Dating Research

Léon Yuan

Table of contents

Import data

Table 1: Speed Dating Data

iid	id	gender	idg	condtn	wave	round	position	positin1	order	partner	pid	match	int_cor
1	1	0	1	1	1	10	7	NA	4	1	11	0	0.1
1	1	0	1	1	1	10	7	NA	3	2	12	0	0.5
1	1	0	1	1	1	10	7	NA	10	3	13	1	0.1
1	1	0	1	1	1	10	7	NA	5	4	14	1	0.6
1	1	0	1	1	1	10	7	NA	7	5	15	1	0.2
1	1	0	1	1	1	10	7	NA	6	6	16	0	0.2

```
> library(tidyverse)
> library(kableExtra)
> Speed_Dating_Data <- read_csv("./data/Speed Dating Data.csv")
> head(Speed_Dating_Data) |>
+   kable(booktabs = TRUE,
+       caption = "Speed Dating Data") |>
+   kable_styling(latex_options="striped")
```

Explore data analysis

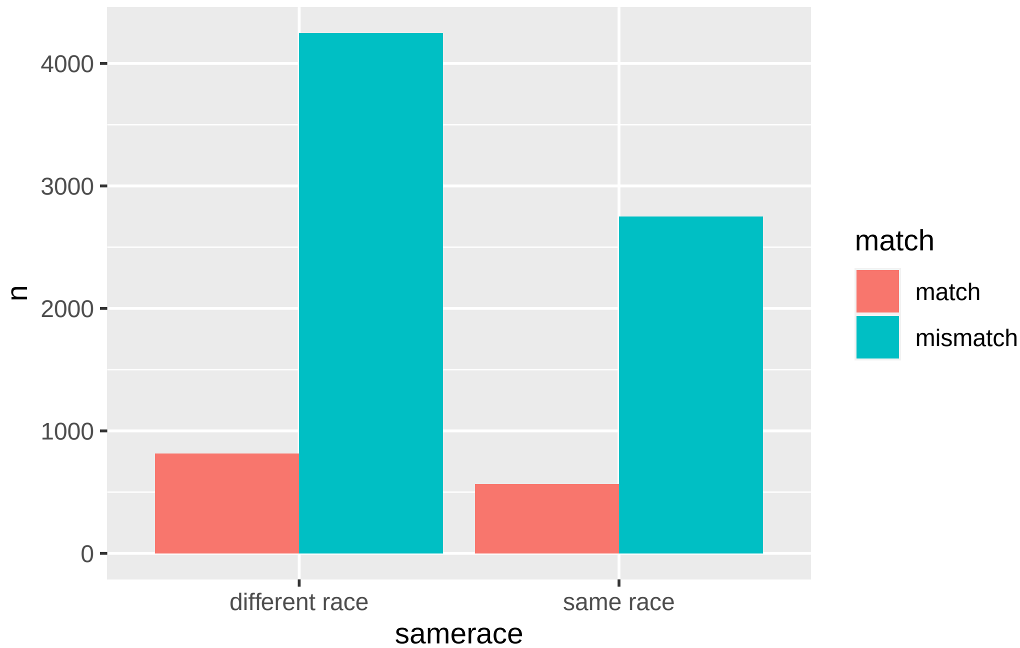
match and same race

From the below bar chart, I

```
> # Define race and match table
> race <- Speed_Dating_Data |>
+   count(match, samerace) |>
+   mutate(match = ifelse(match == 0, "mismatch", "match"),
+          samerace = ifelse(samerace == 0, "different race", "same race"))
> # Plot bar chart for race and match
> race |>
+   ggplot(aes(x = samerace, y = n, fill = match)) +
+   geom_bar(position="dodge", stat="identity")
```

Table 2: Race and Match Table

	same.race	different.race	Row.Prop
match	566	814	41
mismatch	2750	4248	39
Column Prop	17	16	NA




```
> race_table <- data.frame(
+   same.race = c(566, 2750, as.integer(566/(566+2750)*100)),
+   different.race = c(814, 4248, as.integer(814/(814+4248)*100)),
+   Row.Prop = c(as.integer(566/(566+814)*100),
+                as.integer(2750/(2750+4248)*100), NA))
> rownames(race_table) <- c("match", "mismatch", "Column Prop")
> race_table |>
+   kable(booktabs = TRUE,
+       caption = "Race and Match Table") |>
+   kable_styling(latex_options="striped")
```

Conduct a Chi-Sqaure test for independent test:

```
> M <- as.table(rbind(c(566, 814), c(2750, 4248)))
> dimnames(M) <- list(M = c("Match", "Mismatch"),
+                     R = c("Same Race", "Different Race"))
> chisq.test(M)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  M
X-squared = 1.351, df = 1, p-value = 0.2451
```

