# Predicting the Success Rate of Speed Dating

(MS&E226 Project Part 1)

Cindy Kang, Esha Wang

## I. Introduction

WITH the onset of today's busy world, more and more people have been turning to alternative ways to traditional dating. Online dating and speed dating give us an opportunity to explore human interactions, and the prevalence of research in this area has allowed for data to be widely available. With such a rich database, as well as our own curiosity in human behavior, a few questions naturally arose – What traits do individuals look for in a partner? How do males and females differ in their priorities? What ultimately decides whether or not two people will match? In this study, we hope to explore what individuals look for in a partner, and how this translates to the success rate of a second date. Specifically, given a pair of individuals from the test set, we aim to predict the "like" score (continuous 1-10 rating) and "decision" (categorical yes/no answer to "Would you like to see this person again?") of that pair.

## II. The Dataset

Our dataset [1] consists of 8378 rows and 195 columns which summarize the results of a speed dating experiment conducted by Columbia Business School from 2002 to 2004. Each subject's objective attributes, such as age, gender, race, hometown, etc., were collected and recorded as columns in this database. Additionally, upon signing up for the speed dating event, each participant was asked to rank six personality traits (Attractive, Sincere, Intelligent, Fun, Ambitious, Has shared interests/hobbies) on a scale from 1 to 10, according to what he/she looks for in the opposite sex. After every 4-minute "speed date", each person was asked to rate their partner on a scale from 1 to 10 on those same six traits. Each row of the dataset represents a one-way pairing of two participants in the speed dating event. That is, for a speed date between Person A and Person B, one row of the dataset holds Person A's survey answers and ratings of Person B's personality traits, and another row of the dataset holds Person B's survey answers and ratings of Person A's attributes.

## III. Data Cleaning

The documentation for our dataset was not as organized as we would have liked and there were a few columns with very vague descriptions and seemingly no relevance to our topic of study. We have omitted these columns. For purposes of our study, we modified our dataset such that each row, which represents a one-way pairing, contains 7 objective compatibility factors (age difference, same race, income difference, interest correlation, difference in median SAT score of undergraduate institution, differences in goal of attending speed dating event, and differences in frequency of going out) and the six personality trait rankings of their partner, as mentioned above.

In addition, we note that there are a considerable number of NA's in our dataset. In order to ensure that we are working with sufficient data, we have decided to omit rows such that more than 50% of their column entries are missing. Tables 1 and 2 in the appendix contain information on the (relative) frequency of NA values per each column included in our final dataset. In particular, we note that two attributes (difference in median SAT scores of undergraduate institutions and difference in income) are unpopulated for over 50% of our sample. Generally, we would have discarded these columns, but we believe that at least some objective measure of one's intelligence and ambition is needed, since both are included in the six attributes that each participant rated his/her partner on.

## IV. Data Exploration

To visualize the dataset, we plotted a few preliminary graphs to get an idea of what our data looked like. In Figure 1, we plotted six histograms, displaying the univariate distributions of each of the six attributes from the survey, grouped by male and female ratings.

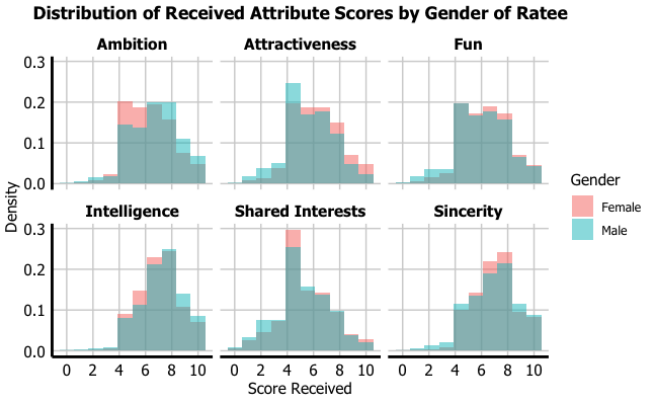


Fig. 1: Score distribution for each of the six pairs attributes

After omitting some columns that we deemed were repetitive or irrelevant, we were left with sixteen columns, and we wanted to determine which columns were highly correlated with one another as well as which were highly correlated with the proposed response variables ("like" score and decision). To methodically determine which columns were worth keeping, we found the correlation matrix for the table of attribute ratings, grouped by male and female raters. (Note that the calculations for a correlation matrix require rows of non-NA values. Therefore, for this experiment, we have omitted those rows that have NA values. In future experiments, we hope to impute missing values rather than simply omitting them.)

From Figure 2, we see that the correlation matrices for males and females seem to look roughly the same. This implies that both males and females generally look for the same traits (attractiveness and fun in particular) when searching for a partner. It appears that the attractiveness, fun, and shared interests attributes are most positively correlated with a higher

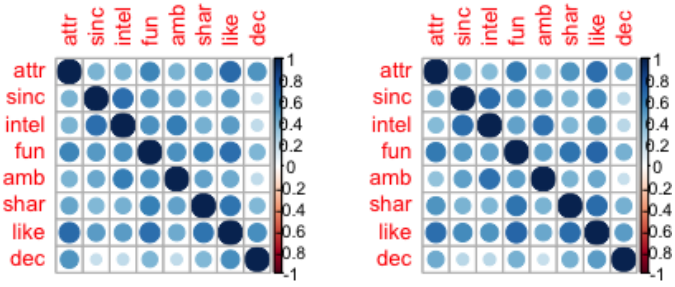


Fig. 2: Correlation matrix of six attributes for males (left) and females (right)

"like" score, while the attractiveness and shared interests attributes are most significant in giving a "yes" decision. Overall, from the correlation models, we found that there were no two columns that were overly correlated with each other. Therefore, we believe that each of the columns that we have chosen have the potential to improve our predictive model.

Our dataset also contains a large number of categorical variables, where each categorical variable can have many types of responses (i.e. race). In order to include these variables into our model, we must define them as factors in R. In turn, R will essentially create additional binary covariate terms to account for the different values that the factor can take. This may become quite computationally expensive as we further our analysis. If so, we will adjust our method accordingly.

We were interested in exploring was if there existed a relationship between positive response rate and average attribute score received. For each participant, let us define positive response rate to be:

$$\text{PRR} = \frac{\text{Number of partners who wanted a second date}}{\text{Number of partners met}}$$

For a particular attribute, let us define the average attribute score to be:

$$\text{AAS} = \text{mean}(\text{Attribute score})$$

In Figure 3, we plotted Positive Response Rate versus Average Score for each of the six subjective attributes. Attractiveness is, by far, most representative of a linear model with $R^2 = 0.592$. Meanwhile, ambition, intelligence, and sincerity have $R^2$ values of nearly zero. This seems to imply that individuals value attractiveness above all else, while other traits did not have much impact. We found these results to be rather surprising.

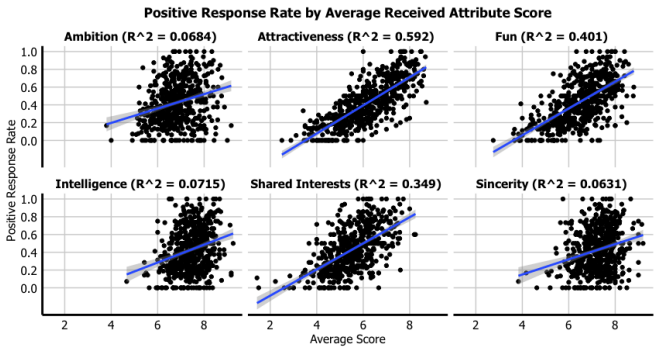


Fig. 3: Positive Response Rate vs. Received Attribute Score

In Figure 4, we plotted each individual's average "like" score versus the difference of that attribute. For example, we would plot a person's "like" score versus the difference in age between that person and his/her partner. From the results, it appears that there is little to no correlation between the objective compatibility factors and the "like" score. We most definitely found this result surprising, since it implies that even factors such as interest correlation have negligible impact on how an individual ranks his or her partner's "like" score. But coupled with the significant positive correlation between the "Shared Interests" rating and the positive response rate from Figure 3, we note that the perception of another person's interests, intelligence, and other attributes is a stronger factor in determining likeability and compatibility than the truth. It is also highly possible that the 4 minutes given to each couple during speed dating was not enough to truly gauge their interests and/or other attributes. This caveat of our study would prevent us from extrapolating our results to dating situations outside of speed dating, where couples are not limited in the amount of time they have to get to know each other.

So far, we have observed the impact of subjective ratings and objective compatibility factors on the likeability and compatibility of a couple. In Figure 5, we explore the correlations between the objective compatibility factors of a couple and the partner's rating of attributes. It seems likely that a person's SAT score would be positively associated with their partner's rating of their intelligence level. However, from the plots, we note that the correlations between objective attributes and the partner's ratings are quite low, with the highest $R^2$ value being that of going-out frequency and "fun-ness" (with $R^2 = 0.12$).

With these observations in mind, we suggest that our population models of likeability and compatibility are likely linear combinations of the objective compatibility factors of a pairing along with the partner's rating of their attributes. Due to the low correlations between the objective factors and the corresponding ratings by the partner, we are doubtful that interaction terms will be relevant since it appears to be perceptions of a partner's attributes that matter more than the objective truths. However, during the model development stage, we can experiment with interaction terms of the more highly correlated factors and determine their relevance by the significance of their coefficients.

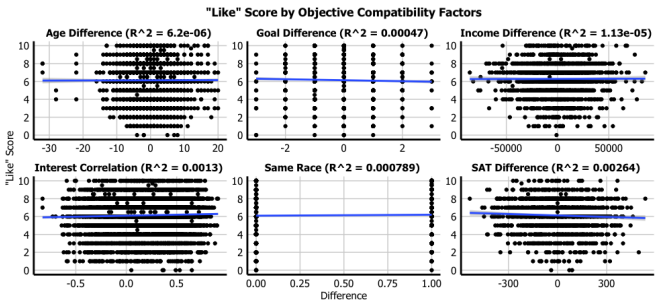


Fig. 4: Like Score vs. Objective Attributes

## V. Future Work

As noted in previous sections, many of the entries in our dataset consist of NA's. Currently we are omiting rows that have NA, however we hope to compute an estimate for those NA values (perhaps using KNN) in order to use them in our models. Furthermore, we hope to implement machine learning algorithms such as Ridge Regression and/or Lasso, logisitic regression, and Naive Bayes to help predict our target variables "like" and "decision".

