HW 4

Eric Rash

12/29/2023

This homework is designed to give you practice fitting a logistic regression and working with statistical/philosophical measures of fairness. We will work with the `titanic` dataset which we have previously seen in class in connection to decision trees.

Below I will preprocess the data precisely as we did in class. You can simply refer to `data_train` as your training data and `data_test` as your testing data.

```r
#this is all of the preprocessing done for the decision trees lecture.

path <- 'https://raw.githubusercontent.com/guru99-edu/R-Programming/master/titanic_data.csv'
titanic <-read.csv(path)
head(titanic)
```

```
##   x pclass survived                                            name    sex
## 1 1      1        1                   Allen, Miss. Elisabeth Walton female
## 2 2      1        1                  Allison, Master. Hudson Trevor   male
## 3 3      1        0                    Allison, Miss. Helen Loraine female
## 4 4      1        0            Allison, Mr. Hudson Joshua Creighton   male
## 5 5      1        0 Allison, Mrs. Hudson J C (Bessie Waldo Daniels) female
## 6 6      1        1                             Anderson, Mr. Harry   male
##      age sibsp parch ticket     fare   cabin embarked
## 1     29     0     0  24160 211.3375      B5        S
## 2 0.9167     1     2 113781   151.55 C22 C26        S
## 3      2     1     2 113781   151.55 C22 C26        S
## 4     30     1     2 113781   151.55 C22 C26        S
## 5     25     1     2 113781   151.55 C22 C26        S
## 6     48     0     0  19952    26.55     E12        S
##                         home.dest
## 1                    St Louis, MO
## 2 Montreal, PQ / Chesterville, ON
## 3 Montreal, PQ / Chesterville, ON
## 4 Montreal, PQ / Chesterville, ON
## 5 Montreal, PQ / Chesterville, ON
## 6                    New York, NY
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.2

##
## 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
```

```r
#replace ? with NA
replace_question_mark <- function(x) {
  if (is.character(x)) {
    x <- na_if(x, "?")
  }
  return(x)
}

titanic <- titanic %>%
  mutate_all(replace_question_mark)

set.seed(678)
shuffle_index <- sample(1:nrow(titanic))
head(shuffle_index)
```

```
## [1]   57  774  796 1044  681  920
```

```r
titanic <- titanic[shuffle_index, ]
head(titanic)
```

```
##         x pclass survived                                              name
## 57     57      1        1                         Carter, Mr. William Ernest
## 774   774      3        0                                   Dimic, Mr. Jovan
## 796   796      3        0                             Emir, Mr. Farred Chehab
## 1044 1044      3        1                          Murphy, Miss. Margaret Jane
## 681   681      3        0                                  Boulos, Mr. Hanna
## 920   920      3        0 Katavelas, Mr. Vassilios ('Catavelas Vassilios')
##         sex  age sibsp parch ticket   fare   cabin embarked     home.dest
## 57     male   36     1     2 113760    120 B96 B98        S Bryn Mawr, PA
## 774    male   42     0     0 315088 8.6625    <NA>        S          <NA>
## 796    male <NA>     0     0   2631  7.225    <NA>        C          <NA>
## 1044 female <NA>     1     0 367230   15.5    <NA>        Q          <NA>
## 681    male <NA>     0     0   2664  7.225    <NA>        C         Syria
## 920    male 18.5     0     0   2682 7.2292    <NA>        C          <NA>
```

```r
library(dplyr)
# Drop variables
clean_titanic <- titanic %>%
select(-c(home.dest, cabin, name, x, ticket)) %>%
#Convert to factor level
    mutate(pclass = factor(pclass, levels = c(1, 2, 3), labels = c('Upper', 'Middle', 'Lower')),
    survived = factor(survived, levels = c(0, 1), labels = c('No', 'Yes'))) %>%
na.omit()
#previously were characters
```

```r
clean_titanic$age <- as.numeric(clean_titanic$age)
clean_titanic$fare <- as.numeric(clean_titanic$fare)
glimpse(clean_titanic)
```

```
## Rows: 1,043
## Columns: 8
## $ pclass   <fct> Upper, Lower, Lower, Middle, Lower, Middle, Lower, Lower, Upp~
## $ survived <fct> Yes, No, No, No, No, No, No, No, Yes, No, Yes, No, No, Yes, N~
## $ sex      <chr> "male", "male", "male", "male", "female", "female", "male", "~
## $ age      <dbl> 36.0, 42.0, 18.5, 44.0, 19.0, 26.0, 23.0, 28.5, 64.0, 36.5, 4~
## $ sibsp    <int> 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0~
## $ parch    <int> 2, 0, 0, 0, 0, 1, 0, 0, 2, 2, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0~
## $ fare     <dbl> 120.0000, 8.6625, 7.2292, 13.0000, 16.1000, 26.0000, 7.8542, ~
## $ embarked <chr> "S", "S", "C", "S", "S", "S", "S", "S", "C", "S", "S", "S", "~
```

```r
create_train_test <- function(data, size = 0.8, train = TRUE) {
    n_row = nrow(data)
    total_row = size * n_row
    train_sample <- 1: total_row
    if (train == TRUE) {
        return (data[train_sample, ])
    } else {
        return (data[-train_sample, ])
    }
}
data_train <- create_train_test(clean_titanic, 0.8, train = TRUE)
data_test <- create_train_test(clean_titanic, 0.8, train = FALSE)
```

