Project

Juraj Uhlar

6/18/2019

Reading data

We begin by reading the data from a CSV file and processing it. We create a new variable `immigr_binary` which is a binary interpretation of the original 4-point scale `immigr` variable describing the subjects attitude towards immigration. `immigr_binary` is our target variable. We also turn some categorical, but numerically represented variables into factors for easier readability and make sure that factors are ordered correctly, where applicable.

```
# Reads data
framing = read.csv("framing.csv", header = T, sep = ",", dec = ".")

# Makes sure there are no NAs (there are not)
anyNA(framing)
```

```
## [1] FALSE
```

```
# Create news variable tracking attitude towards imigration as either generally positive or generally n
framing$immigr_binary = ifelse(framing$immigr >= 3, "Negative attitude", "Positive attitude")
framing$immigr_binary = factor(framing$immigr_binary)

# Turn numerical variables into factors where necessary
framing$cond = factor(framing$cond, levels = c(1, 2, 3, 4),
               labels = c("NegativeLatino", "NegativeEuropean", "PositiveLatino", "PositiveEuropean"))

framing$tone = factor(framing$tone, levels = c(0, 1), labels = c("Positive", "Negative"))
framing$eth = factor(framing$eth, levels = c(0, 1), labels = c("European", "Latino"))
framing$treat = factor(framing$treat, levels = c(0, 1), labels = c("Other", "Negative Latino"))
framing$anti_info = factor(framing$anti_info, levels = c(0, 1), labels = c("No", "Yes"))
framing$cong_mesg = factor(framing$cong_mesg, levels = c(0, 1), labels = c("No", "Yes"))

# Make ordered factors ordered
framing$anx = factor(framing$anx, levels=c("not anxious at all", "a little anxious", "somewhat anxious"
framing$educ = factor(framing$educ, levels=c("less than high school", "high school", "some college", "ba
framing$english = factor(framing$english, levels=c( "Strongly Oppose", "Oppose",  "Favor",  "Strongly Fa

# Looks at the final structure
# str(framing)
```

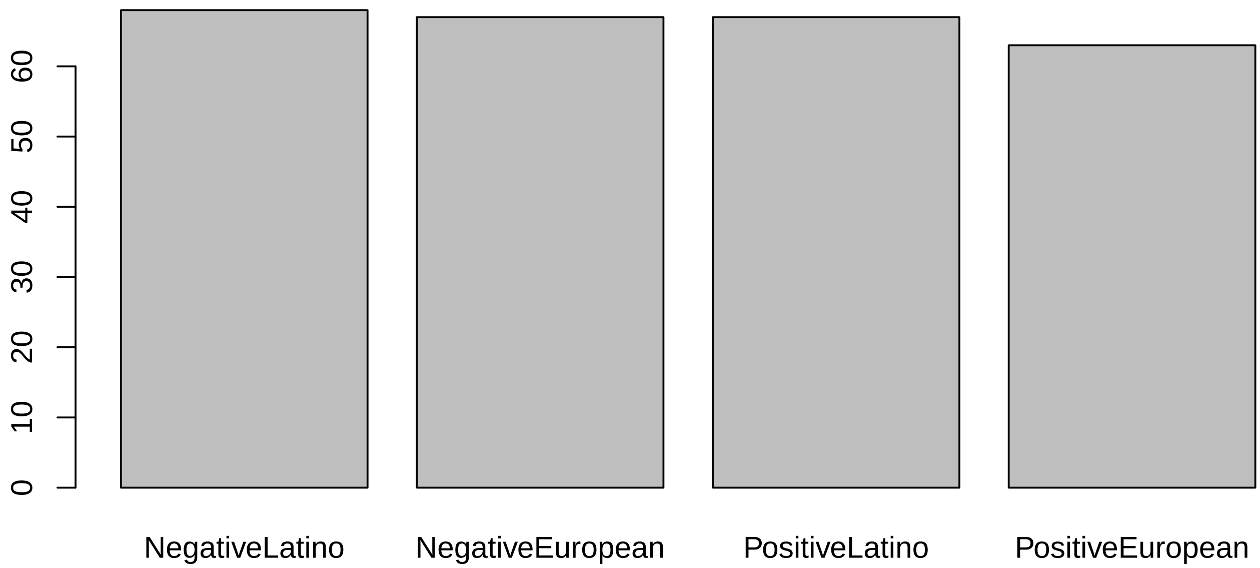
Exploratory data analysis

We begin by looking at the disribution of each variable in the dataset, trying to spot skewed distribution or outliers that might affect the accuracy of our prediction models.

Study conditions

We can see that observations are roughly equally divided into four groups based on treatment (ethnicity and tone of presented information), as would be expected in scientific study.

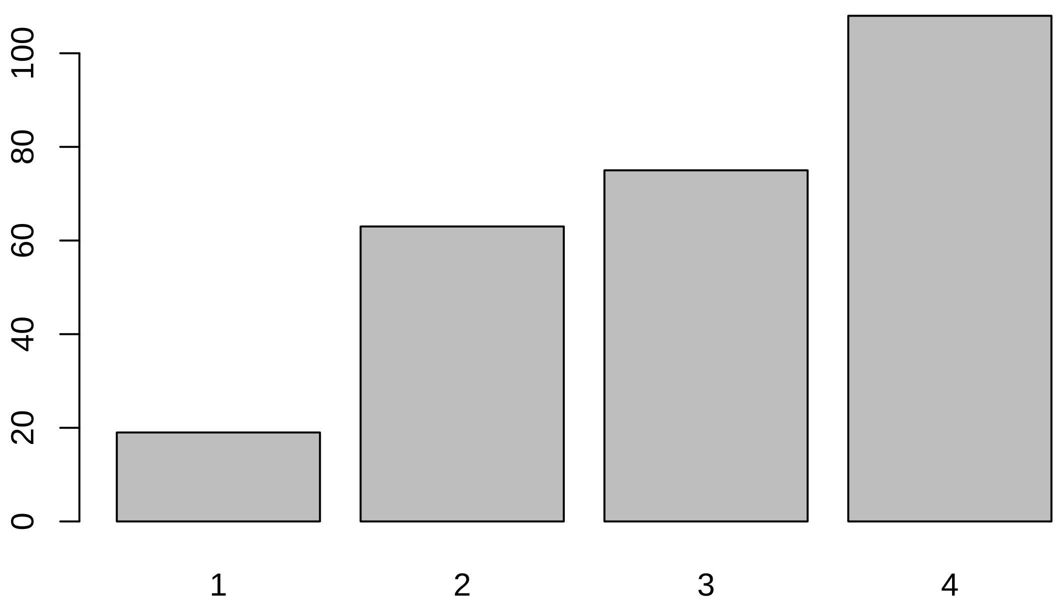
```r
barplot(table(framing$cond))
```



Attitude towards immigration

Looking at the target variable, we can observe that most subjects hold a negative view towards immigration. Negative views outnumber positive ones more than 2:1.

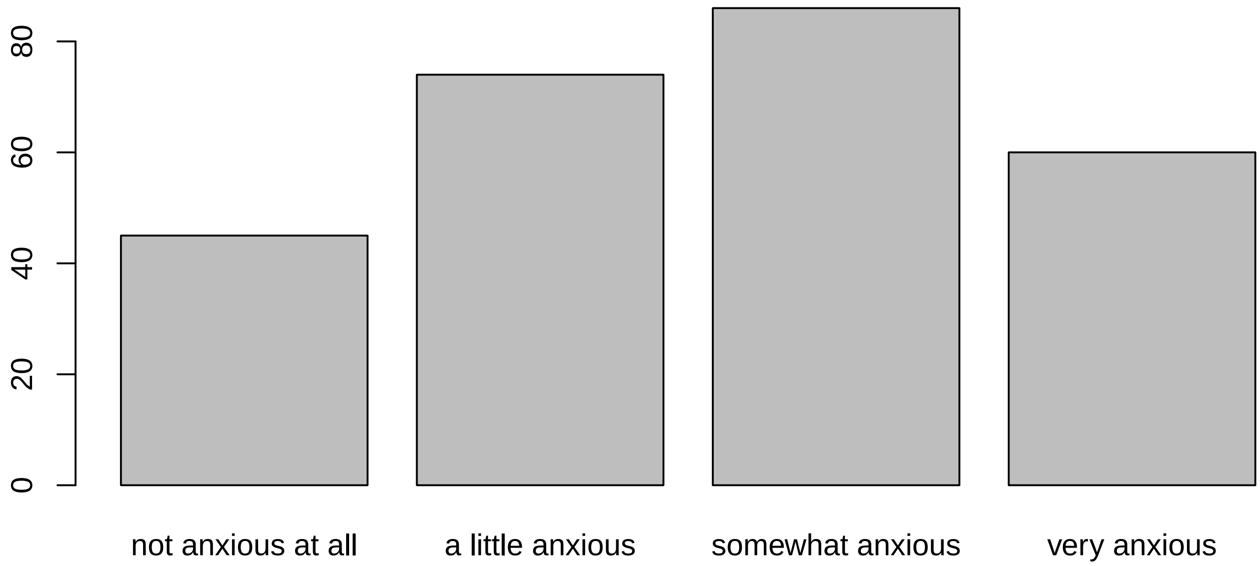
```r
barplot(table(framing$immigr_binary))
barplot(table(framing$immigr))
```