Conduct a Fisher's Exact test for independent test:

```
> M <- as.table(rbind(c(566, 814), c(2750, 4248)))
> dimnames(M) <- list(M = c("Match", "Mismatch"),
+                     R = c("Same Race", "Different Race"))
> fisher.test(M)
```

```
    Fisher's Exact Test for Count Data

data:  M
p-value = 0.2402
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.9531003 1.2099161
sample estimates:
odds ratio
  1.074088
```

From these two tests, the p-value is about 0.2 which is larger than any significant level, I failed to reject the null hypothesis, thus overall there is no relationship between race and match. However, this is an overall conclusion for all gender and races which may be misleading to ignore gender difference. Next I will investigate further for race and match between male and female.

Is female racial preference the same as male's?

First I visualize the difference by gender as following:

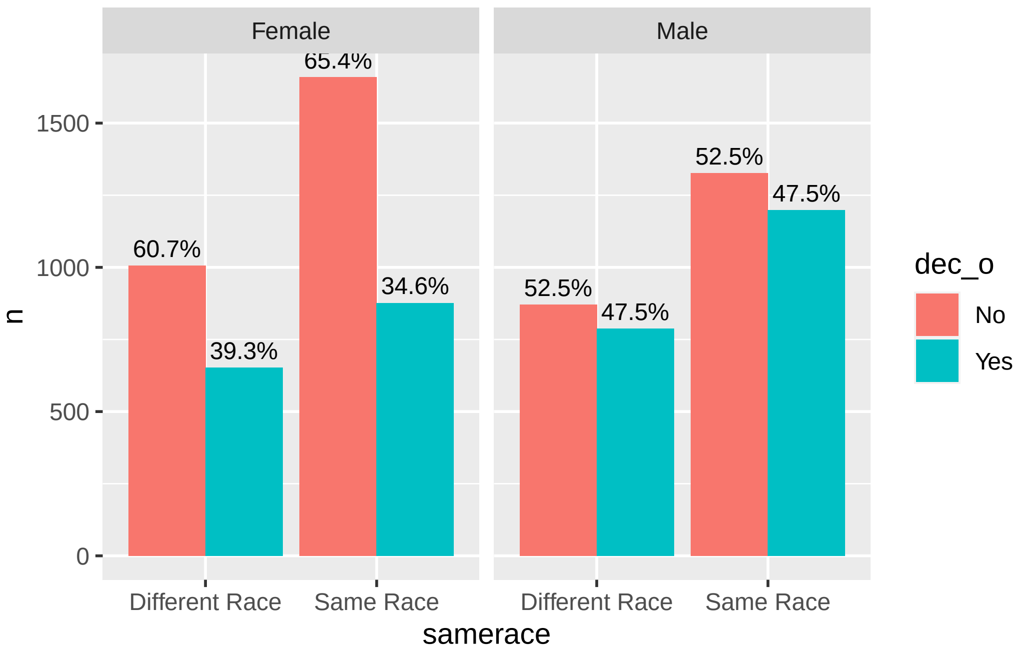
```
> library(scales)
```

```
Attaching package: 'scales'

The following object is masked from 'package:purrr':

    discard

The following object is masked from 'package:readr':

    col_factor
```

```
> Speed_Dating_Data |>
+   mutate(part_gender = ifelse(gender == 0, 1, 0)) |>
+   count(part_gender, dec_o, samerace) |>
+   mutate(part_gender = ifelse(part_gender == 0, "Female", "Male"),
+          dec_o = ifelse(dec_o == 0, "No", "Yes"),
+          samerace = ifelse(samerace == 0, "Same Race", "Different Race")) |>
+   group_by(part_gender, samerace) |>
+   mutate(prop = n / sum(n)) |>
+   ggplot(aes(x = samerace, y = n, fill = dec_o,
+              label = percent(prop, accuracy = 0.1))) +
+   geom_bar(position="dodge", stat="identity") +
+   geom_text(position = position_dodge(width = .9),    # move to center of bars
+               vjust = -0.5,    # nudge above top of bar
+               size = 3) +
+   facet_wrap(~part_gender)
```

Table 3: Race and Match Table

	Same Race.Femal	Difference Race.Femal	Same Race.Male	Difference Race.Male
Yes	877	652	1199	787
No	1659	1006	1327	871