# VI. Appendix

TABLE I: Number of NA Values for Objective Compatibility Factors

| | Age Diff | Same Race | Income Diff | SAT Diff | Goal Diff | Go Out Diff | Interest Corr |
|---|---|---|---|---|---|---|---|
| n | 159 | 0 | 4954 | 5179 | 0 | 0 | 127 |
| % | 2.4 | 0 | 73.9 | 77.3 | 0 | 0 | 1.2 |

TABLE II: Number of NA Values for Partner Ratings

| | Attractive | Sincere | Intelligent | Fun | Ambition | Shared Interests/Hobbies |
|---|---|---|---|---|---|---|
| n | 166 | 226 | 244 | 275 | 583 | 864 |
| % | 2.5 | 3.4 | 3.6 | 4.1 | 8.7 | 12.9 |

TABLE III: Average attribute scores given by Females and Males

| | Ambition | Attractiveness | Fun | Intelligence | Shared Interests | Sincerity |
|---|---|---|---|---|---|---|
| Female | 6.62 | 6.46 | 6.54 | 7.31 | 5.56 | 7.27 |
| Male | 6.96 | 5.90 | 6.27 | 7.45 | 5.38 | 7.12 |

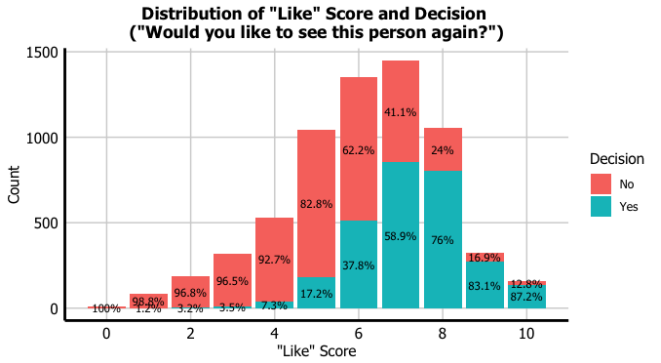


Fig. 5: Distribution of like score for Yes/No decisions

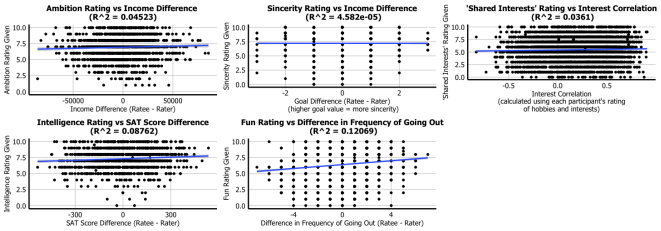


Fig. 6: Objective attribute differences vs. rating given

# References

[1] https://www.kaggle.com/annavictoria/speed-dating-experiment
[2] https://www.lesswrong.com/posts/uBNj85TAB2cikby2W/using-machine-learning-to-predict-romantic-compatibility
[3] https://www.kaggle.com/jph84562/the-ugly-truth-of-people-decisions-in-speed-dating

# Predicting the Success Rate of Speed Dating

(MS&E226 Project Part 2)

Cindy Kang, Esha Wang

## I. Introduction

In part one, we addressed concerns that our data was not well-organized and had a non-trivial number of NA values. To resolve this issue, we combined some manual data patching as well as a k-NN implementation to impute the missing data. For example, we noticed that there was one missing pid, and using the relationships between iid, id, and wave number, we deduced the missing value and manually added this into our dataframe. For the remainder of the data, we used k-NN imputation with $k = 10$. We note that because it's difficult to accurately impute values for covariates such as *income_diff*, *sat_diff*, or attribute ratings, using k-NN imputation may artificially reduce or enhance the signal in our data, thereby affecting the test error estimates that we obtain for our predictive models. On that subject, we also want to note that many rows in our dataset are dependent, since the same person is involved in multiple speed dates and ranks multiple partners based on their own preferences. Because we are treating each row as an independent observation in our analysis and model-building procedures, we are likely enhancing whatever signal (or lack thereof) is present in our data.

## II. Predicting the "Like" Score

We have designated our continuous response variable to be the 1-10 rating a participant gives to his/her partner, known as the "Like" score. Determined through our exploratory data analysis, the potential covariates to include in our model are the six attributes that partners were rated on (attractiveness, sincerity, intelligence, fun-ness, ambition, shared interests/hobbies) in addition to ten objective attributes (gender, age difference, difference in median income of each partner's hometown, difference in median SAT score of each partner's hometown, same race, interest correlation, difference in goals of attending the event, difference in going out frequency, importance of religion to rater, importance of race to rater). We note that "gender" and "same race" take on binary values and are treated as categorical variables in the models discussed below.

### A. Regression Models

As linear regression is one of the most prevalent approaches to prediction with the flexibility of adding increasing levels of complexity, we start off with a baseline model including all 16 of of the covariates mentioned above. We report the following model metrics: Adj. $R^2$, AIC, Train RMSE, CV RMSE of this baseline model and subsequent models in Table 1. The model metrics of this baseline model serve as comparison points as we increase the complexity of our model by adding interactions and higher order terms, adjusting our models through feature selection at each step along the way. Selected features and corresponding coefficients for each of the models mentioned in the table can be found in the appendix.

Recalling that our initial EDA pointed to the rating attributes as most correlated with "like" score, we build our most basic model, including only gender and the six rating attributes. Looking at the summary of our model, we find it interesting that the coefficient for *gender1* is negative, meaning that, on average, males rated their female partners more harshly than females rated males. More importantly, we observe that the coefficient for *ambition* is not statistically significant, and so we remove *ambition* from our model and report the metrics for the final "Ratings model" in Table 1. In particular, we note that in comparison to our baseline model, both our Train and CV RMSE have increased, indicating that there are other covariates (outside of gender and the six ratings) that hold significant incremental predictive power.

TABLE I
Model Metrics with Increased Model Complexity

| Model | Adj. $R^2$ | AIC | Train RMSE | CV RMSE |
|---|---|---|---|---|
| Stupid (Intercept Only) | 0.000 | 27228.630 | 1.844 | 1.845 |
| Baseline (All Covariates) | 0.651 | 20190.805 | 1.088 | 1.092 |
| Ratings | 0.649 | 20226.343 | 1.093 | 1.095 |
| Ratings + Interactions | 0.650 | 20203.427 | 1.091 | 1.093 |
| Ratings + Higher Order Terms | 0.650 | 20201.972 | 1.090 | 1.092 |
| Ratings + Interactions + Higher Order Terms | 0.651 | 20180.557 | 1.088 | 1.091 |
| Ratings + Interactions + Higher Order Terms + Objective Attr | 0.653 | 20155.439 | 1.086 | 1.089 |
| Ratings + Interactions + Higher Order Terms + Objective Attr + Higher Order Terms | 0.654 | 20138.755 | 1.084 | 1.088 |

We consider the possibility that gender affects the magnitude of impact that a covariate has on the "like" score. For this reason, we include interaction terms of gender with each of the 5 attribute ratings. From the resulting output, we note that only the coefficients of *gender*attr*, *gender*sinc*, *gender*intel* were statistically significant. In particular, the coefficient for *gender*attr* was positive and those for *gender*sinc*, *gender*intel* were negative, suggesting that men value attractiveness slightly more than females do when choosing a partner, and value sincerity and intelligence slightly less. However, despite these differences, based solely on the magnitude/direction of regression coefficients, the ranking of attribute importance remains the same for both males and females: attractiveness, shared interests/hobbies, fun-ness, sincerity, intelligence.

Adding in the three significant interaction terms to our previous "Ratings model", we obtain our "Ratings + Interactions model", with metrics reported in Table 1. Comparing the metrics to the original "Ratings model", we observe that the Adj. $R^2$ has increased, AIC has decreased, and both Train and CV RMSE have decreased. The first two points indicate that the increased model complexity (which increases the risk of overfitting) is justified by the incremental gain of adding interaction terms, and the last point suggests that the addition of interaction terms has increased the predictive power of our model.

Next, we add in higher-order terms to our model, hoping to better capture the relationship between our covariates and the "like" score. We looked at 4 different models, each including increasingly higher-order terms for our six ratings attributes (first model including $2^{nd}$ order terms, second model including $2^{nd}$ and $3^{rd}$ order terms, etc.), adjusted each model by removing features with non-statistically significant coefficients, and compared the adjusted model metrics with those of our original "Ratings" model. Based on this comparison, we chose the model including only $2^{nd}$ order terms for *attr*, *intel*, and

*fun*, and deemed this our "Ratings + Higher Order Terms" model. Again, we note that relative to the original "Ratings" model, the Adj. $R^2$ has increased, AIC has decreased, and both Train RMSE and CV RMSE have decreased. We then combine all of the covariates of our "Ratings + Interactions" and "Ratings + Higher-Order Terms" models in order to obtain our "Ratings + Interactions + Higher Order Terms" model. Based on the metrics of this combined model (shown in Table 1), we deem this our final ratings model.

Given our obtained final ratings model, we continue this process of adding on layers of complexity for our 10 objective attributes, adjusting the models through feature selection at each layer. We find that the only objective feature that adds significant incremental predictive power to our model is *imprace*, the importance of a partner being the same race as the rater. Additionally, we find that as the univariate distribution of *imprace* is heavily skewed right, taking the log transform of the variable decreases our observed Train and CV error. Hence, our "Ratings + Interactions + Higher Order Terms + Objective Attributes" model adds *log(imprace)* to our existing "Ratings + Interactions + Higher Order Terms" model, and sees both justified added complexity and increased predictive power. Adding interaction terms for our objective attribute *imprace* to this model were neither interpretable nor predictive, so we did not choose to include those. However, adding higher-order terms for *imprace* up to the $3^{rd}$ order increased our model's predictive power, as shown by the metrics of "Ratings + Interactions + Higher Order Terms + Objective Attributes + Higher Order Terms" displayed in Table 1.

Hence, our final regression model is given by:

$$like \sim gender + attr + sinc + intel + fun + shar + log(imprace + 0.01) + I(attr^2) + I(intel^2) + I(fun^2) + I(imprace^2) + I(imprace^3) + gender * attr + gender * sinc + gender * intel$$

Defining our stupid model to be a model that always predicts mean "like" score and our baseline model to be the model built on every covariate (with no interaction or higher-order terms), we note that the CV error of our stupid model is 1.845 and the CV error of our baseline model is 1.092. Given that the CV error of our final regression is 1.0877, we observe that, on average, our regression model performs 41% and 0.4% better than our stupid model and baseline model, respectively.