Anxiety about immigration

Only 17 % of subjects are not anxious about immigration at all.

```r
barplot(table(framing$anx))
```




```r
table(framing$anx)/nrow(framing)*100
```

```
##
## not anxious at all   a little anxious   somewhat anxious
##           16.98113           27.92453           32.45283
##       very anxious
##           22.64151
```

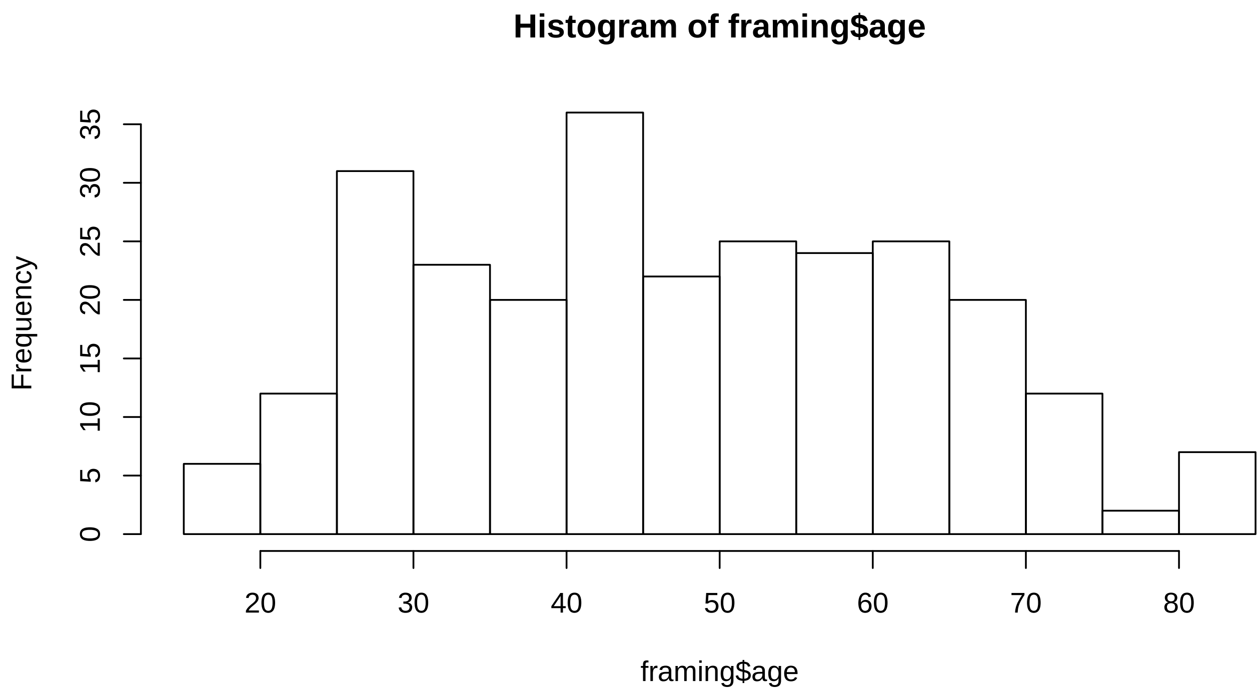
Age

Age of subjects approaches normal distribution. The youngest subject is 18, the oldest one is 85, with median subject age of 47 (mean is 48). No age group is significantly over- or under-represented.

```r
summary(framing$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   18.00   35.00   47.00   47.77   60.00   85.00
```

```r
hist(framing$age)
```



Gender

The data shows a reasoble gender split of 52% women and 48% men.

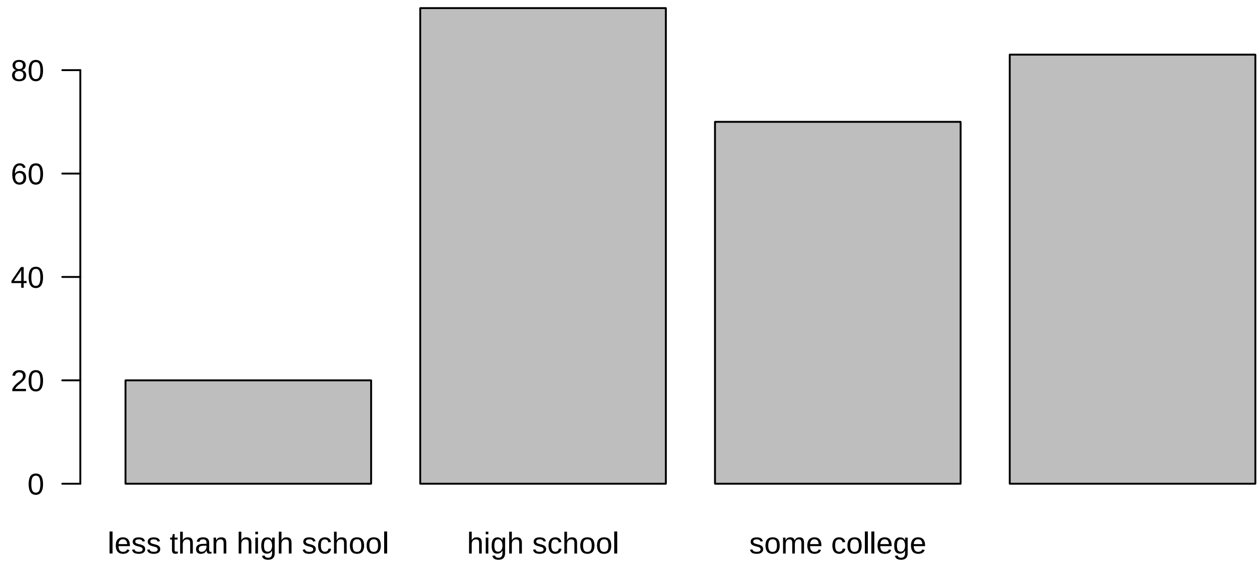
```
summary(framing$gender) / nrow(framing)
```

```
##    female      male
## 0.5245283 0.4754717
```

Education level

Education level distribution seems reasonably reflective of American society at large with 31% of subjects holding a college degree and 8% of subjects not not having a high school diploma.

```
barplot(summary(framing$educ), las = 1)
```



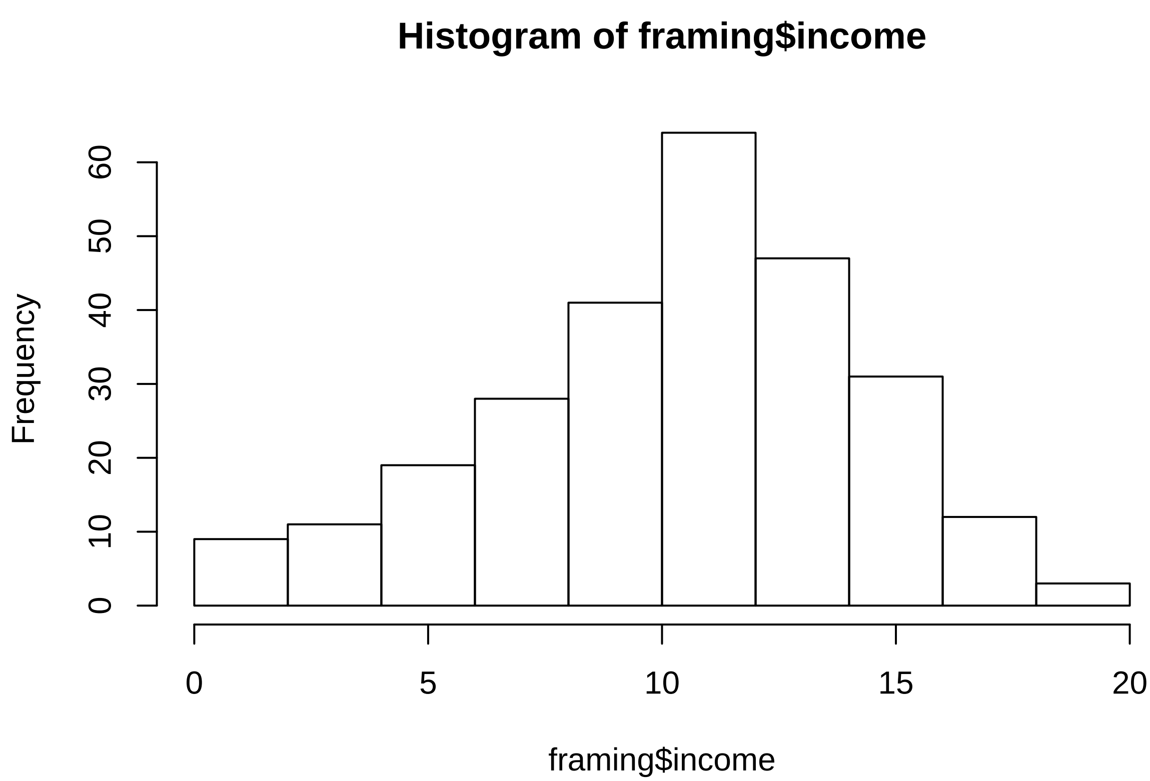

```
summary(framing$educ) / nrow(framing) * 100
```

```
##       less than high school                 high school
##                     7.54717                    34.71698
##                some college bachelor's degree or higher
##                    26.41509                    31.32075
```

Income

Subjects income is normally distributed.