Then I am going to conduct a Mantel-Haenszel chi-squared test as following: First I build an array for this test:

```
> column.names <- c("Same Race", "Difference Race")
> row.names <- c("Yes", "No")
> matrix.names <- c("Femal", "Male")
> gds <- array(data = c(877,1659,652,1006,1199,1327,787,871),
+              dim = c(2,2,2),
+              dimnames = list(row.names, column.names, matrix.names))
> gds |>
+   kable(booktabs = TRUE,
+       caption = "Race and Match Table") |>
+   kable_styling(latex_options="striped")
```

After conducting this test, then I found that p-value is 0.0321 which is slightly less than 0.05, thus I can tell that the odds ratio is not equal to 1 by gender. Obviously, from the above ggplot graph, female says less “yes” to interracial dating than male does.

```
> mantelhaen.test(gds)
```

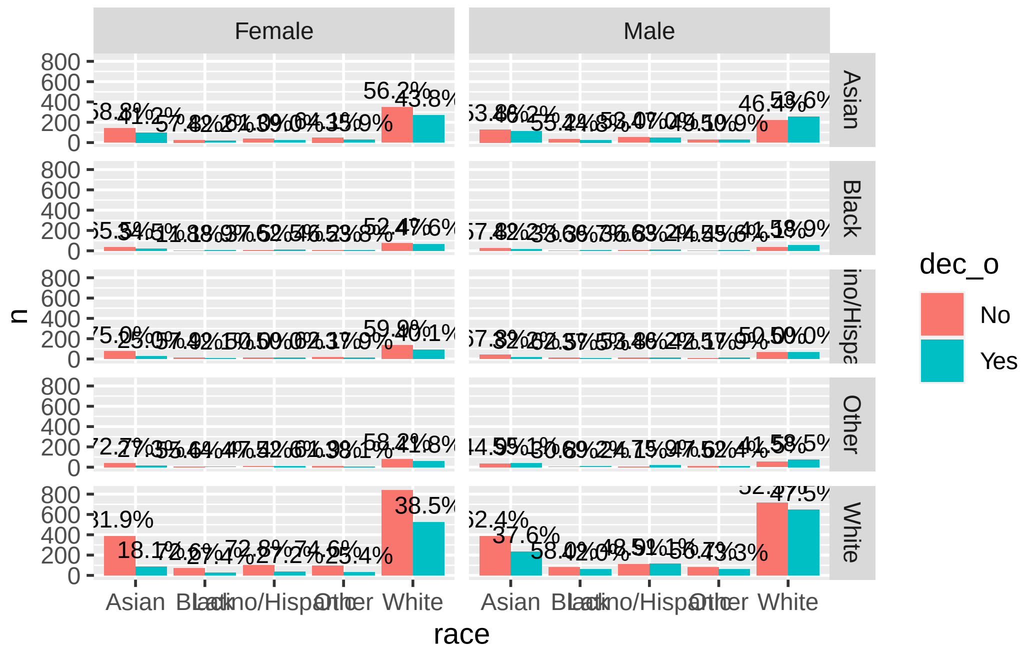
```

	Mantel-Haenszel chi-squared test with continuity correction

data:  gds
Mantel-Haenszel X-squared = 4.5931, df = 1, p-value = 0.0321
alternative hypothesis: true common odds ratio is not equal to 1
95 percent confidence interval:
 0.8290474 0.9907672
sample estimates:
common odds ratio
        0.9063073
```

Difference combination of race and gender in dating preference with Data Visualization

From the below graph, we can tell that Asian males are highest rejected when they date all races' female comparing to other male races, especially Asian male is extremely likely to be rejected by white females, which is also the highest rejection rate in all dating combinations.

```
> Speed_Dating_Data |>
+   drop_na(race_o, race) |>
+   mutate(part_gender = ifelse(gender == 0, 1, 0)) |>
+   count(part_gender, race_o, race, dec_o) |>
+   mutate(part_gender = ifelse(part_gender == 0, "Female", "Male"),
+          dec_o = ifelse(dec_o == 0, "No", "Yes"),
+          race_o = case_when(race_o == 1 ~ "Black",
+                             race_o == 2 ~ "White",
+                             race_o == 3 ~ "Latino/Hispanic",
+                             race_o == 4 ~ "Asian",
+                             race_o == 5 ~ "Native American",
+                             race_o == 6 ~ "Other"),
+          race = case_when(  race == 1 ~ "Black",
+                             race == 2 ~ "White",
+                             race == 3 ~ "Latino/Hispanic",
+                             race == 4 ~ "Asian",
+                             race == 5 ~ "Native American",
+                             race == 6 ~ "Other")) |>
```

```
+   group_by(part_gender, race_o, race) |>
+   mutate(prop = n / sum(n)) |>
+   ggplot(aes(x = race, y = n, fill = dec_o,
+              label = percent(prop, accuracy = 0.1))) +
+   geom_bar(position="dodge", stat="identity") +
+   geom_text(position = position_dodge(width = .9),
+             vjust = -0.5,
+             size = 3) +
+   facet_grid(cols = vars(part_gender), rows = vars(race_o))
```



Conduct logistic regressions separately for male and female

The reason I build two separate models for females and males is because there are some big differences in dating behaviors between genders and separate models are easier to interpreter.