### B. Regularized Regression Models

Regularization provides a systematic method for our regression model to pick out meaningful coefficients, so as to eliminate the risk of overfitting by including too many covariates. Ridge and lasso regression do this by regularizing the objective function. While OLS regression implemented above minimizes the SSE, lasso regression minimizes $SSE + \lambda \sum_{j=1}^{p} |\hat{\beta}_j|$ (zero-ing out many of the less explanatory covariates) and ridge regression minimizes $SSE + \lambda \sum_{j=1}^{p} |\hat{\beta}_j|^2$ (penalizing $\hat{\beta}$ vectors with large norms). We run these two regularized regression models on our data, including all 16 covariate terms, and allow the procedures to select which covariates are most important. Model metrics displayed below.

TABLE II
PERFORMANCE OF REGULARIZED REGRESSION MODELS

| Model | $\lambda$ | $R^2$ | Train RMSE | CV RMSE |
|---|---|---|---|---|
| Chosen Regression Model | | 0.654 | 1.084 | 1.088 |
| Lasso | 0.0091 | 0.652 | 1.089 | 1.091 |
| Ridge | 0.1383 | 0.651 | 1.090 | 1.093 |

From the Train and CV RMSE's reported in table 2, we note that neither lasso nor ridge regression performed as well as our chosen regression model did on our training data; and the $R^2$ values of both lasso and ridge regression are lower than that of our chosen regression model. However, there are still important observations that we can note from these models.

As we know, lasso regression tends to zero out coefficients at a finite value of lambda. Looking at the Lasso coefficients, we note that the coefficients for *amb*, *age_diff*, *income_diff*, and *go_out_diff* were zeroed out, which is in agreement with our observations as we built our chosen regression model that these coefficients were not statistically significant. Additionally, lasso placed highest importance on *attr*, *shar*, and *fun*, which were also the three covariates with highest magnitude in our chosen regression model.

### C. Model Comparison and Selection

From the model metrics shown in Table II, we note that our chosen regression model outperforms both lasso and ridge models when making predictions on training data. Based on its higher predictive power and the fact that it is more interpretable (as covariates were hand-picked through careful analysis of model metrics to optimize predictive power while simultaneously preventing overfitting), we choose our final regression model as our "best model" for regression.

## III. PREDICTING THE "DECISION"

For our categorical response variable, we have chosen the yes/no response to the question "Would you like to see this person again?" following a date. The following predictive models were used to give an estimate of this response variable.

### A. Logistic Regression

Since logistic regression is one of the simplest classification models, we decided to start with this predictive model. Simply put, logistic regression uses the logistic function to model a binary dependent variable, which in our case is the yes/no decision score.

We ran a native logistic regression using all of the covariates, denoting "gender" and "samerace" as factors. After building the model, we obtained a 0-1 loss of 0.2411221. Coefficients are given in Figure 14 in the appendix. This will serve as our *baseline model*. Table 3 below shows the confusion matrix for this baseline model.

TABLE III
CONFUSION MATRIX FOR BASELINE LOGISTIC MODEL

| | | True Values | | |
|---|---|---|---|---|
| | | 0 | 1 | Total |
| Predicted Values | 0 | 3157 | 724 | 3881 |
| | 1 | 892 | 1929 | 2821 |
| | Total | 4049 | 2653 | 6702 |

We then performed a 5-fold cross validation on the training set using logistic regression, again using all columns. We obtained a prediction error estimate of 0.1648953, which looks quite promising.

Additionally, we wanted to see how the 0-1 loss would be affected if a single covariate was removed from the dataset entirely. This step measures how "essential" each covariate is in predicting the outcome variable. Figure 1 (left) shows the effect of removing each covariate on the total 0-1 loss of the training set. As expected, we observe that the attractiveness attribute has by far the most impact on the decision score with a 0-1 loss of 0.28 when it is removed from the data entirely. The fun, shared interest, and going out difference appear to have positive impact on the data as well.

Interestingly enough, we see that the 0-1 loss sometimes *improves* when a certain covariate is removed. There may be a few explanations for this. Quite a few entries in the original data were patched with k-nearest-neighbors. In particular, the NA values for age difference were difficult to measure, since most of the participants were in their mid-twenties, and responses did not vary widely with age. However, it seems more likely that the improved 0-1 losses are simply from randomness – the loss improved by a relatively trivial amount. Based on Figure 1 (left), we don't believe that removing any covariates would be significantly better for our predictive model.

Next, we considered the penalty of a false positive (matching two people when they are not compatible) or false negative (not matching

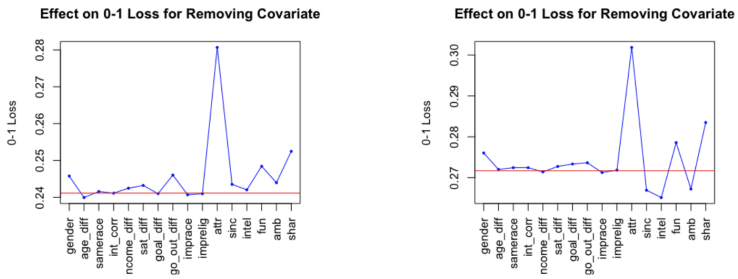


Fig. 1. 0-1 Loss upon removing a covariate using the logistic regression model (left) and the Naive Bayes model (right). The red line indicates the base 0-1 loss when all covariates are included in the model.

two people when are compatible). In speed dating, participants look for an ideal mate with the implicit assumption that they will meet many people who are incompatible. In other words, people expect to go on multiple "bad" dates before finding the right one. Therefore being matched with someone who is not compatible is less bad than missing a match that would be compatible. We would like to model this in our logistic regression.

One way to change the rate of false negatives is by modifying the threshold value. Currently, a decision is being set to 0 if the output of glm is negative and 1 otherwise. To increase the rate of 1's, we would simply need to set the threshold value below 0. For a threshold of $\lambda = -0.1$, we obtain the modified confusion matrix in Table 4. We see that the number of false negatives decreased by roughly 10%.

TABLE IV
CONFUSION MATRIX FOR LOGISTIC MODEL WITH $\lambda = -0.1$

| | | True Values | | |
|---|---|---|---|---|
| | | 0 | 1 | Total |
| Predicted Values | 0 | 3069 | 812 | 3881 |
| | 1 | 808 | 2013 | 2821 |
| | Total | 3877 | 2825 | 6702 |

### B. Naive Bayes

Naive Bayes is a conditional probability model that uses maximum likelihood and prior distribution to classify outcome variables. We thought this would be an appropriate model because of its prevalence and simplicity in classification analysis. Our data was then trained using the Naive Bayes model, and we obtained an in-sample 0-1 loss of 0.2717099.

We then analyzed the 0-1 loss for removing a single covariate from the dataset entirely. Figure 1 (right) is the result of our findings. Overall, the results are quite similar to that of Figure 1 (left). Attractiveness again has the highest impact to 0-1 loss, followed by shared interests and the fun attribute. Removing the intelligence and ambition attributes have a not so insignificant decrease in the 0-1 loss. This is consistent with our data exploration in part 1, where we saw that the intelligence and ambition attributes had insignificant correlation with our response variable.

### C. Classification Trees

Classification trees are one of most visually intuitive classification models, while still being effective in many real-life situations. It uses recursive partitioning of the space of independent variables, and is analogous to the well-known "20 Questions" game.

To begin, we must formalize the definition of a "complexity parameter", or CP. Classification trees are extremely prone to overfitting; adding complexity improves training error, but can be disastrous for the generalization error. The CP, denoted as $\alpha$, is a regularization parameter of the (regularized) cost function:

$$C_\alpha(T) = C(T) + \alpha|T|$$

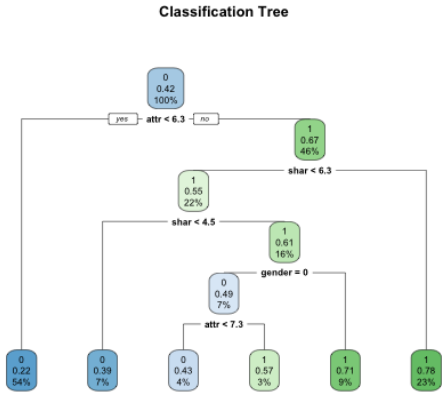


Fig. 2. Generously constrained classification tree initialized with all covariates.

where $C$ is the cost of a tree and $|T|$ is the number of terminal nodes. Therefore a small CP results in more complex trees and overfitting, while a large CP results in less complex trees and underfitting.

Figure 2 shows a baseline classification tree, with relatively generous constraints on complexity (CP $= 0.004$). (No contraints at all would cause the tree to overflow the page!) The variables actually used in the tree construction are attr, gender, and shar. The 0-1 loss using this method came out to 0.2491793.

In an attempt to "prune" the classification tree to limit its complexity as well as its overfitting potential, we can assign to it a larger CP. Figure 3 shows an example of a pruned classification tree with CP $= 0.01$. For practical purposes, the tree in Figure 5 is overly simplistic, and we would most likely use a CP value closer to 0.05 in Figure 2.

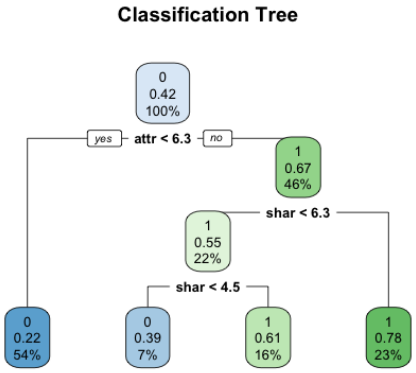


Fig. 3. Pruned classification tree initialized with all covariates.

### D. Model Comparison and Selection

In Figure 4, we plotted FNR (False Negative Rate) against all three classification models. Although Naive Bayes resulted in low FNR, it did not seem like an appropriate model as our covariates are clearly not independent of each other, violating its underlying assumption. Naive Bayes also had the greatest 0-1 loss out of all the categorical predictive models analyzed. With regards to classification trees, in addition to the very high FNR resulting from our tree model, we are also concerned with the potential of overfitting, as is prevalent with tree models. Though not included in our report, we also tried random forest modelling, which also seemed to overfit our data. Thus, we have decided on logistic regression with modified threshold as our "best model". From manual inspection on the training set as well as from Figure 4, we believe the threshold of -0.1 is sufficient. Further analysis and inference will be made in part 3. Lastly, the most basic model that predicts the most common output (a vector of all 0's) gives a 0-1 loss of 0.4209191, which is much worse than the accuracy of the models we built. Therefore we are confident that our "best model" will bring enlightening results.

