class528     -6.990e-02  1.181e+00  -0.059 0.952803
## uspc_class530     -2.984e+00  1.059e+00  -2.818 0.004840 **
## uspc_class534     -3.970e+00  1.600e+00  -2.481 0.013110 *
## uspc_class536     -2.142e+00  1.058e+00  -2.024 0.042977 *
## uspc_class540     -1.886e+00  1.178e+00  -1.601 0.109281
## uspc_class544     -3.432e+00  1.071e+00  -3.205 0.001352 **
## uspc_class546     -3.470e+00  1.072e+00  -3.236 0.001211 **
## uspc_class548     -3.606e+00  1.064e+00  -3.390 0.000700 ***
## uspc_class549     -2.370e+00  1.113e+00  -2.130 0.033171 *
## uspc_class552      1.379e+01  7.913e+02   0.017 0.986098
## uspc_class554      1.352e+01  6.147e+02   0.022 0.982450
## uspc_class556      1.354e+01  6.142e+02   0.022 0.982416
## uspc_class558     -3.266e+00  1.366e+00  -2.391 0.016791 *
## uspc_class560     -1.943e+00  1.145e+00  -1.697 0.089621 .
## uspc_class562     -1.412e+00  1.204e+00  -1.172 0.241242
## uspc_class564     -7.018e-01  1.266e+00  -0.554 0.579432
## uspc_class568     -1.186e+00  1.201e+00  -0.988 0.323332
## uspc_class570     -1.063e+00  1.475e+00  -0.720 0.471445
## uspc_class585     -1.107e+00  1.193e+00  -0.928 0.353276
## uspc_class588      1.356e+01  1.697e+03   0.008 0.993623
## uspc_class600      1.352e+01  1.072e+03   0.013 0.989933
## uspc_class604      1.337e+01  1.695e+03   0.008 0.993706
## uspc_class606      1.364e+01  2.400e+03   0.006 0.995464
## uspc_class700     -1.961e+00  1.068e+00  -1.835 0.066432 .
## uspc_class701     -3.362e+00  1.787e+00  -1.881 0.059936 .
## uspc_class702     -3.839e+00  1.085e+00  -3.537 0.000405 ***
## uspc_class703     -3.982e+00  1.066e+00  -3.734 0.000188 ***
## uspc_class704      1.324e+01  2.400e+03   0.006 0.995596
## uspc_class705      1.271e+01  2.400e+03   0.005 0.995773
## uspc_class706     -1.863e+00  1.092e+00  -1.706 0.087961 .
## uspc_class707     -2.383e+00  1.050e+00  -2.270 0.023216 *
## uspc_class708      1.296e+01  4.490e+02   0.029 0.976982
## uspc_class709     -2.481e+00  1.083e+00  -2.289 0.022050 *
## uspc_class710     -2.659e+00  1.061e+00  -2.506 0.012221 *
## uspc_class711     -1.160e+00  1.071e+00  -1.084 0.278543
## uspc_class712     -3.341e+00  1.084e+00  -3.083 0.002047 **
## uspc_class713     -2.510e+00  1.069e+00  -2.348 0.018891 *
## uspc_class714     -2.465e+00  1.053e+00  -2.342 0.019185 *
## uspc_class715     -3.153e+00  1.053e+00  -2.994 0.002757 **
## uspc_class717     -2.700e+00  1.066e+00  -2.533 0.011298 *
## uspc_class718     -3.454e+00  1.077e+00  -3.206 0.001345 **
## uspc_class719     -3.273e+00  1.114e+00  -2.937 0.003312 **
## uspc_class725     -4.768e+00  1.100e+00  -4.333 1.47e-05 ***
## uspc_class726     -2.599e+00  1.094e+00  -2.376 0.017495 *
## uspc_class800     -1.057e+00  1.054e+00  -1.003 0.315757
## disposal_typeISS   2.073e-01  5.919e-02   3.502 0.000461 ***
## disposal_typePEND -1.575e-01  8.651e-02  -1.821 0.068630 .
## raceblack          2.692e-01  1.353e-01   1.990 0.046628 *
## raceHispanic      -1.019e-01  1.472e-01  -0.692 0.488769
## raceother          1.555e+01  9.008e+02   0.017 0.986224
## racewhite          6.514e-02  5.750e-02   1.133 0.257242
## tenure_days        1.072e-06  4.494e-07   2.385 0.017057 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11694.9  on 9999  degrees of freedom
## Residual deviance:  9675.2  on 9825  degrees of freedom
## AIC: 10025
##
## Number of Fisher Scoring iterations: 15
```