Decisons made by females to male when dating

```
> # Filter data when females date males
> females_to_males <- Speed_Dating_Data |>
```

```
+   filter(gender == 1) |>
+   select(dec_o, samerace, race_o, age_o, attr_o, sinc_o, intel_o, fun_o, amb_o, shar_o,
+          age, race)
> # Convert numerical decision into factor type
> females_to_males$dec_o <- factor(females_to_males$dec_o,
+                                  levels = c(0,1),
+                                  labels = c("No", "Yes"))
> # Make the glm predict the Yes as 1
> contrasts(females_to_males$dec_o)
```

```
    Yes
No    0
Yes   1
```

```
> females_to_males$samerace <- factor(females_to_males$samerace,
+                                  levels = c(0,1),
+                                  labels = c("no", "yes"))
> contrasts(females_to_males$samerace)
```

```
    yes
no    0
yes   1
```

```
> females_to_males$race <- factor(females_to_males$race,
+                              levels = 1:6,
+                              labels = c("Black","White","Latino","Asian","Native","Othe
> contrasts(females_to_males$race)
```

```
       White Latino Asian Native Other
Black      0      0     0      0     0
White      1      0     0      0     0
Latino     0      1     0      0     0
Asian      0      0     1      0     0
Native     0      0     0      1     0
Other      0      0     0      0     1
```

```
> # delete the missing value rows
> females_to_males <- females_to_males |>
+   drop_na()
```

Logistic regression only on race

First I only care about how race affects females' decisions to males, only including the `samerace` and `race` columns in this logistic classification model. From the summary of model, we can tell that all females are likely to reject the `Asian` males because `Asian` males has 0.008 p-value which is the most significant in this model. The log odd of saying "yes" to Asian males by all females is -0.48977 given other variables fixed and this is significant negative coefficient meaning that Asian males are very unpopular when dating. Thus, the odd ratio of say "yes" to Asian males is $e^{-0.48977} = 0.6127673$ which means when females date Asian males, they likely decrease 40% probability of saying "yes" to Asian males. Also, `samerace` doesn't show statistical significance because of relatively large p-value 0.06 which is counter intuitive to common sense that females are preferred same race dating.

```
> fit <- glm(data = females_to_males,
+     formula = dec_o ~ samerace+race,
+     family=binomial(link='logit'))
> summary(fit)
```

```
Call:
glm(formula = dec_o ~ samerace + race, family = binomial(link = "logit"),
    data = females_to_males)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0617  -1.0044  -0.7908   1.2976   1.6216

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.51241    0.16640  -3.079  0.00207 **
sameraceyes  0.14333    0.07829   1.831  0.06715 .
raceWhite    0.09080    0.17639   0.515  0.60671
raceLatino  -0.08731    0.21892  -0.399  0.69003
raceAsian   -0.48977    0.18588  -2.635  0.00842 **
raceOther   -0.17208    0.21686  -0.794  0.42748
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4466.4  on 3380  degrees of freedom
Residual deviance: 4415.1  on 3375  degrees of freedom
```

```
AIC: 4427.1

Number of Fisher Scoring iterations: 4
```

Then the `ANOVA` table shows that `race` variable has very small p-value which shows it is very significant as I said before.

```
> anova(fit, test="Chisq")
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: dec_o

Terms added sequentially (first to last)

         Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                      3380     4466.4
samerace  1   11.952      3379     4454.5 0.0005459 ***
race      4   39.378      3375     4415.1  5.82e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Logistic regression includes the race and six attributes

Once I included six attribute scores and race together in the logistic regression model, the significance level of race changes radically, because none of race is significant once six attributes are included. This suggests that males' personal attributes can overturn/change the females' impressions or decisions deeply. As we can see from the p-values, all six attributes are statistically significant, especially physical attractiveness, fun, ambitious, shared interests play major roles in making decisions.

The coefficient of *physical attractiveness* is *0.39356*, this means log odds of saying "yes" to males by females increases *0.39356* givens other variables fixed, and odd ratios of saying "yes" to males by females increases $e^{0.39356} = 1.482248$ when one more score is given to *attractiveness*.

The coefficient of *fun* is *0.27850*, this means log odds of saying "yes" to males by females increases *0.27850* givens other variables fixed, and odd ratios of saying "yes" to males by females increases $e^{0.27850} = 1.321147$ when one more score is given to *fun*.

The coefficient of *shared interests* is *0.27081*, this means log odds of saying "yes" to males by females increases *0.27081* givens other variables fixed, and odd ratios of saying "yes" to males by females increases $e^{0.27081} = 1.311026$ when one more score is given to *shared interests*.

All these three most significant attributes have positive coefficient meaning that more scores on these attributes will help females a lot make "yes" decisions to males.