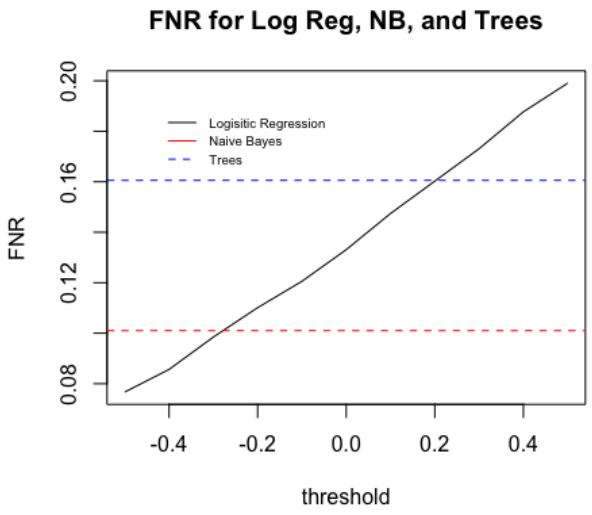


Fig. 4. FNR Curves for logistic regression, Naive Bayes, and classification trees.

## REFERENCES


[1] https://www.kaggle.com/colinleverger/exploring-the-speed-dating-dataset
[2] https://towardsdatascience.com/how-to-handle-missing-data-8646b18db0d4
[3] https://github.com/vincefav/speed-dating/blob/master/ML%20Capstone%20Report.pdf
[4] https://www.stat.berkeley.edu/ breiman/RandomForests/cc_home.htm#ooberr


## IV. APPENDIX

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.367e-01  7.595e-02   1.799 0.071996 .
gender1     -2.687e-02  2.832e-02  -0.949 0.342763
age_diff    -1.080e-03  2.816e-03  -0.383 0.701434
samerace1    2.685e-02  2.746e-02   0.978 0.328291
int_corr     7.451e-02  4.422e-02   1.685 0.092012 .
income_diff  1.289e-09  5.665e-07   0.002 0.998185
sat_diff    -3.145e-04  8.850e-05  -3.554 0.000382 ***
goal_diff    2.384e-02  1.802e-02   1.323 0.185910
go_out_diff -6.053e-03  8.715e-03  -0.695 0.487307
imprace     -2.441e-02  5.229e-03  -4.669 3.09e-06 ***
imprelig    -8.654e-03  5.365e-03  -1.613 0.106789
attr         2.963e-01  8.876e-03  33.379  < 2e-16 ***
sinc         1.283e-01  1.066e-02  12.032  < 2e-16 ***
intel        9.243e-02  1.310e-02   7.054 1.91e-12 ***
fun          2.278e-01  9.976e-03  22.837  < 2e-16 ***
amb         -9.264e-03  1.005e-02  -0.922 0.356544
shar         2.320e-01  8.249e-03  28.129  < 2e-16 ***
```

Fig. 5. Coefficients of Baseline Regression Model

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.008987   0.070444   0.128    0.898
gender1     0.015032   0.027222   0.552    0.581
attr        0.297363   0.008820  33.715  < 2e-16 ***
sinc        0.126162   0.010622  11.877  < 2e-16 ***
intel       0.086591   0.011965   7.237  5.1e-13 ***
fun         0.224724   0.009864  22.783  < 2e-16 ***
shar        0.234407   0.008154  28.746  < 2e-16 ***
```

Fig. 6. Coefficients of Ratings Model

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.166832   0.094097  -1.773 0.076276 .
gender1        0.405079   0.139159   2.911 0.003616 **
attr           0.269973   0.011458  23.563  < 2e-16 ***
sinc           0.142599   0.013921  10.244  < 2e-16 ***
intel          0.113395   0.016042   7.069 1.72e-12 ***
fun            0.227774   0.009863  23.094  < 2e-16 ***
shar           0.234623   0.008139  28.826  < 2e-16 ***
gender1:attr   0.058885   0.015476   3.805 0.000143 ***
gender1:sinc  -0.043283   0.021102  -2.051 0.040298 *
gender1:intel -0.060062   0.023476  -2.558 0.010537 *
```

Fig. 7. Coefficients of Ratings + Interactions Model

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.542128   0.175933  -3.081  0.00207 **
gender1      0.005821   0.027224   0.214  0.83071
attr         0.198497   0.035671   5.565 2.73e-08 ***
sinc         0.125904   0.010602  11.876  < 2e-16 ***
intel        0.225100   0.055985   4.021 5.87e-05 ***
fun          0.359342   0.036849   9.752  < 2e-16 ***
shar         0.233850   0.008139  28.733  < 2e-16 ***
I(attr^2)    0.008359   0.002943   2.840  0.00452 **
I(intel^2)  -0.009704   0.003947  -2.458  0.01399 *
I(fun^2)    -0.011405   0.002993  -3.810  0.00014 ***
```

Fig. 8. Coefficients of Ratings + Higher Order Terms Model

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.698311   0.180458  -3.870 0.000110 ***
gender1        0.361796   0.142092   2.546 0.010912 *
attr           0.176174   0.035899   4.908 9.44e-07 ***
sinc           0.140908   0.013906  10.133  < 2e-16 ***
intel          0.248922   0.056542   4.402 1.09e-05 ***
fun            0.358545   0.036838   9.733  < 2e-16 ***
shar           0.234065   0.008125  28.809  < 2e-16 ***
I(attr^2)      0.007902   0.002997   2.637 0.008385 **
I(intel^2)    -0.009518   0.003945  -2.413 0.015868 *
I(fun^2)      -0.011086   0.002993  -3.704 0.000214 ***
gender1:attr   0.059453   0.015710   3.784 0.000155 ***
gender1:sinc  -0.039627   0.021094  -1.879 0.060349 .
gender1:intel -0.059466   0.023480  -2.533 0.011343 *
```

Fig. 9. Coefficients of Ratings + Interactions + Higher Order Terms Model

```
Coefficients:
                          Estimate Std. Error t value Pr(>|t|)
(Intercept)             -0.5952755  0.1809686  -3.289 0.001009 **
gender1                  0.3993198  0.1417289   2.817 0.004854 **
attr                     0.1758982  0.0357915   4.915 9.11e-07 ***
sinc                     0.1452195  0.0138777  10.464  < 2e-16 ***
intel                    0.2291802  0.0564419   4.060 4.95e-05 ***
fun                      0.3579695  0.0367427   9.743  < 2e-16 ***
shar                     0.2335156  0.0081239  28.744  < 2e-16 ***
log(imprace + 0.01)     -0.1301849  0.0501151  -2.598 0.009405 **
I(attr^2)                0.0078951  0.0029871   2.643 0.008234 **
I(intel^2)              -0.0079797  0.0039402  -2.025 0.042887 *
I(fun^2)                -0.0109903  0.0029853  -3.681 0.000234 ***
I(imprace^2)             0.0150095  0.0056089   2.676 0.007469 **
I(imprace^3)            -0.0016283  0.0005005  -3.253 0.001146 **
gender1:attr             0.0567724  0.0156703   3.623 0.000293 ***
gender1:sinc            -0.0409981  0.0210314  -1.949 0.051292 .
gender1:intel           -0.0648686  0.0234155  -2.770 0.005616 **
```

Fig. 10. Coefficients of Ratings + Interactions + Higher Order Terms + Objective Attr Model

```
Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -0.5952755  0.1809686  -3.289 0.001009 **
gender1             0.3993198  0.1417289   2.817 0.004854 **
attr                0.1758982  0.0357915   4.915 9.11e-07 ***
sinc                0.1452195  0.0138777  10.464  < 2e-16 ***
intel               0.2291802  0.0564419   4.060 4.95e-05 ***
fun                 0.3579695  0.0367427   9.743  < 2e-16 ***
shar                0.2335156  0.0081239  28.744  < 2e-16 ***
log(imprace + 0.01) -0.1301849  0.0501151  -2.598 0.009405 **
I(attr^2)           0.0078951  0.0029871   2.643 0.008234 **
I(intel^2)         -0.0079797  0.0039402  -2.025 0.042887 *
I(fun^2)           -0.0109903  0.0029853  -3.681 0.000234 ***
I(imprace^2)        0.0150095  0.0056089   2.676 0.007469 **
I(imprace^3)       -0.0016283  0.0005005  -3.253 0.001146 **
gender1:attr        0.0567724  0.0156703   3.623 0.000293 ***
gender1:sinc       -0.0409981  0.0210314  -1.949 0.051292 .
gender1:intel      -0.0648686  0.0234155  -2.770 0.005616 **
```

Fig. 11. Coefficients of Ratings + Interactions + Higher Order Terms + Objective Attr + Higher Order Terms Model

```
                            1
(Intercept)  6.128624342
(Intercept)  .
age_diff     .
samerace1    0.006209340
int_corr     0.014729613
income_diff  .
sat_diff    -0.037756574
goal_diff    0.008724703
go_out_diff  .
imprace     -0.062662410
imprelig    -0.014988547
attr         0.572135210
sinc         0.219430280
intel        0.133202230
fun          0.440902829
amb          .
shar         0.493226803
```

Fig. 12. Coefficients of Lasso Model

```
                            1
(Intercept)  6.1195152747
(Intercept)  .
age_diff    -0.0046190763
samerace1    0.0289633354
int_corr     0.0218124878
income_diff  0.0006347398
sat_diff    -0.0458787678
goal_diff    0.0157017578
go_out_diff -0.0024185413
imprace     -0.0660664108
imprelig    -0.0247792553
attr         0.5441911603
sinc         0.2209848884
intel        0.1464710729
fun          0.4338458512
amb          0.0108902149
shar         0.4730398404
```

Fig. 13. Coefficients of Ridge Model