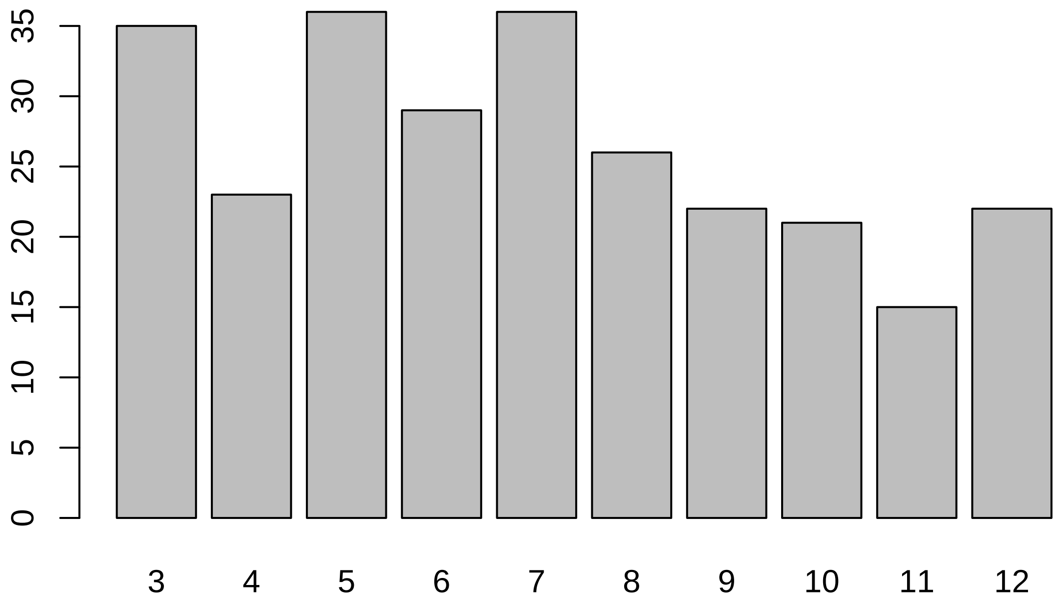
```
hist(framing$income)
```



Emotional response

The emotional response to the experiment is roughly evenly distrubuted among subjects, but skews towards negative emotional response. (3 indicates the most negative feeling)

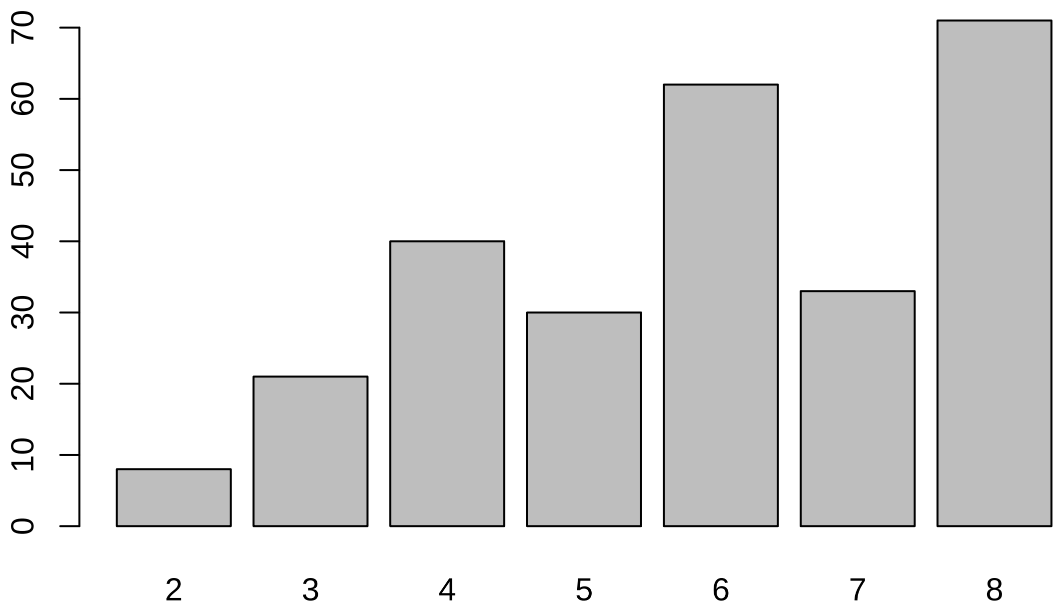
```
barplot(table(framing$emo))
```



Percieved harm caused by immigration

Most subjects percieve harm caused by immigration as high.

```
barplot(table(framing$p_harm))
```



Request for information from anti-immigration organizations

11% of subjects wanted to receive information from anti-immigration organizations.

```r
summary(framing$anti_info) / nrow(framing)
```

```
##       No      Yes
## 0.890566 0.109434
```

Request to send message to Congress

33% of subjects requested sending an anti-immigration message to Congress on their behalf.

```r
summary(framing$cong_mesg) / nrow(framing)
```

```
##        No       Yes
## 0.6679245 0.3320755
```

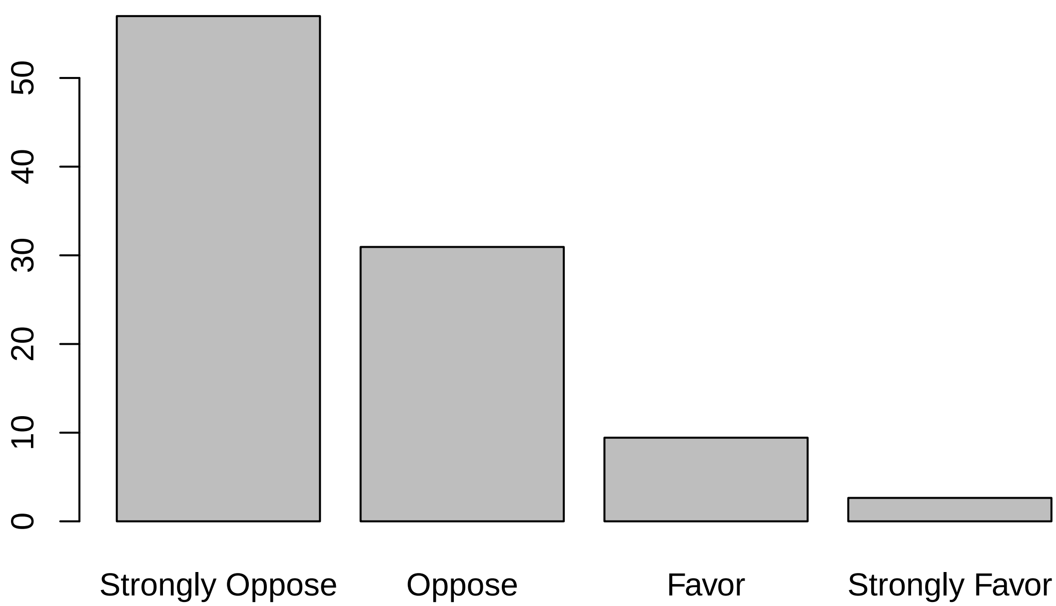
Making English the official language of USA

Somewhat surprisingly (considering majority of subjects were opposed to immigration), a majority of subjects strongly oppose a law making English the official language of the U.S.

```r
summary(framing$english) / nrow(framing) * 100
```

```
## Strongly Oppose          Oppose           Favor  Strongly Favor
##       56.981132       30.943396        9.433962        2.641509
```

```r
barplot(summary(framing$english) / nrow(framing) * 100)
```