```
# Making predictions

# Ensuring 'uspc_class' is numeric in the training dataset
applications_subset$uspc_class <-
as.numeric(as.character(applications_subset$uspc_class))

# Refitting the model with 'uspc_class' as numeric
mlogit <- glm(AU_move_indicator ~ filing_date + examiner_art_unit +
uspc_class + disposal_type + race + tenure_days,
              data = applications_subset,
              family = "binomial")

# Creating a new data frame for prediction with 'uspc_class' as numeric
Prob_1 <- data.frame(
  filing_date = as.Date("2000-01-26"),
  examiner_art_unit = 1734,
  uspc_class = 5156,  # Keep uspc_class as numeric
  disposal_type = factor("ISS", levels =
levels(applications_subset$disposal_type)),
  race = factor("Asian", levels = levels(applications_subset$race)),
  tenure_days = 5600
)

# Making predictions using the logistic regression model
predicted_probabilities <- predict(mlogit, newdata = Prob_1, type =
"response")

# Viewing the predicted probabilities
predicted_probabilities
```

```
##  1
## NA
```

We can also use train/test split prior to have a validation set. This allows us to better evaluate our model's predictions.

```
install.packages("caTools")

## Installing package into '/Users/sheidamajidi/Library/R/arm64/4.3/library'
## (as 'lib' is unspecified)

##
## The downloaded binary packages are in
##
/var/folders/zh/7hbjyl3x1y953yvj5t_7dbbw0000gn/T//Rtmpz8tLIl/downloaded_packa
ges

install.packages("pROC")

## Installing package into '/Users/sheidamajidi/Library/R/arm64/4.3/library'
## (as 'lib' is unspecified)

##
## The downloaded binary packages are in
##
/var/folders/zh/7hbjyl3x1y953yvj5t_7dbbw0000gn/T//Rtmpz8tLIl/downloaded_packa
ges

library(caTools)
library(pROC)

## Warning: package 'pROC' was built under R version 4.3.1

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

# Splitting the data into training (70%) and test (30%) sets
set.seed(123) # for reproducibility
split <- sample.split(applications$AU_move_indicator, SplitRatio = 0.7)
training_set <- subset(applications, split == TRUE)
test_set <- subset(applications, split == FALSE)
```

We have to fit our model onto the training set.

#```{r} # Check for NA values in gender and count them sum(is.na(applications$gender.x))

Check the unique values and data type of gender before conversion

unique(applications$gender.x)str(applications$gender.x)

If the number of NA values is significant, decide how to handle them (e.g., imputation)

If imputation is not feasible or desirable, you might consider excluding these rows

Convert gender to factor after handling NA values, if any

applications$gender.x < -as.factor(applications$gender.x)

#```

```
summary(applications)