```
  (Intercept)       gender1      age_diff     samerace1      int_corr
-5.545063e+00  3.435384e-01 -5.861864e-03  3.711633e-02  7.705541e-02
  income_diff      sat_diff     goal_diff   go_out_diff       imprace
-1.561362e-06  7.784367e-04 -1.324496e-02  6.749381e-02 -4.837949e-02
     imprelig          attr          sinc         intel           fun
-5.173395e-03  5.360377e-01 -9.492335e-02  1.688266e-02  2.659263e-01
          amb          shar
-1.387839e-01  2.709061e-01
```

Fig. 14. Coefficients of Baseline Logistic Regression Model, run on all covariates

# Predicting the Success Rate of Speed Dating

(MS&E 226 Project Part 3)

Cindy Kang, Esha Wang

## I. Introduction

In this study, we hoped to explore what individuals look for in a partner during a speed date, and how this translates to the success rate of obtaining a second date. From Part I, we performed extensive exploratory data analysis to determine which response variables would be most realistic and fulfilling to predict. Ultimately, we decided to analyze how to predict the "like" score (continuous 1-10 rating) and "decision" (categorical yes/no answer to "Would you like to see this person again?") of a pair. In Part II, we generated several predictive models for both the regression and classification problems. In the end, we chose models on the basis of low estimated test error, using metrics such as $R^2$, AIC/BIC, 0-1 loss, confusion matrices, and CV error. In Part III, we will infer upon the previous sections using the set-aside test data, and discuss the overall performance of our chosen model.

## II. Prediction

### A. Regression – Linear Regression with Interactions and Higher-Order Terms

We first note that our test dataset was comprised of 1676 observations, of which 1418 were complete cases based on the eight raw variables included in our final regression model. Since predictions of "like score" can only be made in the case that we have no missing data values to input into our chosen regression model, we calculate two test errors: one based only on the 1418 test observations with no missing data values, and one based on 1635 observations with missing values imputed using k-NN with k=10. We clarify that there were 41 observations for which "like score" was not reported – so since it does not make sense to impute the value we are trying to predict, our imputed test set contains only 1635 observations instead of the original 1676.

The test errors we obtained were the following RMSE's: 1.0493 for the 1418 complete observations and 1.0488 for the imputed test set containing 1635 observations. This compares to 1.0877, which was our estimate for test error that we derived in part II using 10-fold cross validation. While the difference between our estimated and actual test errors is not large on an absolute scale, we found it quite surprising that our test errors for both cases were not only smaller than the estimated test error of 1.0877, but also smaller than our training RMSE of 1.0843.

We rule out the most obvious reasons for having lower test RMSE than train RMSE by going back to our train/test datasets and looking at the univariate distributions of "like" score for each. As shown in the histograms below, the distribution of "like" score for our train and test datasets do not differ significantly; and with the range of scores limited to values between 1 and 10, it is unlikely that our RMSE is pulled up by large errors on outlier values of "like" score.

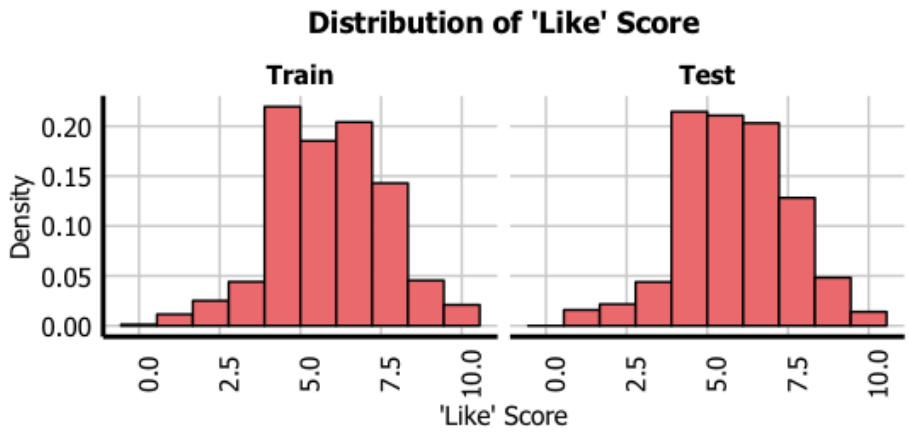


Fig. 1. Univariate distribution of "like" score for train and test sets

The most plausible explanation for this observation is actually that many rows of our original dataset were dependent, since the same person was involved in multiple speed dates and ranked multiple partners based on their own preferences. As noted in part II, we implemented model-building procedures and conducted analyses treating each row as as independent observation. It is very likely that this dependence between rows, coupled with the randomness involved in splitting our dataset into a train and test set, has indirectly introduced bias into our obtained test errors.

For example, if participant A was involved in 16 speed dates, and 15 of these observations were included in the train set, then the outcome of speed date 16 (in the test set) would be much easier to predict than in the case that only one or two of participant A's speed date records were included in the train set. In the hypothetical scenario that this situation (where most of a participant's speed date records were included in the train set and only one or two records were included in the test set) occurred with the large majority of the participants included in the study, the fact that our obtained test error is smaller than the estimated test error obtained using 10-fold CV would make much more sense.

This discussed dependence between the rows of our dataset is an inherent limitation of our original dataset that we pointed out even before starting the model-building process. Hence, while we wanted to make clear note of this technicality and its likely repercussions on our results, we still continue with the standard interpretation of our model and its prediction error values. That is, with a test RMSE of 1.049, we say that on average, our prediction of the "like" score that a speed dating participant will assign their partner, given information on how that participant rated their partner's attributes, will be off by approximately 1.05 points.

### B. Classification – Logistic Regression with Threshold

In Part II, we decided to use logistic regression with the threshold $\lambda = -0.1$ as our final classification model. We reported the confusion matrix in Table I, which corresponded to an in-sample training 0-1 loss of 0.2417189.

TABLE I
Confusion Matrix for Logistic Model with $\lambda = -0.1$, applied on training data

| | | True Values | | |
|---|---|---|---|---|
| | | 0 | 1 | Total |
| Predicted Values | 0 | 3069 | 808 | 3877 |
| | 1 | 812 | 2013 | 2825 |
| | Total | 3881 | 2821 | 6702 |

We estimated the test error using 5-fold cross validation, and reported a 0-1 loss of 0.1648903. Applying our model on the test set, we obtained a 0-1 loss of 0.2643198. The confusion matrix for predictions on the test set are given in Table II. The discrepancy between the estimated test error and actual test error is rather notable. We believe that the most significant factor that led to this discrepancy was due to imputation of missing values in the training and test sets. As mentioned previously, our data has quite a few missing or NA values. We hoped to salvage the data by using k-nearest-neighbors to complete the missing entries in both the training and test sets, treating them as two separate data sets during the imputation process. In hindsight, it may have been more practical to impute the training and test sets together as one complete dataset, in order to guarantee that the same imputation methodology is being used on both sets of data.

TABLE II
Confusion Matrix for Logistic Model with $\lambda = -0.1$, applied on test data

| | | True Values | | |
|---|---|---|---|---|
| | | 0 | 1 | Total |
| Predicted Values | 0 | 776 | 240 | 1016 |
| | 1 | 203 | 457 | 660 |
| | Total | 979 | 697 | 1676 |

The overall performance of our final classification model is summarized via ROC curves in Fig. 2. The AUC for the training set is 0.8343, while the AUC for the test set is 0.8108. This indicates strong evidence that our final classification model performs relatively well.

## III. Inference

### A. Statistical Significance of Regression Coefficients (Train)

At a 5% significance level, all coefficients in our chosen model were significant in the regression output with the exception of the *gender1:sinc* covariate, which had a p-value of 0.0513. In determining the statistical significance of a coefficient, a one-sample t-test is run to test the null hypothesis that the coefficient is equal to 0. The following t-statistic is calculated:

$$t = \frac{\hat{\beta}_j - 0}{SE\left(\hat{\beta}_j\right)}$$

and a corresponding p-value is determined using the t-distribution. If the p-value for a coefficient is less than the pre-determined significance level, that means our sample data provides enough evidence to reject the null hypothesis for the entire population.

By this definition, at a 5% significance level, we have sufficient evidence from our sample data to conclude (on a population-level) that changes in each of the covariates included in our regression model (with the exception of the *gender1:sinc* interaction term) are associated to nonzero changes in the "like" score. We believe these results since the raw variables included in our final model exhibited non-negligible correlation