Group analysis

We continue with basic exploration of relationships between variables

Percieved harm

Percieved harm by immigration broken down by study conditions. We see that negative news coverage leads to higher percieved harm caused by immigration. Latino etnicity clues make the effect stronger. Curiously, latino ethnicity cues lead to less percieved harm with positive news than european etnicity clues.

```r
tapply(framing$p_harm, framing$cond, FUN=mean)
```

```
##   NegativeLatino NegativeEuropean   PositiveLatino PositiveEuropean
##         6.264706         6.194030         5.402985         5.666667
```

```r
# tapply(framing$p_harm, framing$treat, FUN=mean)
```

Anxiety

Anxiety about immigration broken down by study conditions. Contrary to expectations, according to our dataset, negative news coverage leads to less anxiety than positve news coverage. This contradicts the underlying study and could indicate that the values in our dataset are incorrectly coded or desribed.

```r
tapply(as.numeric(framing$anx), framing$cond, FUN=mean)
```

```
##   NegativeLatino NegativeEuropean   PositiveLatino PositiveEuropean
##         2.235294         2.686567         2.820896         2.698413
```

```r
# tapply(as.numeric(framing$anx), framing$treat, FUN=mean)
```

Attitude to immigration

Attitude to immigration broken down by anxiety levels. The association runs in the opposite direction than expected.

```r
tapply(framing$immigr, framing$anx, FUN=mean)
```

```
## not anxious at all   a little anxious   somewhat anxious
##           3.688889           3.405405           2.732558
##       very anxious
##           2.483333
```

Attitude toward immigration broken down by study conditions. Larger is more negative (according to the dictionary). This association runs as expected.

```r
tapply(framing$immigr, framing$cond, FUN=mean)
```

```
##   NegativeLatino NegativeEuropean   PositiveLatino PositiveEuropean
##         3.352941         3.074627         2.746269         2.920635
```

Demographic correlations with attitude to immigration

Let's use visualization to explore the relationships between demographic variables and negative attitude to immigration.

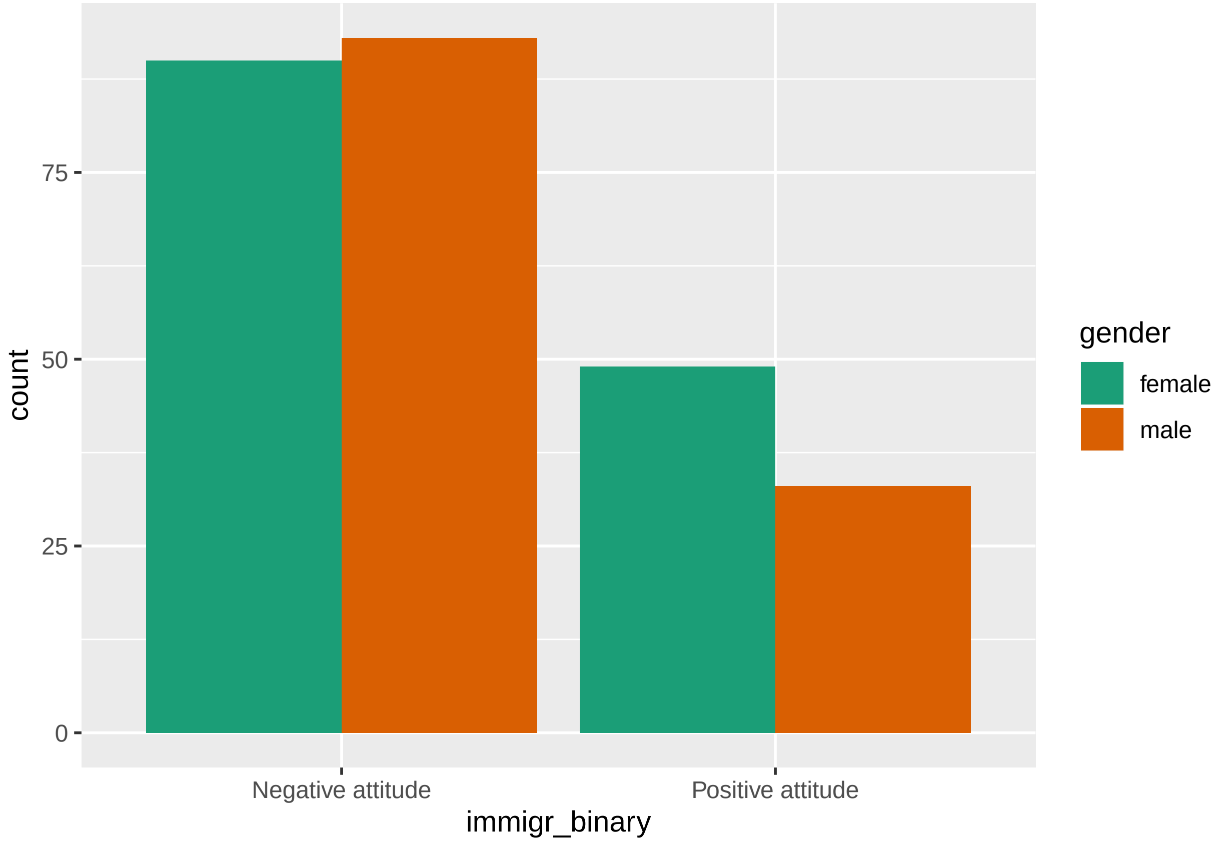
```r
# install.packages("ggplot2", repos = "http://cran.us.r-project.org")
library(ggplot2)
```

Gender

There is a slight gender effect present. Majority of people with positive attitude to immigration are women. People with negative attitude to immigration are more evenly split, with men taking a slight majority.

```r
genderB <- ggplot(framing, aes(x = immigr_binary, fill = gender)) +
          geom_bar(position = "dodge") +
          scale_fill_brewer(palette = 2, type = "qual")
```

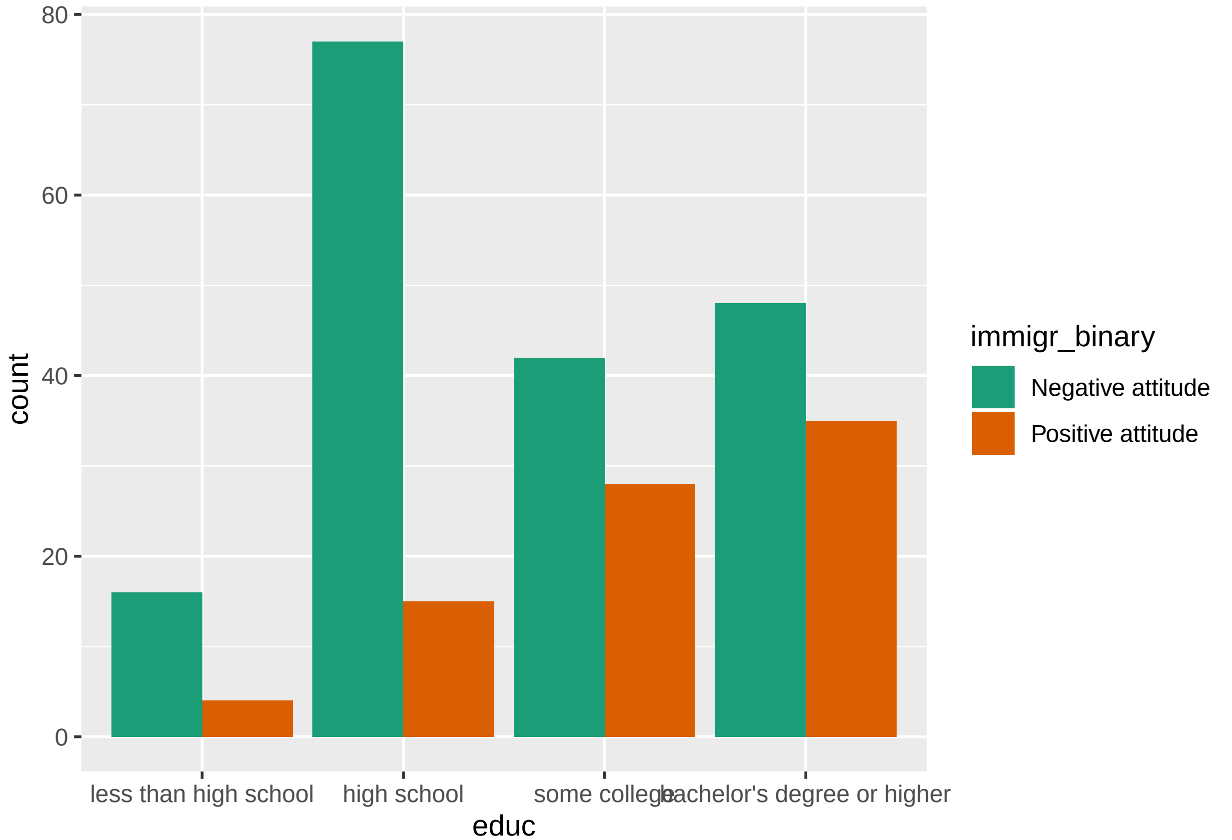
```
genderB
```



Education level

As the subjects' education level increases, they are less likely to have a negative attitude to immigration.