Create a table reporting the proportion of people in the training set surviving the Titanic. Do the same for the testing set. Comment on whether the current training-testing partition looks suitable.

```r
#calculates survival rates for train and test data
test_survival_rate <- (sum(data_test$survived == "Yes"))/(nrow(data_test))
train_survival_rate <- (sum(data_train$survived == "Yes"))/(nrow(data_train))
##makes prop table of survival rates
proportion_table <- data.frame(
  Set = c("Training", "Testing"),
  Proportion_of_Survivors = c(train_survival_rate, test_survival_rate)
)
proportion_table
```

```
##        Set Proportion_of_Survivors
## 1 Training               0.3980815
## 2  Testing               0.4449761
```

student input

Use the `glm` command to build a logistic regression on the training partition. `survived` should be your response variable and `pclass`, `sex`, `age`, `sibsp`, and `parch` should be your response variables.

```
##Builds a logistic model on training data
model <- glm(survived ~pclass + age + sibsp + parch,family=binomial(link='logit'),data=data_train)
summary(model)
```

```
##
## Call:
## glm(formula = survived ~ pclass + age + sibsp + parch, family = binomial(link = "logit"),
##     data = data_train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.021028   0.315423   6.407 1.48e-10 ***
## pclassMiddle -1.142704   0.218243  -5.236 1.64e-07 ***
## pclassLower  -2.239737   0.220892 -10.140  < 2e-16 ***
## age          -0.037390   0.006536  -5.720 1.06e-08 ***
## sibsp        -0.298433   0.097044  -3.075  0.00210 **
## parch         0.345236   0.093445   3.695  0.00022 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1121.27  on 833  degrees of freedom
## Residual deviance:  984.57  on 828  degrees of freedom
## AIC: 996.57
##
## Number of Fisher Scoring iterations: 4
```

We would now like to test whether this classifier is *fair* across the sex subgroups. It was reported that women and children were prioritized on the life-boats and as a result survived the incident at a much higher rate. Let us see if our model is able to capture this fact.

Subset your test data into a male group and a female group. Then, use the `predict` function on the male testing group to come up with predicted probabilities of surviving the Titanic for each male in the testing set. Do the same for the female testing group.

```
#subsets the data into male and female groups
m_split <- subset(data_test, sex=="male")
f_split <- subset(data_test, sex=="female")
#calculates individual likelihood of surviving
m.fitted.results <- predict(model,newdata=m_split,type='response')
f.fitted.results <- predict(model, newdata=f_split, type="response")
##adds column with predicted outcomes
m_split$prediction <-m.fitted.results
f_split$prediction <-f.fitted.results
```

Now recall that for this logistic *regression* to be a true classifier, we need to pair it with a decision boundary. Use an `if-else` statement to translate any predicted probability in the male group greater than 0.5 into `Yes` (as in Yes this individual is predicted to have survived). Likewise an predicted probability less than 0.5 should be translated into a `No`.

Do this for the female testing group as well, and then create a confusion matrix for each of the male and female test set predictions. You can use the `confusionMatrix` command as seen in class to expidite this process as well as provide you necessary metrics for the following questions.

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.3.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.3.3

## Loading required package: lattice
```

```
#creates a decision boundary of 0.5 for if ind is predicted to survive or not
m.surv <- ifelse(m.fitted.results > 0.5,"Yes","No")
f.surv <- ifelse(f.fitted.results > 0.5, "Yes", "No")
##adds columns with classifications
m_split$predict_surv <- m.surv
f_split$predict_surv <- f.surv
##creates confusion  matrices
confusionMatrix(as.factor(m.surv),reference=m_split$survived, positive = "Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction No Yes
##        No  76  21
##        Yes 21  11
##
##                Accuracy : 0.6744
##                  95% CI : (0.5864, 0.7543)
##     No Information Rate : 0.7519
##     P-Value [Acc > NIR] : 0.9816
##
##                   Kappa : 0.1273
##
##  Mcnemar's Test P-Value : 1.0000
##
##             Sensitivity : 0.34375
##             Specificity : 0.78351
##          Pos Pred Value : 0.34375
##          Neg Pred Value : 0.78351
##              Prevalence : 0.24806
##          Detection Rate : 0.08527
```