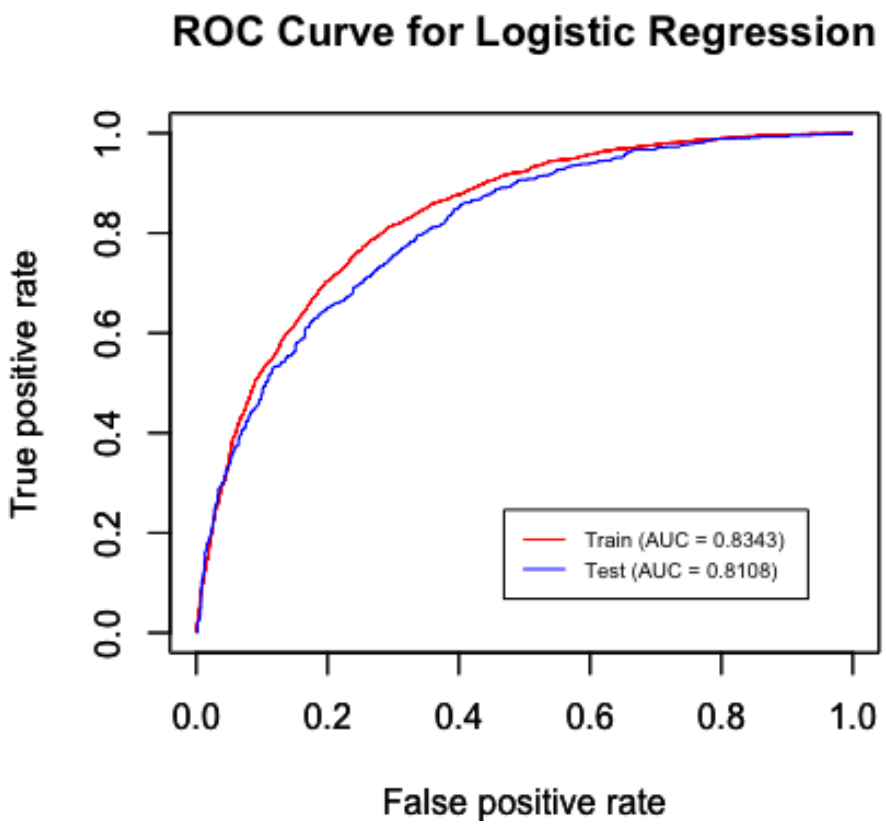


Fig. 2. ROC curves for train and test set using the logisitic regression model.

with "like" score during our initial EDA from part I (plot in Appendix Fig. 9). On top of that, the higher-order and interaction terms included in our chosen model demonstrated significant individual incremental predictive power during the model selection process through decreased AIC scores as well as CV errors.

### *B. Statistical Significance of Regression Coefficients (Test)*

Re-fitting our chosen model on our test data, however, we note that less than half of the covariates included in our model remained statistically significant at a 5% significance level. Only *gender1*, *attr*, *intel*, *fun*, *shar*, and *gender1:intel* had p-values less than 0.05. We note that there are three main plausible explanations for this observation:

– Increased standard errors of coefficients

The statistical significance of regression coefficients is determined by t-values and their corresponding p-values. By the t-distribution, extremely high or low t-values have lower probability – and the t-value is normalized using $SE\left(\hat{\beta}_j\right)$, which decreases as the sample size $n$ gets larger. Since our imputed test dataset contains $n = 1635$ observations, as compared to our training dataset with $n = 6702$ observations, it is analytically clear that $SE\left(\hat{\beta}_j\right)$ will likely be higher for coefficients in our test set than training set purely due to sample size differences. This makes it more difficult for resulting t-values to fall in the extremities of the t-distribution and hence more difficult to claim that a particular coefficient $\hat{\beta}_j$ is significant. Hence, small test set sample size and resulting higher standard errors of coefficients could be a factor in why so many of our covariates lost statistical significance when our model was fitted onto the test set.

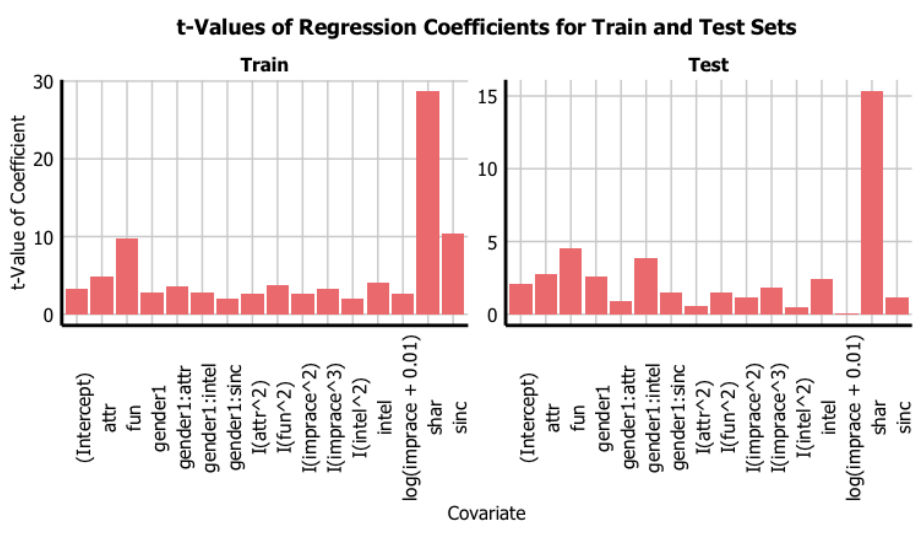


Fig. 3. Bar chart of t-values for regression coefficients of our model fitted on train and test sets. Higher t-values indicate higher attribute importance in predicting "like" score. Corresponding p-values included in appendix.

This reasoning is further supported by the fact that the distribution of t-values for the coefficients of covariates included in our chosen model appear to be quite similar between the train and test sets, as shown in Fig. 3 above. Note that higher t-values indicate higher attribute importance in predicting "like" score. Hence, similar distributions of t-values between train and test sets indicate that we observed similar signal in both datasets; but the fact that the range of t-values for the train set is much larger (y-axes scaled differently) points out that while the observed signals may have been similar, these signals were much stronger in the training set.

– Overfitting

The most obvious reason for coefficients losing statistical significance when the model is re-fitted onto the test set is the occurrence of overfitting. In the case that a model is too finely tuned to the training data, it is highly likely that when the model is refitted to a test set, many of the included covariates will lose statistical significance. This is because these additional covariates were fitted onto noise in the training data after most of the signal was already explained away by the primary covariates, which (presumably) should still be significant in the test set. Typically, the comparison of test error to train error would shed light on whether or not overfitting occurred with the model; but as discussed earlier, due to likely downward bias on our obtained test error from dependence of rows in our dataset, we are unable to confidently state whether or not we believe overfitting occurred with our model based purely on observed test errors.

However, based on our t-value distribution plot above, it appears likely that some overfitting did occur. In particular, while the coefficient of our transformed *imprace* covariate was significant when the model was fitted onto the training set, there was almost no signal coming from that covariate in the test set. Similar differences and signs of overfitting occurred with the covariates *gender1:attr*, *I(attr^2)*, *I(intel^2)*, and *sinc* (discussed in more detail below).

– Dependence of rows in data

While we are not retracting our statement made above that there are signs of overfitting from the t-value distribution plot in Fig. 3, we want to clarify that the observations noted above could also be explained by the dependence of rows in our data. As an example, we focus on the *sinc* covariate, and consider the case that the average person does not value sincerity in a speed dating partner, but there were a few participants in the study who valued sincerity as the topmost important attribute they were looking for. We note, in particular, that our full dataset is comprised of only 551 unique participants giving 8378 total observations, as each participant went on an average of 15.2 speed dates.

If by the randomness involved in splitting our dataset into a train and test set, most observations for speed dating participants who highly valued sincerity (*sinc*) in their partner were included in the training set and not the test set, it would make sense why sincerity was the second most important attribute in predicting "like" score in the train set, but exhibited relatively little signal in the test set. This compares to the case where, if each observation in our dataset were actually independent and the average person did not value sincerity, then the preferences of the few people who did value sincerity would not hold so much weight when we fit a model onto the training set, as much of their signal would be diversified away. Hence, we would have much more confidence in stating whether the observations noted from the t-value distribution plot in Fig. 3 are the result of overfitting if either our rows were not dependent with each other *or* we had a much larger sample size of unique participants in the study.

### *C. Bootstrap Analysis*

We performed a bootstrap analysis on the training set to estimate the sampling distribution for each of the sixteen covariates in our final regression model, including the intercept. To do this, we treated the training set as a sample from the population model, and drew $B = 10,000$ new samples of size nrow(train) $= 6702$ from the training data. We then conducted linear regression on the sample data to generate $B$ new models, and recorded the coefficients for each. A histogram of frequency versus value for each of the covariates is given in Fig. 4.

As the the size of each bootstrapped sample becomes very large, we expect to see the histograms to be normally distributed. We observe from Fig. 4 that this is approximately the case. Thus, for the purposes of this study, we will compare the 95% confidence interval using the normal interval method as well as the quantile interval method. The confidence intervals for the normal interval method were calculated using

$$\Big[\text{mean}(\beta_i^{(1)},\ldots,\beta_i^{(B)}) - 1.96\cdot\text{sd}(\beta_i^{(1)},\ldots,\beta_i^{(B)}),$$
$$\text{mean}(\beta_i^{(1)},\ldots,\beta_i^{(B)}) + 1.96\cdot\text{sd}(\beta_i^{(1)},\ldots,\beta_i^{(B)})\Big]$$

where $\beta_i$ is the value of the $i$th covariate, with $i$ ranging from 0 to 15. The confidence intervals for the quantile interval method were calculated using

$$\Big[\text{quantile}(\beta_i^{(1)},\ldots,\beta_i^{(B)}, 0.025),$$
$$\text{quantile}(\beta_i^{(1)},\ldots,\beta_i^{(B)}, 0.975)\Big]$$

We compare the normal interval and quantile interval method with the confidence intervals from R's standard regression output. Results all three types of confidence intervals are recorded in Table III. The confidence intervals calculated using the

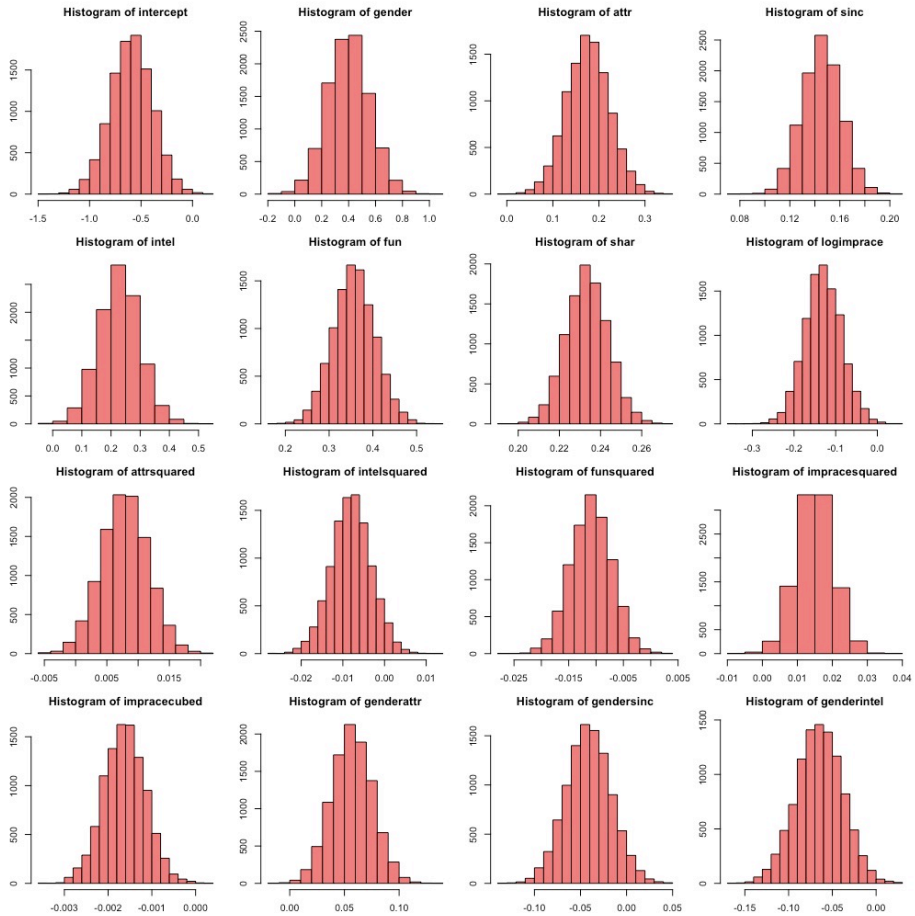


Fig. 4. Bootstrap distribution with 10,000 trials for each of the chosen covariates in the final regression model, including intercept. Each bootstrapped sample is the same size as the training set, sampled from the training set with replacement.

normal interval method and the quantile interval method are quite similar, as expected. This is because the distributions given in Fig. 4 are both approximately normal and symmetrical, consequently giving similar results for the confidence interval. Overall, the normal interval and quantile interval are both greater than R's interval. This is expected because bootstrap makes no assumptions on the population model.

TABLE III
95% CONFIDENCE INTERVALS USING NORMAL APPROXIMATION APPROACH AND QUANTILE APPROACH FOR BOOTSTRAP

| Covariate | Normal C.I. | Quantile C.I. | R's C.I. |
|---|---|---|---|
| intercept | [-0.994, -0.191] | [-1.000, -0.197] | [-0.950, -0.241] |