```
> fit1 <- glm(data = females_to_males,
+     formula = dec_o ~ samerace+race+attr_o+sinc_o+intel_o+fun_o+amb_o+shar_o,
+     family=binomial(link='logit'))
> summary(fit1)
```

```
Call:
glm(formula = dec_o ~ samerace + race + attr_o + sinc_o + intel_o +
    fun_o + amb_o + shar_o, family = binomial(link = "logit"),
    data = females_to_males)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2882  -0.8258  -0.3876   0.8537   3.1893

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.77842    0.33291 -17.358  < 2e-16 ***
sameraceyes -0.06528    0.09255  -0.705  0.48059
raceWhite    0.33768    0.20007   1.688  0.09145 .
raceLatino   0.04752    0.24955   0.190  0.84899
raceAsian    0.11319    0.21205   0.534  0.59348
raceOther    0.08152    0.24908   0.327  0.74346
attr_o       0.39356    0.02991  13.160  < 2e-16 ***
sinc_o      -0.08210    0.03535  -2.323  0.02019 *
intel_o      0.12259    0.04534   2.704  0.00685 **
fun_o        0.27850    0.03397   8.198 2.45e-16 ***
amb_o       -0.15945    0.03480  -4.582 4.61e-06 ***
shar_o       0.27081    0.02705  10.011  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4466.4  on 3380  degrees of freedom
Residual deviance: 3399.2  on 3369  degrees of freedom
```

```
AIC: 3423.2

Number of Fisher Scoring iterations: 5
```

The `ANOVA` table also shows that attractiveness, fun and shared interests explain the most deviance residuals by 710.98, 155.36, 106.23 compared to other variables' explained variations which are consistent with our above finding.

```
> anova(fit1, test="Chisq")
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: dec_o

Terms added sequentially (first to last)

         Df Deviance Resid. Df Resid. Dev  Pr(>Chi)    
NULL                      3380     4466.4              
samerace  1    11.95      3379     4454.5 0.0005459 ***
race      4    39.38      3375     4415.1  5.82e-08 ***
attr_o    1   710.98      3374     3704.1 < 2.2e-16 ***
sinc_o    1    14.36      3373     3689.7 0.0001508 ***
intel_o   1    16.84      3372     3672.9  4.06e-05 ***
fun_o     1   155.36      3371     3517.5 < 2.2e-16 ***
amb_o     1    12.13      3370     3505.4 0.0004952 ***
shar_o    1   106.23      3369     3399.2 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Random Forests for females' decisions on more variables

```
> # Filter data when females date males with more variables than logistics regression
> tree_females_to_males <- Speed_Dating_Data |>
+   filter(gender == 1) |>
+   select(dec_o,
+          samerace,
+          attr_o, sinc_o, intel_o, fun_o, amb_o, shar_o,
+          int_corr, age, race, field, from)
```

```r
> # Convert numerical decision into factor type
> tree_females_to_males$dec_o <- factor(tree_females_to_males$dec_o,
+                                       levels = c(0,1),
+                                       labels = c("No", "Yes"))
> # Make the glm predict the Yes as 1
> contrasts(tree_females_to_males$dec_o)
```

```
    Yes
No    0
Yes   1
```

```r
> tree_females_to_males$samerace <- factor(tree_females_to_males$samerace,
+                                       levels = c(0,1),
+                                       labels = c("no", "yes"))
> contrasts(tree_females_to_males$samerace)
```

```
    yes
no    0
yes   1
```

```r
> tree_females_to_males$race <- factor(tree_females_to_males$race,
+                                   levels = 1:6,
+                                   labels = c("Black","White","Latino","Asian","Native","Othe
> contrasts(tree_females_to_males$race)
```

```
       White Latino Asian Native Other
Black      0      0     0      0     0
White      1      0     0      0     0
Latino     0      1     0      0     0
Asian      0      0     1      0     0
Native     0      0     0      1     0
Other      0      0     0      0     1
```

```r
> # Drop missing rows
> tree_females_to_males <-
+   tree_females_to_males |>
+   drop_na()
```