##  application_number  filing_date         examiner_name_last
examiner_name_first
##  Length:2018477     Min.   :2000-01-02   Length:2018477     Length:2018477
##  Class :character   1st Qu.:2005-03-30   Class :character   Class
:character
##  Mode  :character   Median :2009-07-23   Mode  :character   Mode
:character
##                     Mean   :2009-03-23
##                     3rd Qu.:2013-05-22
##                     Max.   :2017-05-26
##
##  examiner_name_middle  examiner_id    examiner_art_unit  uspc_class
##  Length:2018477       Min.   :59012   Min.   :1600      Length:2018477
##  Class :character     1st Qu.:66476   1st Qu.:1671      Class :character
##  Mode  :character     Median :75243   Median :1773      Mode  :character
##                       Mean   :78712   Mean   :1928
##                       3rd Qu.:93754   3rd Qu.:2171
##                       Max.   :99990   Max.   :2498
##                       NA's   :9229
##  uspc_subclass      patent_number      patent_issue_date
##  Length:2018477     Length:2018477     Min.   :1997-03-04
##  Class :character   Class :character   1st Qu.:2008-04-29
##  Mode  :character   Mode  :character   Median :2012-05-22
##                                        Mean   :2011-06-20
##                                        3rd Qu.:2015-01-20
##                                        Max.   :2017-06-20
##                                        NA's   :931178
##   abandon_date        disposal_type      appl_status_code appl_status_date
##  Min.   :1965-07-20   Length:2018477     Min.   :  1.0    Length:2018477
##  1st Qu.:2008-06-23   Class :character   1st Qu.:150.0    Class :character
##  Median :2011-04-19   Mode  :character   Median :150.0    Mode  :character
##  Mean   :2011-01-28                      Mean   :145.9
```

```
##  3rd Qu.:2014-04-15                      3rd Qu.:161.0  
##  Max.   :2050-06-30                      Max.   :865.0  
##  NA's   :1417057                         NA's   :4609   
##        tc         gender.x             race           earliest_date       
##  Min.   :1600   Length:2018477     Length:2018477     Min.   :2000-01-02  
##  1st Qu.:1600   Class :character   Class :character   1st Qu.:2000-01-11  
##  Median :1700   Mode  :character   Mode  :character   Median :2000-08-18  
##  Mean   :1877                                         Mean   :2002-03-10  
##  3rd Qu.:2100                                         3rd Qu.:2003-09-29  
##  Max.   :2400                                         Max.   :2016-03-03  
##                                                                           
##   latest_date          tenure_days      AU_move_indicator   gender.y        
##  Min.   :2000-09-14   Min.   :     27   Min.   :0.0000    Length:2018477    
##  1st Qu.:2017-05-19   1st Qu.:   4963   1st Qu.:0.0000    Class :character  
##  Median :2017-05-20   Median :   6094   Median :1.0000    Mode  :character  
##  Mean   :2030-05-04   Mean   :  10282   Mean   :0.7242                      
##  3rd Qu.:2017-05-23   3rd Qu.:   6336   3rd Qu.:1.0000                      
##  Max.   :9468-10-16   Max.   :2727903   Max.   :1.0000                      
## 
```

```
str(applications)
```

```
## tibble [2,018,477 × 23] (S3: tbl_df/tbl/data.frame)
##  $ application_number  : chr [1:2018477] "08284457" "08413193" "08531853"
"08637752" ...
##  $ filing_date         : Date[1:2018477], format: "2000-01-26" "2000-10-
11" ...
##  $ examiner_name_last  : chr [1:2018477] "HOWARD" "YILDIRIM" "HAMILTON"
"MOSHER" ...
##  $ examiner_name_first : chr [1:2018477] "JACQUELINE" "BEKIR" "CYNTHIA"
"MARY" ...
##  $ examiner_name_middle: chr [1:2018477] "V" "L" NA NA ...
##  $ examiner_id         : num [1:2018477] 96082 87678 63213 73788 77294 ...
##  $ examiner_art_unit   : num [1:2018477] 1764 1764 1752 1648 1762 ...
##  $ uspc_class          : chr [1:2018477] "508" "208" "430" "530" ...
##  $ uspc_subclass       : chr [1:2018477] "273000" "179000" "271100"
"388300" ...
##  $ patent_number       : chr [1:2018477] "6521570" "6440298" "5607816"
"6927281" ...
##  $ patent_issue_date   : Date[1:2018477], format: "2003-02-18" "2002-08-
27" ...
##  $ abandon_date        : Date[1:2018477], format: NA NA ...
##  $ disposal_type       : chr [1:2018477] "ISS" "ISS" "ISS" "ISS" ...
##  $ appl_status_code    : num [1:2018477] 150 250 250 250 161 150 135 161
161 250 ...
##  $ appl_status_date    : chr [1:2018477] "30jan2003 00:00:00" "27sep2010
00:00:00" "30mar2009 00:00:00" "07sep2009 00:00:00" ...
##  $ tc                  : num [1:2018477] 1700 1700 1700 1600 1700 1700
1600 1600 1600 1700 ...
##  $ gender.x            : chr [1:2018477] "female" "male" "female" "female"
```