| gender | [0.0961, 0.7010] | [0.0959, 0.7030] | [0.122, 0.677] |
| attr | [0.0844, 0.2680] | [0.0846, 0.2680] | [0.106, 0.246] |
| sinc | [0.115, 0.176] | [0.115, 0.175] | [0.118, 0.172] |
| intel | [0.0912, 0.3660] | [0.0903, 0.3680] | [0.119, 0.340] |
| fun | [0.263, 0.451] | [0.263, 0.452] | [0.286, 0.430] |
| shar | [0.213, 0.254] | [0.213, 0.254] | [0.218, 0.249] |
| logimprace | [-0.217, -0.042] | [-0.2150, -0.0396] | [-0.228, -0.032] |
| attr^2 | [0.000515, 0.0152] | [0.000519, 0.0153] | [0.00204, 0.0137] |
| intel^2 | [-0.0174, 0.00147] | [-0.0176, 0.0014] | [-0.0157, -0.000257] |
| fun^2 | [-0.0182, -0.0036] | [-0.01820, -0.00372] | [-0.01680, -0.00514] |
| imprace^2 | [0.00467, 0.0253] | [0.00465, 0.0253] | [0.00402, 0.026] |
| imprace^3 | [-0.00257, -0.000684] | [-0.00257, -0.000678] | [-0.00261, -0.000647] |
| genderattr | [0.0205, 0.0935] | [0.0206, 0.0933] | [0.0261, 0.0875] |
| gendersinc | [-0.0891, 0.0069] | [-0.08890 , 0.00817] | [-0.082200 , 0.000223] |
| genderintel | [-0.1170, -0.0127] | [-0.1180, -0.0133] | [-0.111, -0.019] |

### D. Clustered Bootstrap Analysis

As mentioned previously, our data was constructed such that each row represents a participant-participant pair. A natural result of this is that rows in our data will be correlated, since each participant will be associated with multiple observations. In our regression model however, we assumed independence between observations. Clustered bootstrap is a way to measure how much this assumption affects our model.

We first define a cluster to be all observations given a specific participant, or "iid". Then 1,000 bootstrapped samples are generated, where each sample consists of several clusters of participants with each participant drawn uniformly from all participants with replacement. To be specific, suppose we have unique participants with $IID = \{iid_1, iid_2, \ldots, iid_n\}$. We draw samples from $IID$ with replacement, and for each $iid_i$ drawn, place $iid_i$'s cluster into our bootstrap sample. Repeat until the bootstrap sample's size reaches that of the original sample. Then using this procedure, we create 1,000 bootstrap samples. Histograms and confidence intervals for each coefficient, including intercept, are displayed in Fig. 5 and Table IV, respectively. Comparing Table III and IV, we see that the confidence interval is consistently equal or larger for clustered bootstrap compared to that of normal bootstrap. This suggests that the assumption that the rows in our data are independent is likely incorrect.

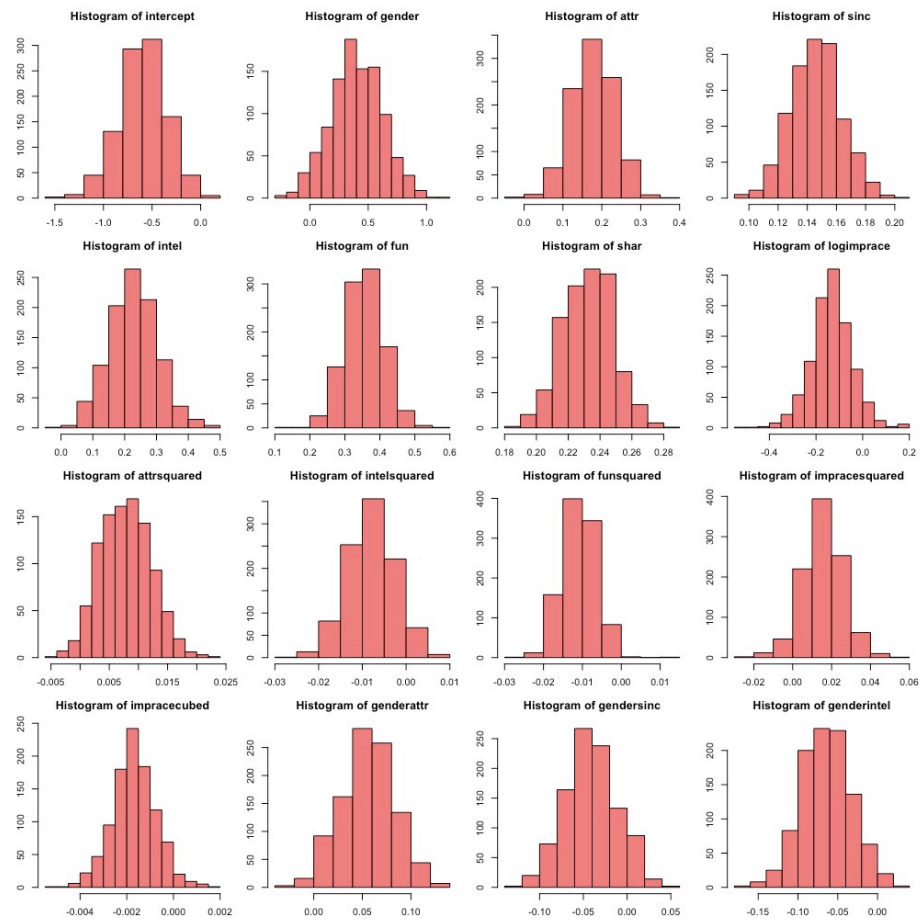


Fig. 5. Clustered bootstrap distribution with 1,000 trials for each of the chosen covariates in the final regression model, including intercept. A cluster is defined by participant IID. Each clustered bootstrapped sample is the same size as the training set, sampled from the training set with replacement.

TABLE IV
95% CONFIDENCE INTERVALS USING NORMAL APPROXIMATION APPROACH AND QUANTILE APPROACH FOR CLUSTERED BOOTSTRAP

| Covariate | Normal C.I. | Quantile C.I. | R's C.I. |
|---|---|---|---|
| intercept | [-1.070, -0.118] | [-1.090, -0.121] | [-0.950, -0.241] |
| gender | [-0.0358, 0.8360] | [-0.0427, 0.8420] | [0.122, 0.677] |
| attr | [0.071, 0.283] | [0.0688, 0.2770] | [0.106, 0.246] |
| sinc | [0.112, 0.180] | [0.112, 0.180] | [0.118, 0.172] |
| intel | [0.0802, 0.3760] | [0.0847, 0.3790] | [0.119, 0.340] |
| fun | [0.246, 0.466] | [0.248, 0.463] | [0.286, 0.430] |
| shar | [0.201, 0.264] | [0.202, 0.262] | [0.218, 0.249] |
| logimprace | [-0.3080, 0.0411] | [-0.3090, 0.0396] | [-0.228, -0.032] |
| attr^2 | [-0.000554, 0.016300] | [-6.82e-06, 1.63e-02] | [0.00204, 0.0137] |
| intel^2 | [-0.01860, 0.00274] | [-0.01860, 0.00256] | [-0.0157, -0.000257] |
| fun^2 | [-0.01910, -0.00252] | [-0.01910, -0.00277] | [-0.01680, -0.00514] |
| imprace^2 | [-0.00481, 0.03580] | [-0.00579, 0.03630] | [0.00402, 0.026] |
| imprace^3 | [-0.003510, 0.000165] | [-0.003570, 0.000295] | [-0.00261, -0.000647] |
| genderattr | [0.00257, 0.10900] | [0.00353, 0.10800] | [0.0261, 0.0875] |
| gendersinc | [-0.0985, 0.0166] | [-0.0973, 0.0168] | [-0.082200 , 0.000223] |
| genderintel | [-0.12600, -0.00163] | [-0.12300, -0.00253] | [-0.111, -0.019] |

### E. Comparison to Baseline Model

Based on our test dataset, the significant coefficients in our chosen model were: *attr*, *intel*, *fun*, *shar*, *gender*, and *gender:intel*; and the significant coefficients of our baseline model (which included all covariates) were *attr*, *intel*, *fun*, *shar*, *sinc*,and *imprelig*. We note that the first four out of six covariates in each list are the same, which makes sense because these are raw attributes included in both the baseline and chosen model, and additionally, these are also the top four important attributes of our test set based on the distribution of t-values in Fig. 3. Changes in the magnitudes of these coefficients going from baseline to chosen model, however, are likely due in higher part to the addition of interaction and higher-order terms in the chosen model (which subsequently changes the interpretation of these coefficients), rather than collinearity (discussed below) or the inclusion of variables competing for significance in the baseline model. Further supporting this statement, the fact that the coefficients of *gender* and *gender:intel* in our chosen model are both significant suggest that there are, in fact, significant differences in the magnitudes of attribute importance for males and females that could explain the changes in coefficients for the overlapping significant covariates between our baseline and chosen model. However, the fact that the *sinc* and *imprelig* covariates

were significant in the baseline model but not the chosen model suggests that there may be some partial collinearity occurring or perhaps variables competing for significance in the chosen model.

### F. Potential Problems with Analysis

*1) Collinearity:* Based on the nature of our data, we suspect that there is a high likelihood of (partial) multi-collinearity occurring in our model. This is due to the fact that humans have an innate propensity to generate first impressions of a person within seconds of meeting them. In fact, based on a 2006 Princeton University study, trait inferences of attractiveness, likeability, trustworthiness, competence, and aggressiveness are determined from facial appearance of other people within a tenth of a second of meeting them. Moreover, additional exposure time did not significantly change their judgments, but rather boosted confidence in initial judgments [3]. Hence, we suspect that ratings of attractiveness, intelligence, sincerity, fun-ness, and shared interests will all be correlated based on a participant's first impression of their partner, subsequently leading to the occurrence of (partial) multi-collinearity in our model. Below, we discuss the analytical evidence of the occurrence of (partial) multi-collinearity based upon four common warning signs.

– Our covariates have high pairwise correlations.

As we suspected based on psychological research [3] mentioned above, Fig. 6 below displays pairwise correlations between each of the 5 raw attributes included in our model (*attr*, *intel*, *sinc*, *fun*, *shar*) and supports that there exists significant correlation between our covariates (raw ratings of these attributes).

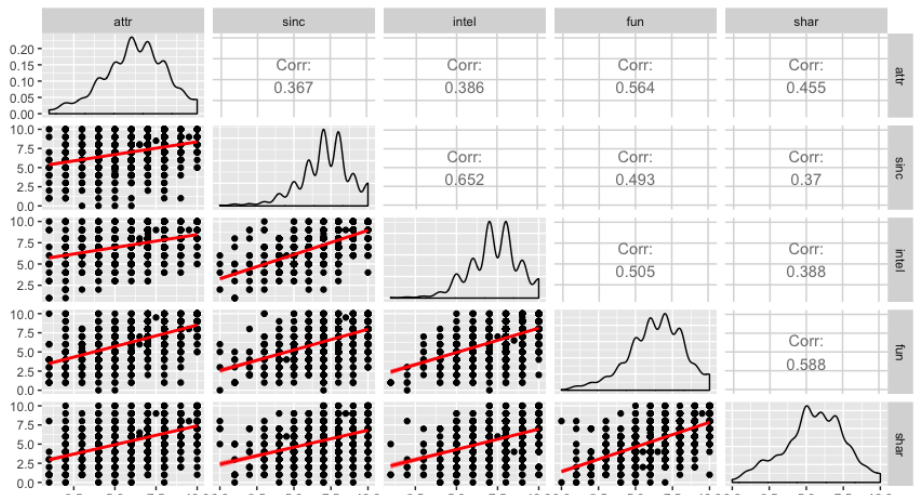


Fig. 6. Pairwise correlation plot of raw attributes included in model

– A negative (positive) regression coefficient is observed when the covariate is expected to have a positive (negative) correlation with the response variable.

According to the regression coefficients displayed in Fig. 8, the directions of all coefficients included in our model are sensible. In particular, we note that all non-interaction terms have a positive coefficient, indicating that higher ratings on individual attributes were associated with higher "like" scores. Additionally, we note that although the regression coefficient on the quadratic term of the attributes *intel* and *fun* are negative, due to possible ratings on these attributes being bounded between 1 and 10, the associative relationship between these individual attributes and "like" score are still all monotonically increasing. Hence, there does not appear to be a sign of collinearity for this consideration.

– A covariate is expected to have high correlation with the response variable, but its regression coefficient is not significant.

From our exploratory data analysis, we noted that attractiveness was by far the most highly correlated with "like" score, with an $R^2$ value of 0.592, followed by fun-ness rating, which had an $R^2$ value of 0.401. Based on this, we expected that attractiveness would rank as one of the topmost important attributes in our regression model as well. However, based on our distribution of t-values of regression coefficients in Fig. 3 and corresponding p-values displayed in Fig. 8, we note that the *attr* coefficient ranks only the sixth most significant, after the intercept term. Hence, it is likely that some form of (partial) collinearity is occurring.