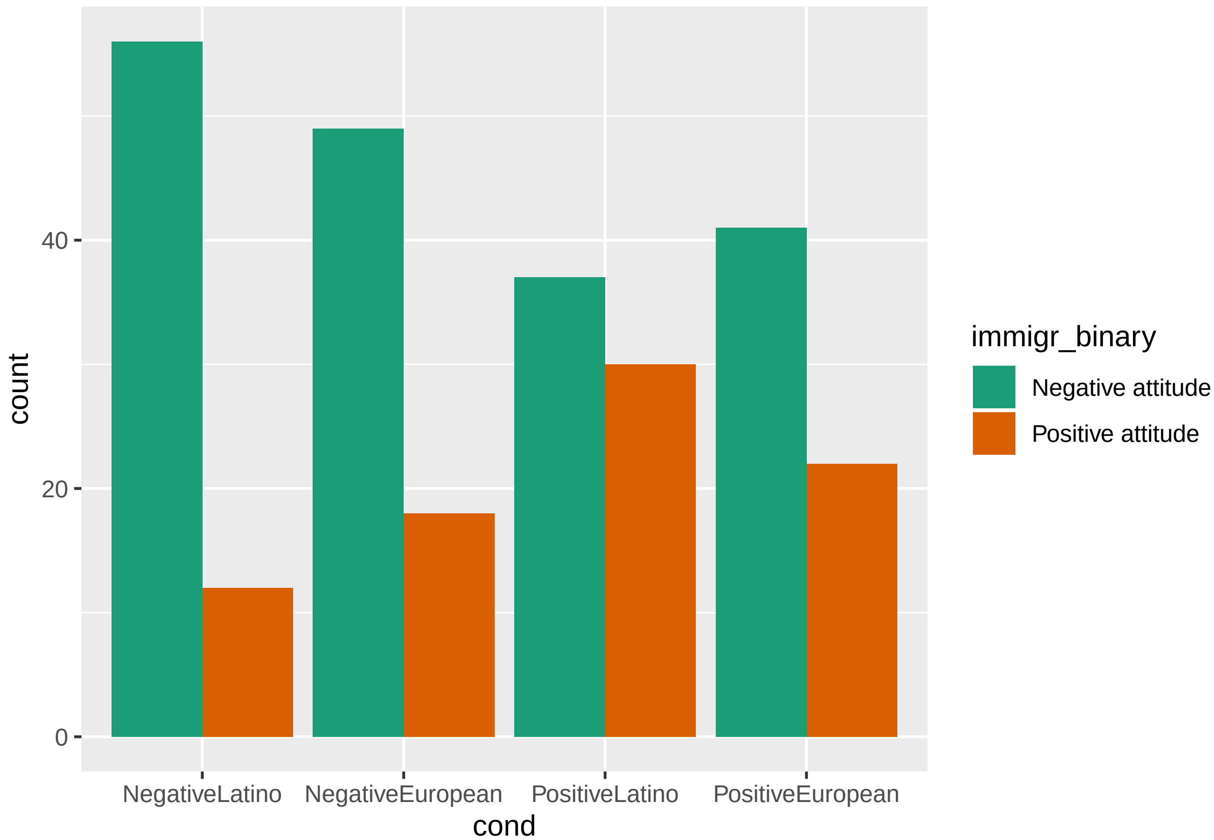
```
educB <- ggplot(framing, aes(x = educ, fill = immigr_binary)) +
          geom_bar(position = "dodge") +
          scale_fill_brewer(palette = 2, type = "qual")

educB
```



Study condition

We see that negative news coverage leads to more negative attitude to immigration. Latino etnicity clues make the effect stronger. Curiously, latino ethnicity cues lead to less negative attitude with positive news than european etnicity clues.

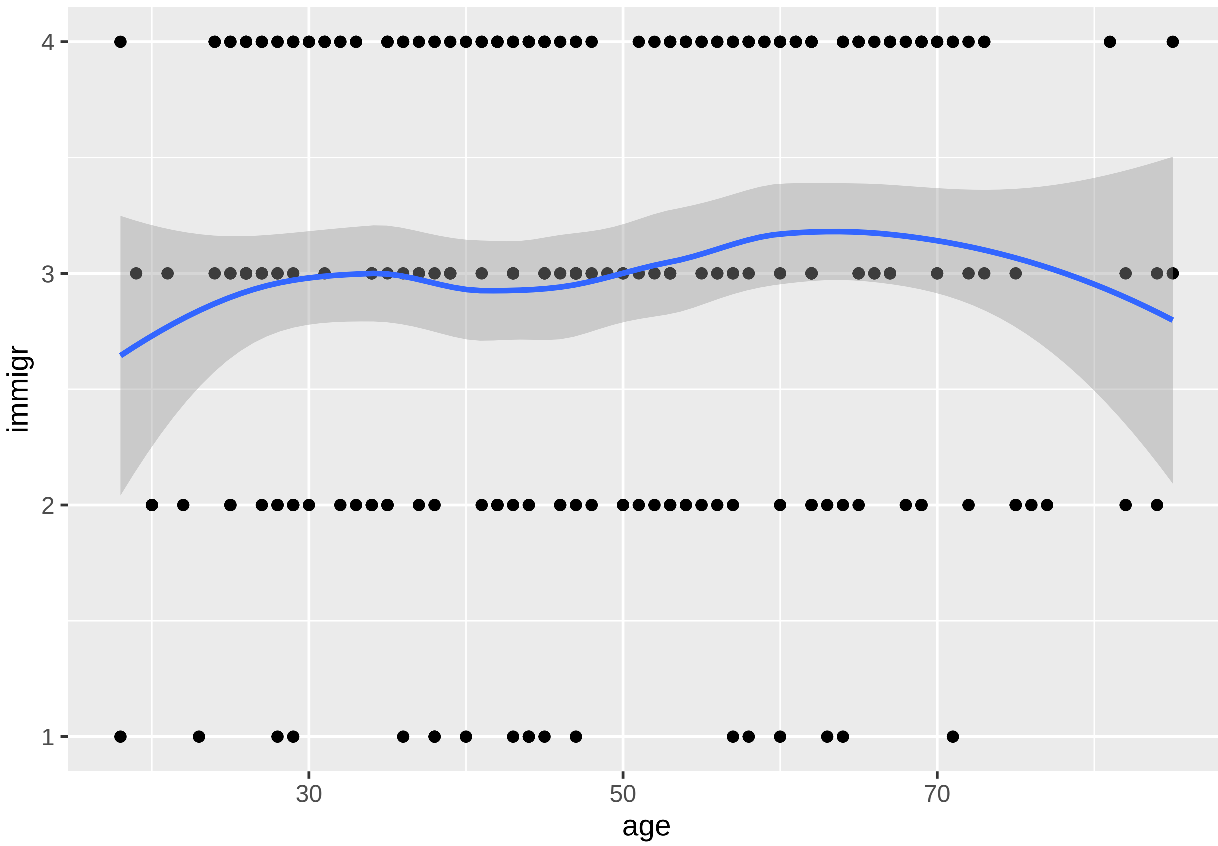
```
condB <- ggplot(framing, aes(x = cond, fill = immigr_binary)) +
          geom_bar(position = "dodge") +
          scale_fill_brewer(palette = 2, type = "qual")

condB
```



Age

There isn't a clear linear relationsip between age and negative attitude to immigration. There is slight bend toward positivity at very young and very old age. There are slight local maximums of negative attitude around age 35 and 60.