```
...
##  $ race                : chr [1:2018477] "white" "white" "white" "white"
...
##  $ earliest_date       : Date[1:2018477], format: "2000-01-10" "2000-01-
04" ...
##  $ latest_date         : Date[1:2018477], format: "2016-04-01" "2016-09-
09" ...
##  $ tenure_days         : num [1:2018477] 5926 6093 6344 6331 6332 ...
##  $ AU_move_indicator   : int [1:2018477] 0 0 1 0 1 1 1 1 1 1 ...
##  $ gender.y            : chr [1:2018477] "female" NA "female" "female" ...
```

```
# Print structure and names of the applications data frame
str(applications)
```

```
## tibble [2,018,477 × 23] (S3: tbl_df/tbl/data.frame)
##  $ application_number  : chr [1:2018477] "08284457" "08413193" "08531853"
"08637752" ...
##  $ filing_date         : Date[1:2018477], format: "2000-01-26" "2000-10-
11" ...
##  $ examiner_name_last  : chr [1:2018477] "HOWARD" "YILDIRIM" "HAMILTON"
"MOSHER" ...
##  $ examiner_name_first : chr [1:2018477] "JACQUELINE" "BEKIR" "CYNTHIA"
"MARY" ...
##  $ examiner_name_middle: chr [1:2018477] "V" "L" NA NA ...
##  $ examiner_id         : num [1:2018477] 96082 87678 63213 73788 77294 ...
##  $ examiner_art_unit   : num [1:2018477] 1764 1764 1752 1648 1762 ...
##  $ uspc_class          : chr [1:2018477] "508" "208" "430" "530" ...
##  $ uspc_subclass       : chr [1:2018477] "273000" "179000" "271100"
"388300" ...
##  $ patent_number       : chr [1:2018477] "6521570" "6440298" "5607816"
"6927281" ...
##  $ patent_issue_date   : Date[1:2018477], format: "2003-02-18" "2002-08-
27" ...
##  $ abandon_date        : Date[1:2018477], format: NA NA ...
##  $ disposal_type       : chr [1:2018477] "ISS" "ISS" "ISS" "ISS" ...
##  $ appl_status_code    : num [1:2018477] 150 250 250 250 161 150 135 161
161 250 ...
##  $ appl_status_date    : chr [1:2018477] "30jan2003 00:00:00" "27sep2010
00:00:00" "30mar2009 00:00:00" "07sep2009 00:00:00" ...
##  $ tc                  : num [1:2018477] 1700 1700 1700 1600 1700 1700
1600 1600 1600 1700 ...
##  $ gender.x            : chr [1:2018477] "female" "male" "female" "female"
...
##  $ race                : chr [1:2018477] "white" "white" "white" "white"
...
##  $ earliest_date       : Date[1:2018477], format: "2000-01-10" "2000-01-
04" ...
##  $ latest_date         : Date[1:2018477], format: "2016-04-01" "2016-09-
09" ...
##  $ tenure_days         : num [1:2018477] 5926 6093 6344 6331 6332 ...
##  $ AU_move_indicator   : int [1:2018477] 0 0 1 0 1 1 1 1 1 1 ...
##  $ gender.y            : chr [1:2018477] "female" NA "female" "female" ...
```