Now I checked if the response variable `dec_o` is balanced or not. The ratio of No to Yes is 1.68 which shows relative balanced within the accepted range from 0.5 to 2. Thus, I don't need to make any efforts to balance the dataset.

```
> table(tree_females_to_males$dec_o)
```

```
  No  Yes
2125 1259
```

Initially I did want to include `income` variable in the random forest, however, I found there are half of income variables missing, so I have to drop this variable.

```
> sum(is.na(Speed_Dating_Data$income))
```

```
[1] 4099
```

Build random forests model for it

Load packages

```
> library(randomForest)
```

```
randomForest 4.7-1.1

Type rfNews() to see new features/changes/bug fixes.


Attaching package: 'randomForest'

The following object is masked from 'package:dplyr':

    combine

The following object is masked from 'package:ggplot2':

    margin
```

```
> library(datasets)
> library(caret)
```

```
Loading required package: lattice


Attaching package: 'caret'

The following object is masked from 'package:purrr':

    lift
```

```r
> library(pROC)
```

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


Attaching package: 'pROC'

The following objects are masked from 'package:stats':

    cov, smooth, var
```

```r
> library(glmnet)
```

```
Loading required package: Matrix


Attaching package: 'Matrix'

The following objects are masked from 'package:tidyr':

    expand, pack, unpack

Loaded glmnet 4.1-6
```

Split the data into training and test set

I randomly splited data into 80% training and 20% test set.

```
> set.seed(222)
> ind <- sample(2, nrow(tree_females_to_males), replace = TRUE, prob = c(0.8, 0.2))
> train <- tree_females_to_males[ind==1,]
> test <- tree_females_to_males[ind==2,]
```

Check how many observations in training and how many in test set. There are 3366 rows in training and 826 in the test set.

```
> dim(train)
```

```
[1] 2717   13
```

```
> dim(test)
```

```
[1] 667  13
```

Construct a random forest model for this training data

I chose 500 trees and 4 random predictors at each split.

```
> rf <- randomForest(x = train[-1],
+                    y = train$dec_o,
+                    xtest = test[-1],
+                    ytest = test$dec_o,
+                    ntree = 500,
+                    mtry = 4,
+                    proximity = TRUE)
```

Print out the random forests. The OOB estimate of error rate is 24.99% which has 75% accuracy on the training set while on the test set, this RF has roughly 73% test accuracy which is not bad on this dating data.

```
> print(rf)
```

```
Call:
 randomForest(x = train[-1], y = train$dec_o, xtest = test[-1],      ytest = test$dec_o, ntre
               Type of random forest: classification
```

```
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 25.43%
Confusion matrix:
      No Yes class.error
No  1446 268   0.1563594
Yes  423 580   0.4217348
                Test set error rate: 26.24%
Confusion matrix:
     No Yes class.error
No  350  61   0.1484185
Yes 114 142   0.4453125
```

Confusion matrix

Print out the confusion matrix and other statistical measures on this classification results. The whole accuracy on the test set is 72.71%. The true "Yes" rate is $138/(138+118) = 53.90$ which is a little bit over 50% random guess rate. However, the true "No" rate is $347/(347+64) = 84.43$ which is a better prediction rate on the test set because training set has more "No" classes than "Yes".

```
> confusionMatrix(data = rf$test$predicted,
+                 reference = test$dec_o)
```

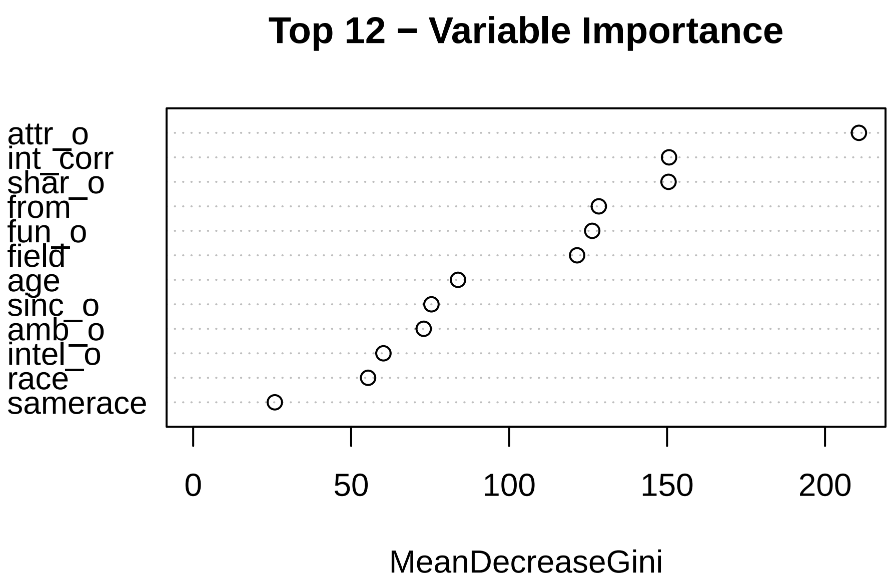
```
Confusion Matrix and Statistics