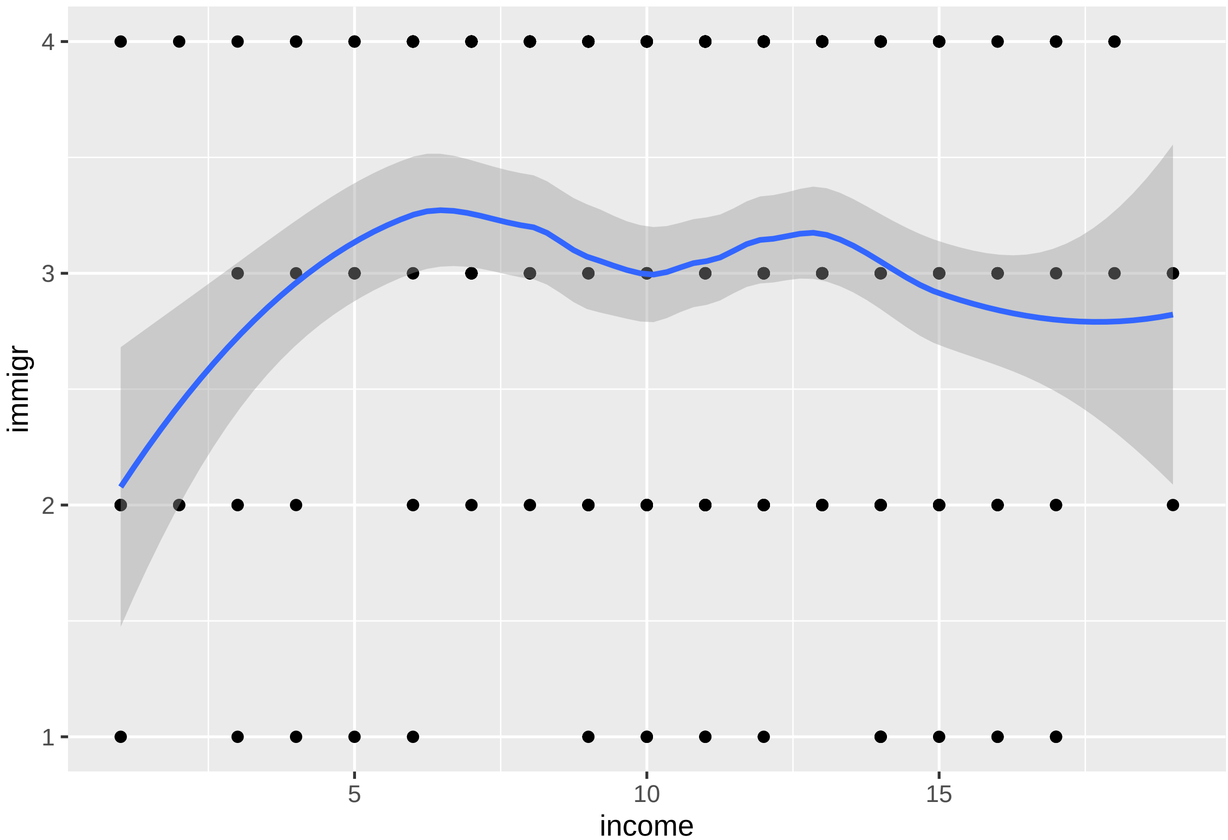
```r
ggplot(framing, aes(x = age, y = immigr)) +
    geom_point() +              # a layer of points
    geom_smooth()              # add a fitted line; try also: method = "lm")
```



Income

At the low end of the income distrubution, we can observe an almost linear relationship between income and negative attitude. This effect levels off at some point. We can observe two small local maximums of negative attitude that are similar in shape to the age curve.

```
ggplot(framing, aes(x = income, y = immigr)) +
    geom_point() +                # a layer of points
    geom_smooth()                # add a fitted line; try also: method = "lm")
```



Correlation matrices

We use correlation matrices to explore relationships between variables further.

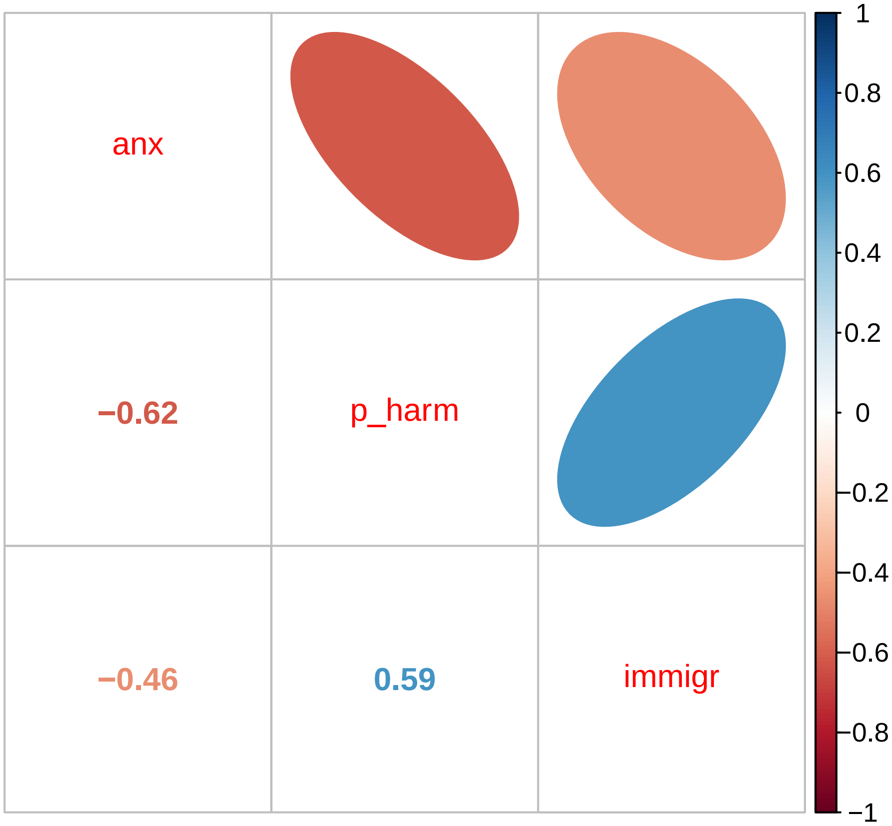
```
#Install required packages
# install.packages("corrplot", repos="http://cran.us.r-project.org")
# install.packages("gplots",repos="http://cran.us.r-project.org")
library(corrplot)
library(gplots)
```

Preparing data for analysis:

```
# Flip emo so that higher values signal more negative attitude (as with other variables)
framing$emo = framing$emo * (-1)

# Convert factors back into integers
framing$anx = as.integer(framing$anx)
framing$english = as.integer(framing$english)
framing$cong_mesg = as.integer(framing$cong_mesg) -1
framing$anti_info = as.integer(framing$anti_info) -1
```

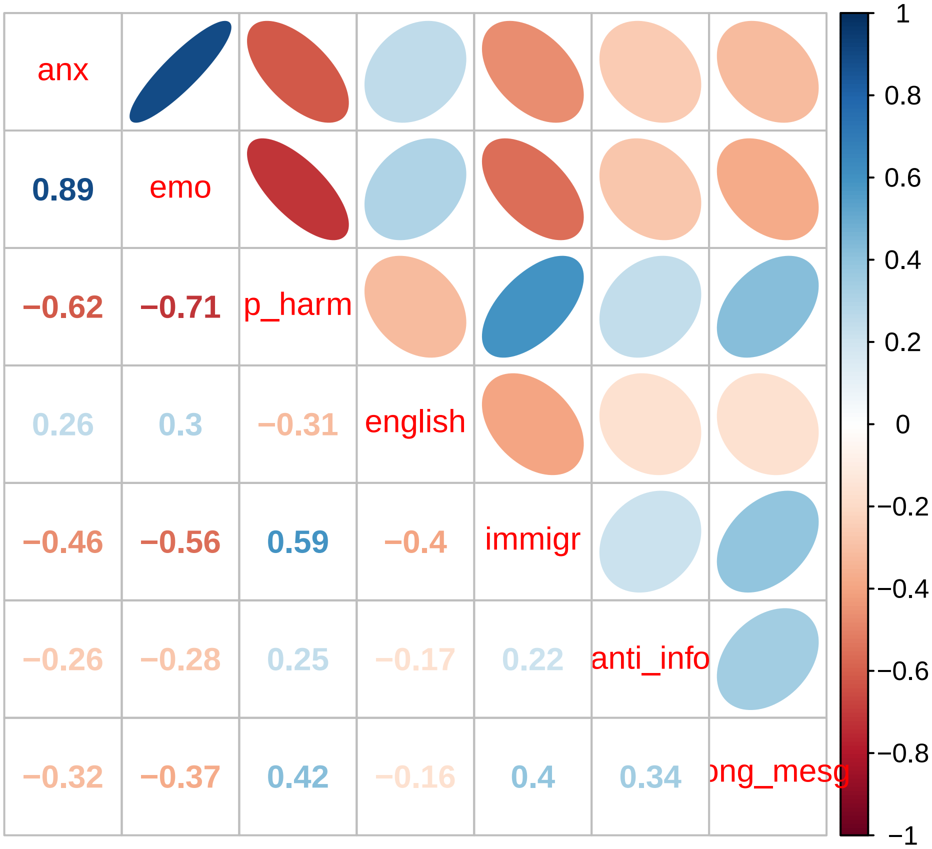
Starting with a simple correlation matrix of `p_harm`, `immigr` and `anx`. Intuitively, one would expect all of them to be positively correlated, but anxiety seemms to run in the opposite direction. There is -0.46 negative correlation between anxiety about immigration and negative attitude to immigration. There is -0.62 negative correlation between anxiety about immigration and perceived harm of immigration. Again, this could caused by a wrong encoding of the data.

```r
# p_harm, immigr and anxiety correlation matrix
corrplot.mixed(corr=cor(framing[ ,c(3,9,14)]), upper="ellipse")
```