```r
names(applications)
```

```
##  [1] "application_number"   "filing_date"          "examiner_name_last"  
##  [4] "examiner_name_first"  "examiner_name_middle" "examiner_id"         
##  [7] "examiner_art_unit"    "uspc_class"           "uspc_subclass"       
## [10] "patent_number"        "patent_issue_date"    "abandon_date"        
## [13] "disposal_type"        "appl_status_code"     "appl_status_date"    
## [16] "tc"                   "gender.x"             "race"                
## [19] "earliest_date"        "latest_date"          "tenure_days"         
## [22] "AU_move_indicator"    "gender.y"
```

```r
# Load required packages
library(caTools)
library(dplyr)

# Check if 'gender.x' column exists in the applications data frame
if ("gender.x" %in% names(applications)) {
  # Convert 'gender.x' to factor, and other categorical variables as well
  applications <- applications %>%
      mutate(
          gender.x = as.factor(gender.x),
          disposal_type = as.factor(disposal_type),
          race = as.factor(race)
      )
} else {
  cat("'gender.x' column not found in applications data frame.\n")
}

# Further processing if 'gender.x' exists
if ("gender.x" %in% names(applications)) {
  # Print some information about gender.x after conversion
  cat("Number of rows in applications:", nrow(applications), "\n")
  cat("Number of unique values in applications$gender.x:",
length(unique(applications$gender.x)), "\n")
  cat("First few values of applications$gender.x:",
head(applications$gender.x), "\n")

  # Handle non-numeric values in uspc_class
  applications$uspc_class <-
as.numeric(as.character(applications$uspc_class))

  # Check for NAs after conversion and decide how to handle them
  sum_na_uspc_class <- sum(is.na(applications$uspc_class))
  cat("Number of NA values in applications$uspc_class:", sum_na_uspc_class,
"\n")

  # Splitting the data into a smaller subset, training (70%) and test (30%)
sets
  set.seed(123) # for reproducibility
  applications_subset <- applications[sample(nrow(applications), 10000), ]

  # Ensure loading caTools before using sample.split
  split <- sample.split(applications_subset$AU_move_indicator, SplitRatio =
0.7)
  training_set <- subset(applications_subset, split == TRUE)
  test_set <- subset(applications_subset, split == FALSE)

  # Fitting the model on the training set
  model <- glm(AU_move_indicator ~ filing_date + examiner_art_unit +
uspc_class + disposal_type + gender.x + race + tenure_days,
               family = binomial(link = 'logit'),
               data = training_set)
} else {
  cat("Skipping model fitting as 'gender.x' is not present in the
applications data frame.\n")
}
```

```
## Number of rows in applications: 2018477
## Number of unique values in applications$gender.x: 2
## First few values of applications$gender.x: 1 2 1 1 2 1

## Warning: NAs introduced by coercion

## Number of NA values in applications$uspc_class: 34
```

```
summary(model)
```

```
##
## Call:
## glm(formula = AU_move_indicator ~ filing_date + examiner_art_unit +
##     uspc_class + disposal_type + gender.x + race + tenure_days,
##     family = binomial(link = "logit"), data = training_set)
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        2.406e+00  3.140e-01   7.662 1.83e-14 ***
## filing_date       -1.123e-04  1.965e-05  -5.715 1.10e-08 ***
## examiner_art_unit  1.651e-04  1.034e-04   1.597   0.1104
## uspc_class        -8.672e-04  1.651e-04  -5.252 1.51e-07 ***
## disposal_typeISS   3.855e-01  6.267e-02   6.152 7.66e-10 ***
## disposal_typePEND -4.234e-02  9.309e-02  -0.455   0.6492
## gender.xmale       6.348e-02  6.137e-02   1.034   0.3010
## raceblack          2.892e-01  1.467e-01   1.972   0.0486 *
## raceHispanic       4.104e-02  1.603e-01   0.256   0.7979
## raceother          1.185e+01  1.447e+02   0.082   0.9347
## racewhite          5.530e-02  6.165e-02   0.897   0.3697
## tenure_days        3.243e-06  1.370e-06   2.367   0.0179 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8185.4  on 6999  degrees of freedom
## Residual deviance: 8007.1  on 6988  degrees of freedom
## AIC: 8031.1
##
## Number of Fisher Scoring iterations: 11
```

After fitting on the training set, we can tets our model using the test set.