```
##    Detection Prevalence : 0.24806
##       Balanced Accuracy : 0.56363
##
##        'Positive' Class : Yes
##
```

```r
confusionMatrix(as.factor(f.surv),reference=f_split$survived, positive = "Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction No Yes
##        No  19  23
##        Yes  0  38
##
##                Accuracy : 0.7125
##                  95% CI : (0.6005, 0.8082)
##     No Information Rate : 0.7625
##     P-Value [Acc > NIR] : 0.88
##
##                   Kappa : 0.4397
##
##  Mcnemar's Test P-Value : 4.49e-06
##
##             Sensitivity : 0.6230
##             Specificity : 1.0000
##          Pos Pred Value : 1.0000
##          Neg Pred Value : 0.4524
##              Prevalence : 0.7625
##          Detection Rate : 0.4750
##    Detection Prevalence : 0.4750
##       Balanced Accuracy : 0.8115
##
##        'Positive' Class : Yes
##
```

We can see that indeed, at least within the testing groups, women did seem to survive at a higher proportion than men (24.8% to 76.3% in the testing set). Print a summary of your trained model and interpret one of the fitted coefficients in light of the above disparity.

```r
#student input
summary(model)
```

```
##
## Call:
## glm(formula = survived ~ pclass + age + sibsp + parch, family = binomial(link = "logit"),
##     data = data_train)
##
## Coefficients:
```

```
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.021028   0.315423   6.407 1.48e-10 ***
## pclassMiddle -1.142704   0.218243  -5.236 1.64e-07 ***
## pclassLower  -2.239737   0.220892 -10.140  < 2e-16 ***
## age          -0.037390   0.006536  -5.720 1.06e-08 ***
## sibsp        -0.298433   0.097044  -3.075  0.00210 **
## parch         0.345236   0.093445   3.695  0.00022 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1121.27  on 833  degrees of freedom
## Residual deviance:  984.57  on 828  degrees of freedom
## AIC: 996.57
##
## Number of Fisher Scoring iterations: 4
```

Keeping in mind what we know about the disparity in predicted survival rates between males and females on the Titanic we can look at the "parch" variable. The coefficient on this variable is 0.345236, which tells us that for every child a person has there is a small positive impact on your log-likelihood of surviving the wreck.

Now let's see if our model is *fair* across this explanatory variable. Calculate five measures (as defined in class) in this question: the Overall accuracy rate ratio between females and males, the disparate impact between females and males, the statistical parity between females and males, and the predictive equality as well as equal opportunity between females and males (collectively these last two comprise equalized odds). Set a reasonable ϵ each time and then comment on which (if any) of these five criteria are met.

```r
#OARR
f_pv_acc <-(38+19)/(38+19+23)
m_pv_acc <-(26+11)/(26+11+21+21)
OARR <- f_pv_acc / m_pv_acc
#DI
f_di <- 19/(38+19+23)
m_di <-(76+21)/(76+21+21+11)
DI <- f_di/m_di
#SP
SP <- f_di-m_di
#PE
f_pe <-(19+38)/19
m_pe <- (76+11)/(76+21)
PE <- f_pe - m_pe
##EO
f_eo <- (19+38+23-19)/(19+38+23-19-38)
m_eo <- (76+21+21+11-76-21)/(76+21+21+11-76-11)
EO <- f_eo/m_eo


cat("Overall Accuracy Rate Ratio (OARR): ", OARR, "\n")
```

```
## Overall Accuracy Rate Ratio (OARR):  1.521284
```

```r
cat("Disparate Impact (DI): ", DI, "\n")
```

```
## Disparate Impact (DI):  0.3158505
```

```r
cat("Statistical Parity (SP): ", SP, "\n")
```

```
## Statistical Parity (SP):  -0.514438
```

```r
cat("Predictive Equality (PE): ", PE, "\n")
```

```
## Predictive Equality (PE):  2.103093
```

```r
cat("Equal Opportunity (EO): ", EO, "\n")
```

```
## Equal Opportunity (EO):  3.480978
```

With an epsilon of 0.05, there are no measures of statistical fairness that would approve of our model.

It is always important for us to interpret our results in light of the original data and the context of the analysis. In this case, it is relevant that we are analyzing a historical event post-facto and any disparities across demographics identified are unlikely to be replicated. So even though our model fails numerous of the statistical fairness criteria, I would argue we need not worry that our model could be misused to perpetuate discrimination in the future. After all, this model is likely not being used to prescribe a preferred method of treatment in the future.

Even so, provide a *philosophical* notion of justice or fairness that may have motivated the Titanic survivors to act as they did. Spell out what this philosophical notion or principle entails?

The thought process of those on the Titanic may have been reflective of those stated by Utilitarians. The idea being that you could maximize future happiness by protecting those who would likely live longer/those who have the ability to have kids and promote future generations. The utilitarian calculus here would argue that children who will live longer have more opportunity to benefit society than an old man.