Furthermore, we look at correlations of all "attitude" related variables expecting them to be largely positively correlated. This is not the case. We were not able to find a reasonable explanation for these results.

```r
corrplot.mixed(corr=cor(framing[ ,c(3,8,9,13:16)]), upper="ellipse")
```



Modeling and prediction

Installing the required packages:

```
# packages
options(repos=c(CRAN = "http://cran.us.r-project.org"))
# ROC AUC
# install.packages('pROC')
library(pROC)
# building decision trees
# install.packages("rpart")
library(rpart)
# plotting
# install.packages("rpart.plot")
library(rpart.plot)
```

Splittin data and preparation

```
# splitting the data into train and test
set.seed(777)
train.Index <- sample(1:nrow(framing), round(0.7*nrow(framing)), replace = F)
framing.train <- framing[train.Index,]
framing.test <- framing[-train.Index,]
```

```
# convert to numbers for calculations
framing.test$immigr_binary = as.integer(framing.test$immigr_binary) - 1
framing.train$immigr_binary = as.integer(framing.train$immigr_binary) - 1


# features to be used for model training
features <- c('cond', 'anx', 'age', 'educ', 'gender', 'income', 'emo', 'p_harm',
'tone', 'eth', 'english', 'anti_info', 'cong_mesg', 'immigr_binary')
```

Creating a baseline prediction

We create a naive baseline prediction based on probability of a negative attitude to immigration. We calculate its area under curve (AUC) and root mean square error (RMSE). This is the benchmark that our models have to surpass (RMSE = 0.45, AUC = 0.5)

```
baseline_probability <- sum(framing.train$immigr_binary == 1)/nrow(framing.train)
pred.baseline <- rep(baseline_probability, nrow(framing.test))

# Calculating RMSE
( rmse.naive <- sqrt(mean((framing.test$immigr_binary - pred.baseline)^2)) )
```

```
## [1] 0.4550347
```

```
# Calculating Area under curve
auc(framing.test$immigr_binary, pred.baseline)
```

```
## Area under the curve: 0.5
```

Decision Tree model

We start by creating a decision tree model (with default parameters) for predicting negative attitude to immigration.

```
# Training classification decision tree
dt <- rpart(immigr_binary ~ ., data = framing.train[,features], method = "class")

# Predicting the instance of negative attitude to immigration
# first column - probability of 0 for each observation
# second column - probability of 1
pred.dt <-predict(dt, newdata = framing.test, type = "prob")[,2]


# Calculate performance with AUC
auc(framing.test$immigr_binary, pred.dt)
```

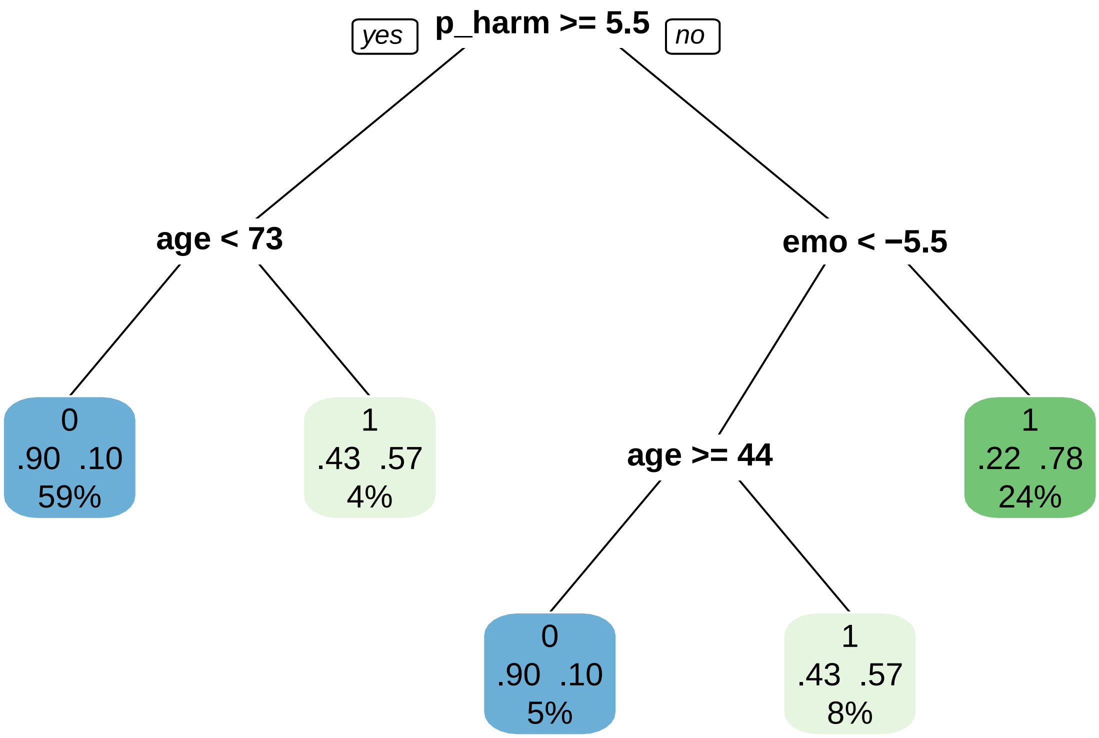
```
## Area under the curve: 0.7096
```

```
# Calculate performance with RMSE
( rmse.dt <- sqrt(mean((framing.test$immigr_binary - pred.dt)^2)) )
```

```
## [1] 0.4241035
```

Visualizing the decision tree:

```r
# Visualizing the results from "dt" using the prp() function
prp(dt, extra = 104, border.col = 0, box.palette="auto", roundint=FALSE) # Print the percentage of obse
```



Finding the best parameters for the decision tree

We loop over possible parameter values to find a combination that performs the best.

```r
parameter_values <- expand.grid("cp" = seq(0.00, 0.02, by = 0.005),
                                "minsplit" = seq(10, 50, by = 5))
num_folds <- 5


# Vector to store results (i.e., performance estimates per CV iteration)
cv_results <- matrix(nrow = nrow(parameter_values), ncol = num_folds)

# Create k folds of approximately equal size
folds <- cut(1:nrow(framing.train), breaks = num_folds, labels = F)


for (i in 1:num_folds) {

  print(paste0(i, "/", num_folds))

  idx_val <- which(folds == i)
  cv_train <- framing.train[-idx_val,]
  cv_valid <- framing.train[ idx_val,]

  for (j in 1:nrow(parameter_values)) {
    dt <- rpart(immigr_binary ~ ., data = cv_train[, features], method = "class",
```

```r
                 cp = parameter_values$cp[j],
                 minsplit = parameter_values$minsplit[j])

    pred.dt <- predict(dt, newdata = cv_valid, type = "prob")[,2]

    cv_results[j, i] <- auc(cv_valid$immigr_binary, pred.dt, quiet = T)
    }
}
```

```
## [1] "1/5"
## [1] "2/5"
## [1] "3/5"
## [1] "4/5"
## [1] "5/5"
```

We find the best combination based on average AUC. The winning parameters are a `cp` of 0 and minimum split of 25.

```r
parameter_values$mean_auc <- apply(cv_results, 1, mean)
parameter_values[order(parameter_values$mean_auc), ]
```

```
##       cp minsplit  mean_auc
## 1  0.000       10 0.7106810
## 2  0.005       10 0.7106810
## 3  0.010       10 0.7154429
## 6  0.000       15 0.7186508
## 7  0.005       15 0.7186508
## 8  0.010       15 0.7273810
## 9  0.015       15 0.7383762
## 10 0.020       15 0.7383762
## 11 0.000       20 0.7402257
## 12 0.005       20 0.7402257
## 4  0.015       10 0.7414294
## 5  0.020       10 0.7414294
## 13 0.010       20 0.7426067
## 14 0.015       20 0.7426067
## 15 0.020       20 0.7426067
## 41 0.000       50 0.7576000
## 42 0.005       50 0.7576000
## 43 0.010       50 0.7576000
## 44 0.015       50 0.7576000
## 45 0.020       50 0.7576000
## 18 0.010       25 0.7649427
## 19 0.015       25 0.7649427
## 20 0.020       25 0.7649427
## 21 0.000       30 0.7649427
## 22 0.005       30 0.7649427
## 23 0.010       30 0.7649427
## 24 0.015       30 0.7649427
## 25 0.020       30 0.7649427
## 26 0.000       35 0.7649427
## 27 0.005       35 0.7649427
## 28 0.010       35 0.7649427
## 29 0.015       35 0.7649427
## 30 0.020       35 0.7649427
```

```
## 31 0.000       40 0.7649427
## 32 0.005       40 0.7649427
## 33 0.010       40 0.7649427
## 34 0.015       40 0.7649427
## 35 0.020       40 0.7649427
## 36 0.000       45 0.7649427
## 37 0.005       45 0.7649427
## 38 0.010       45 0.7649427
## 39 0.015       45 0.7649427
## 40 0.020       45 0.7649427
## 16 0.000       25 0.7698378
## 17 0.005       25 0.7698378
```

```
parameter_values[which.max(parameter_values$mean_auc), ]
```

```
##    cp minsplit  mean_auc
## 16  0       25 0.7698378
```