          Reference
Prediction  No Yes
       No  350 114
       Yes  61 142

               Accuracy : 0.7376
                 95% CI : (0.7025, 0.7707)
    No Information Rate : 0.6162
    P-Value [Acc > NIR] : 2.398e-11

                  Kappa : 0.4228

 Mcnemar's Test P-Value : 8.465e-05
```

```
            Sensitivity : 0.8516
            Specificity : 0.5547
         Pos Pred Value : 0.7543
         Neg Pred Value : 0.6995
             Prevalence : 0.6162
         Detection Rate : 0.5247
   Detection Prevalence : 0.6957
      Balanced Accuracy : 0.7031

       'Positive' Class : No
```

Variable Importance

From the variable importance plot, I roughly classify the top 6 predictors into three classes. The first top class only has one predictor, physical attractiveness. This is consistent with my logistic regression. *Physical Attractiveness* has 205.911 mean decrease of Gini that is measurement of building trees. This Gini decrease is almost twice of other variables. Thus, *Physical attractiveness* is the most significant factor when females make decision to males. The second top class has *shared interests* and the correlation between participant's and partner's ratings of interests. These two variables actually are highly correlated however random forest is robust to the highly correlated predictors because of its ability of randomly selecting a subset of variables at each split. *Shared Interests* has 150.324 decrease Gini of mean which is as three times as other less importance variables. Females secondary emphasize shared interests with males. The third top class has three variables: *fun, from, field*. They have very close Gini decrease mean about 125 which is as twice as other less important variables. Females put equally emphasis on the fun, where males are from, and which field males' careers belong to. Overall females are likely to date males who are very physical attractive then have common/shared interests as they do while males' career fields and where they're from play a secondary role in dating.

```r
> varImpPlot(rf,
+            sort = T,
+            n.var = 12,
+            main = "Top 12 - Variable Importance")
```




```
> importance(rf)
```

```
          MeanDecreaseGini
samerace          25.83871
attr_o           210.72323
sinc_o            75.41528
intel_o           60.21535
fun_o            126.30369
amb_o             72.96617
shar_o           150.46322
int_corr         150.62441
age               83.82029
race              55.36472
field            121.52390
from             128.39067
```

Receiver Operating Characteristic comparing random forests with logistics regression on the same train and test set

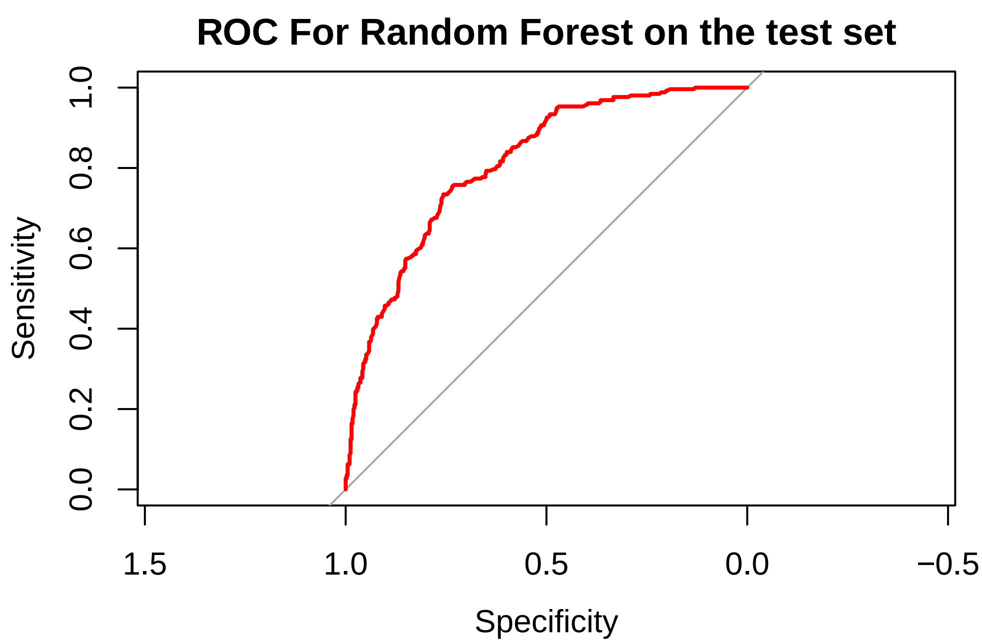
First I built the same random forest model and plot the ROC curve:

```
> # Build a random forest model
> rf <- randomForest(x = train[-1],
+                    y = train$dec_o,
```

```
+                    ntree = 500,
+                    mtry = 4,
+                    proximity = TRUE)
> # Make "Yes" as positive classes
> pred_roc <- predict(rf, newdata = test, type = "prob")[,2]
> ROC_rf <- roc(response = test$dec_o,
+               predictor = pred_roc)
```

```
Setting levels: control = No, case = Yes

Setting direction: controls < cases
```

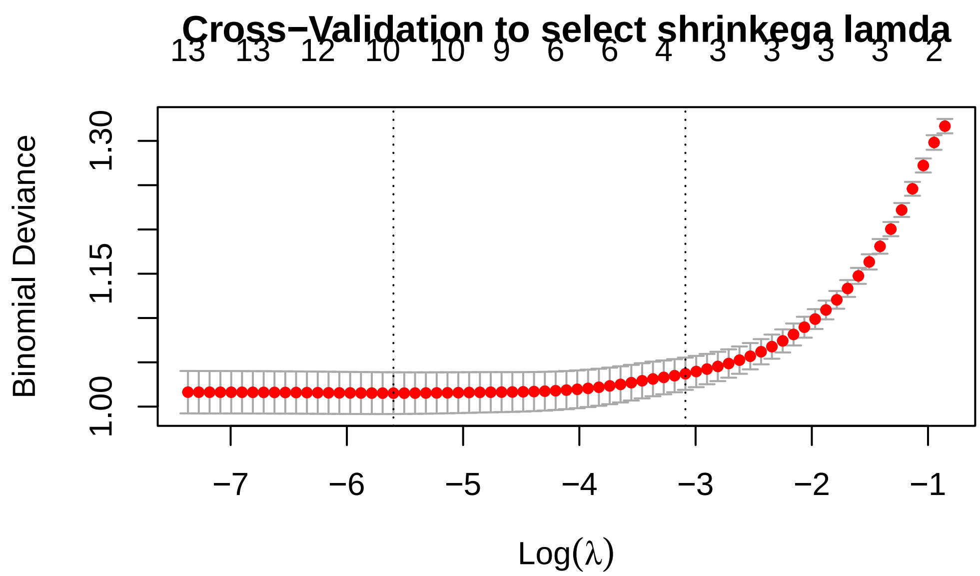
```
> plot(ROC_rf, col = "red", main = "ROC For Random Forest on the test set")
```



Second I built a Penalized Logistic Regression because there are many predictors, thus selecting and shrinking variables are necessary to build a good logistic regression.

First I encoded train data frame into a form of dummy variables for all categorical variables and I deleted the `from` column because it has 164 unique values which produces huge number of variables and also `field`. Then I use Elastic net with logistics regression on this train matrix with alpha = 0.5. I build a final model with the lambda that gives the simplest model but also lies within one standard error of the optimal lambda selected by cross validation measured by Binomial Deviance. Finally I plot the cross-validation plot when selecting lambda.

```
> set.seed(123)
> # Encode matrix into dummy variable forms for all categorical variables
> x.train <- model.matrix(dec_o~., train[,-c(12,13)])[,-1]
> # Use Elastic net with logistics regression on this train dataset with alpha = 0.5
> cv.elastic <- cv.glmnet(x = x.train, y = train$dec_o,
+                         alpha = 0.5, family = "binomial")
> # Build a final model with the best lambda selected by cross validation measured by Binomia
> best_elastic <- glmnet(x.train, y = train$dec_o, alpha = 0.5, family = "binomial",
+                  lambda = cv.elastic$lambda.1se)
> plot(cv.elastic, main = "Cross-Validation to select shrinkega lamda")
```



Next I make predictions on the test set by this best Elastic net model as following:

```
> x.test <- model.matrix(dec_o ~., test[,-c(12,13)])[,-1]
> prob_elastic <- predict(best_elastic, newx = x.test, type = "response")
```

From this plot, I can tell that Random Forest and Penalized Logistic Regression perform almost equally while Penalized Logistic regression performs slightly better than Random Forest.

```
> ROC_lr <- roc(response = test$dec_o,
+               predictor = prob_elastic)
```

```
Setting levels: control = No, case = Yes
```

```
Warning in roc.default(response = test$dec_o, predictor = prob_elastic):
Deprecated use a matrix as predictor. Unexpected results may be produced,
please pass a numeric vector.
```

```
Setting direction: controls < cases
```

```r
> plot(ROC_rf, col = "red", main = "Compare ROC of Random Forest and Penalized Logistic Regr
> lines(ROC_lr, col = "blue")
```