```r
# Predicting probabilities on the test set
probabilities <- predict(model, newdata = test_set, type = "response")

# Binarizing the predictions based on a threshold (e.g., 0.5) ?
# predictions <- ifelse(probabilities > 0.5, 1, 0)
```

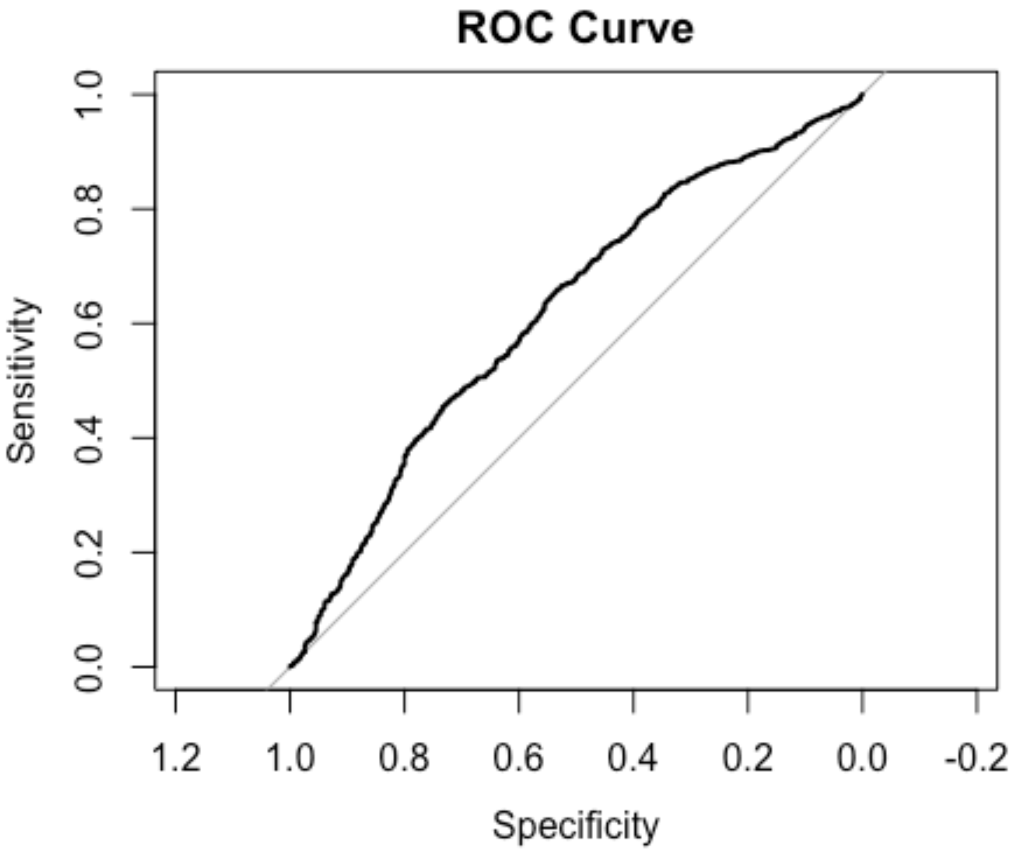
Now that we've tested our predictions, we can plot the ROC curve.

```r
# ROC Curve
roc_curve <- roc(test_set$AU_move_indicator, probabilities)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

plot(roc_curve, main = "ROC Curve")
```



We can also calculate the AUC using the ROC curve we found above.

```
# Calculating AUC
auc(roc_curve)

## Area under the curve: 0.6215
```