Training the model with the chosen parameters:

```
dt2 <- rpart(immigr_binary ~ ., data = framing.train[, features], method = "class",
            cp = parameter_values$cp[which.max(parameter_values$mean_auc)],
            minsplit = parameter_values$minsplit[which.max(parameter_values$mean_auc)])

pred.dt2 <- predict(dt, newdata = framing.test, type = "prob")[,2]
```

Calculating AUC and RMSE:

```
auc(framing.test$immigr_binary, pred.dt2, quiet = T)
```

```
## Area under the curve: 0.7613
```

```
( rmse.dt <- sqrt(mean((framing.test$immigr_binary - pred.dt2)^2)) )
```

```
## [1] 0.409649
```

Area under curve increased from 0.71 (default parameters) to 0.76. RMSE decreased from 0.42 to 0.42. Setting optimal DT parameters lead to a modest increase in the effectiveness of the model.

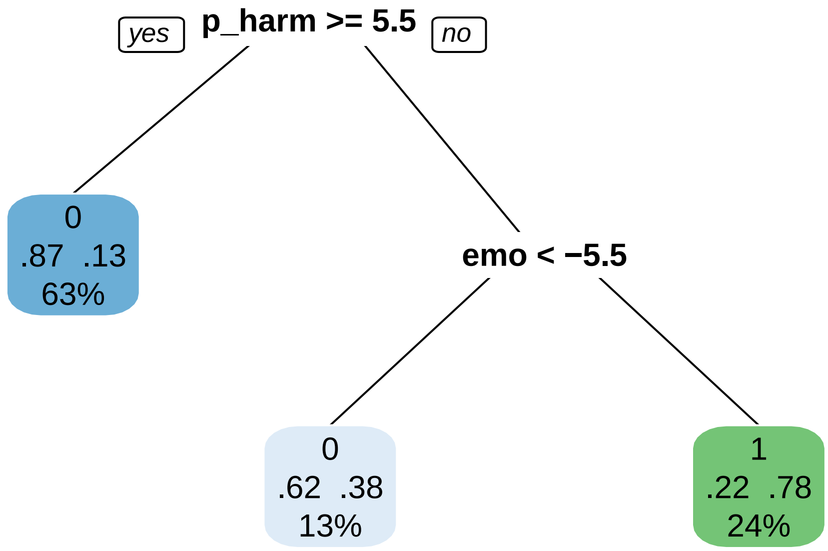
Listing factors by importance:

```
as.matrix(dt$variable.importance, ncol = 1)
```

```
##                 [,1]
## p_harm    16.8049456
## emo       11.2501809
## anx        7.3308538
## cong_mesg  3.0143074
## english    2.4662515
## educ       1.7950043
## cond       0.6372968
## age        0.4248646
```

Visualizing the final tree. It is actually much simpler, featuring only two branching conditions.

```
prp(dt2, extra = 104, border.col = 0, box.palette="auto", roundint=FALSE)
```



Logistic regression model

Installing required packages:

```r
# stargazer for nice tables
# install.packages("stargazer", repos = "http://cran.us.r-project.org")
library(stargazer)
```

Training the model:

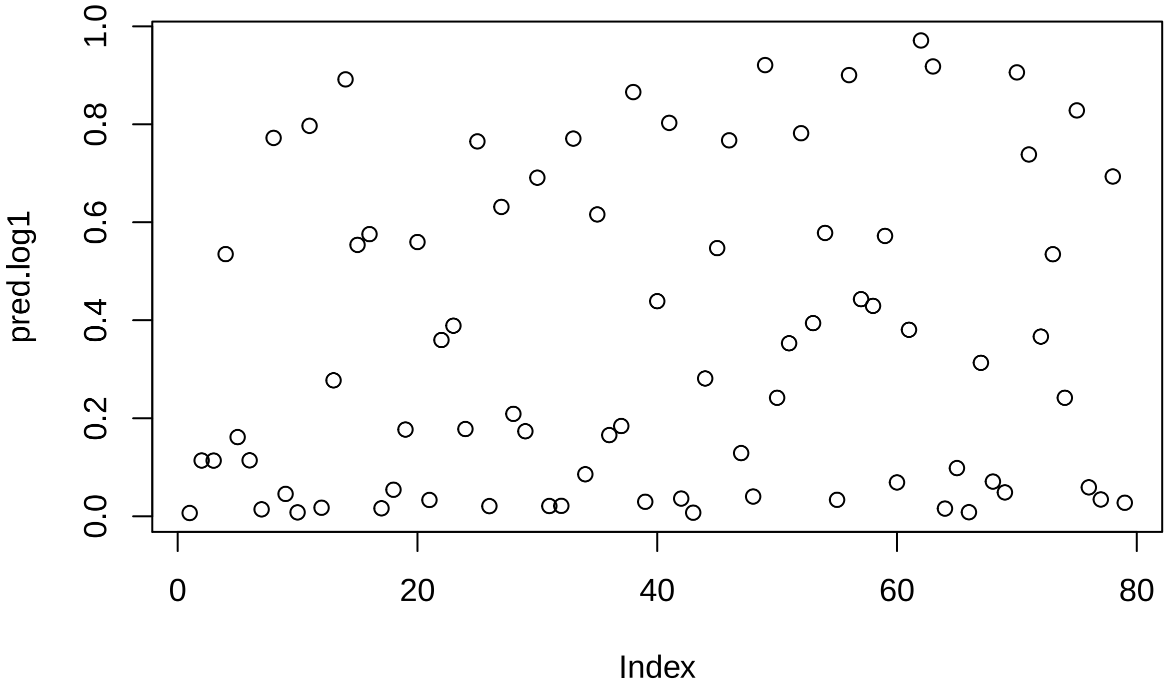
```r
log1 <- glm(immigr_binary ~ ., data = framing.train[, features],
        family = binomial(link = "logit"))


stargazer(log1, type = "text")
```

```
##
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                            immigr_binary
## -----------------------------------------------
## condNegativeEuropean            0.184
##                                (0.633)
##
## condPositiveLatino             1.086*
##                                (0.633)
##
## condPositiveEuropean            0.465
##                                (0.642)
##
## anx                            -0.363
##                                (0.489)
##
## age                            -0.0003
##                                (0.013)
```

```
##
## educ.L                         0.833
##                               (0.703)
##
## educ.Q                         0.019
##                               (0.560)
##
## educ.C                        -0.527
##                               (0.427)
##
## gendermale                    -0.520
##                               (0.431)
##
## income                        -0.068
##                               (0.056)
##
## emo                            0.277
##                               (0.197)
##
## p_harm                       -0.428**
##                               (0.174)
##
## toneNegative
##
##
## ethLatino
##
##
## english                       0.674**
##                               (0.290)
##
## anti_info                     -0.383
##                               (0.955)
##
## cong_mesg                     -0.869
##                               (0.610)
##
## Constant                       3.977
##                               (2.793)
##
## -----------------------------------------------
## Observations                    186
## Log Likelihood                -75.448
## Akaike Inf. Crit.             182.895
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```

Making predictions and visualizing them:

```r
pred.log1 <- predict(log1, newdata = framing.test, type = "response")
par(mfrow = c(1,1))
plot(pred.log1)
```



Evaluating the model using RMSE and AUC:

```
auc(framing.test$immigr_binary, pred.log1)
```

```
## Area under the curve: 0.8983
```

```
( rmse.log <- sqrt(mean((framing.test$immigr_binary - pred.log1)^2)) )
```

```
## [1] 0.3456151
```

The logistic regresion beats the decision tree model. (AUC: $0.89 > 0.76$, RMSE $0.35 < 0.41$)