– Regression coefficients change dramatically when we add or delete a covariate from the model.

Multi-collinearity refers to the general situation of an exact linear dependence among the covariates. By removing one covariate at a time, and comparing the coefficients of regressors included in the reduced model to those of regressors in the full (chosen) model, we don't see an *exact* linear dependence amongst our included covariates. However, we do note that, particularly for regressors with coefficients of large magnitudes, removing that particular covariate significantly changes coefficients of many other covariates in the model. This further supports that the issue of partial multi-collinearity exists in our model – that is to say, our design matrix $\mathbb{X}$ may have full rank, but is ill-conditioned such that the variances of our estimated coefficients will be large and our estimated coefficients will be very sensitive to the design matrix. This could also potentially explain why we observe that so many coefficients lose significance when we refit our chosen model onto the test set.

*2) Multiple Hypothesis Testing:* For our analysis, we aimed to test whether the entire set of covariates contributed significantly to the response variable. This acts as our multiple hypothesis scenario, where the null and alternative hypotheses $H_0$ and $H_A$ are given by:

$$H_0 : \beta_1 = \beta_2 = \ldots = \beta_{15} = 0$$

$$H_A : \text{Not all } \beta_i = 0$$

Given the nature of our data, we were wary of the actual significance of our covariates. For multiple hypothesis testing, we used several approaches including the Bonferonni Correction, the Benjamini-Hochberg (BH) procedure, and the F-statistic in R as outlined below.

– Bonferonni Correction

The Bonferonni correction approach ensures that the probability of declaring even one false positive is no more than 5%. It does so by rejecting any individual covariate as significant if its p-value is less than $0.05/n$, where $n$ is the number of individual hypothesis tests (in our case, $n = 15$). Therefore, we conclude a covariate is significant only if its reported p-value is less than $0.05/15 = 0.003333$. From Fig. 7, we see that the only covariates that satisfy this requirement are *attr*, *sinc*, *intel*, *fun*, *shar*, *I(funˆ2)*, *I(imparaceˆ3)*, and *gender:attr*. In summary, only half of the original covariates in the chosen model are deemed significant under multiple hypothesis testing with Bonferonni correction. However, it is known that the Bonferonni correction can often be too conservative, especially as the number of covariates becomes large.

– BH Procedure

A somewhat less conservative approach to multiple hypothesis testing is the Benjamini-Hochberg procedure. This procedure aims to ensure that the false discovery rate (FDR) is less than or equal to $\alpha$. We first order the p-values given in Fig. 7 in increasing order as $p_{(1)}, \ldots, p_{(n)}$ where $n = 15$, and reject null hypotheses $1, \ldots, j$ such that

$$p_{(j)} \leq \frac{\alpha j}{n}$$

The largest $j$ such that the above inequality holds is $j = 14$, where $p_{(j=14)} = 4.288678e^{-02}$ is the p-value of the $I(intel^2)$ attribute.

$$p_{(j=15)} = 0.042887 \leq \frac{\alpha j}{n} = \frac{0.05 \cdot 14}{15} = 0.04666667$$

By the BH algorithm, we then reject all null hypotheses for $j = 1, \ldots, j = 14$. Thus, the BH approach fails to reject the null for only one covariate (*gender:sinc*), unlike the Bonferroni method which failed to reject the null for eight covariates. This observation is not surprising, since the BH correction is constructed to be less conservative than the Bonferroni correction. For our particular model, the Bonferroni correction method seems more feasible than the BH method due to our post-selection inference analysis indicating that about half of our covariates should be insignificant. (See the following section on post-selection inference.)

– F-statistic

A final method used for multiplied hypothesis testing is the F-statistic provided by R. An F-test is performed in which the F-statistic has an F distribution under the null hypothesis. According to the summary of our regression model, the F-statistic of our model is 843.8, with a corresponding p-value of $2.2 \times 10^{-16}$. This p-value is sufficiently small to confidently reject the null hypothesis.

*3) Post-selection Inference:* By doing model-building and inference on the same data that we used to select our model, we inherently overestimate the association between the covariates and the response variable, thereby introducing some amount of bias in determining which coefficients are significant. There are a number of aspects in our model-building and inference process that may have contributed to this bias. First, during the model selection step of Part 2 of the project, we looked at statistical significance of coefficients in addition to adjusted $R^2$, AIC/BIC, and CV error in selecting our final model, in efforts to optimize for both low prediction error and model interpretability. (Based on feedback from Part 2, we now realize we were only supposed to optimize for predictive power during model selection in Part 2.) Clearly, if statistical significance was one of the factors we considered in selecting covariates to include in our final model, making inferences based off of the significance of the coefficients in the model fitted to our training data would be very biased and the inferences would most definitely be overstated.

In addition, during the model building step, we started with inclusion of only raw attribute ratings as covariates, and built upon this simple model by adding on interaction terms, higher-order terms, and finally objective attributes, including a term only if it added significant incremental predictive power to our existing model. We chose this order of step-wise model building due to the fact that, based on our exploratory data analysis from part 1 of the project, our raw attribute ratings seemed most positively correlated with "like" score. By following this order in performing step-wise regression, however, we, again, favorably biased which coefficients would be significant in our final model. Formally, for a coefficient $\hat{\beta}_j$, we are performing the hypothesis test assuming that under the null, $\hat{\beta}_j$ is $N(0, SE_j^2)$ when in fact, under the null together with our selection process, $\hat{\beta}_j$ is actually more likely to be positive.

One way to account for this problem is to validate our finding on new data, in this case the test set. To do this, we fit our final regression model on the test set, and calculated corresponding p-values and confidence intervals. Fig. 7 and Fig. 8 display

the differences between our model applied on old data (the training set) and our model applied on new data (the test set). Specifically, in part a of the inference section, we found that all covariates except for *gender:sinc* were significant at a 0.05 significance level when our model was applied to the training set. From this, we concluded that a change in any one of the "significant" coefficients (all but *gender:sinc*) resulted in a non-negligible change in the response variable. However, in part b of the inference section, after refitting the chosen model to the test set, we found that only seven of the original sixteen covariates were significant when our model was applied on new data. This suggests that our initial stated inference regarding the association between the covariates and response variable was most likely very overstated, due to reasons discussed above.

### G. Interpretations of Causality

From previous analyses, we demonstrated strong association between a partner's attractiveness and his/her "like" score. But the issue with declaring causality, or in other words, stating that a participant will be rated highly if he/she is attractive, is unfortunately problematic. For each case where an individual perceived as universally attractive got a high "like" score, there exists an analogous case where the same individual is universally perceived as not attractive. Because we do not have data for the latter cases, and whether or not these cases translate to a low "like" score, we cannot draw the causal inference that higher attractiveness results in higher "like" score. In many situations, the lack of data for these alternative cases can be remedied by creating a control group. However, creating a control group of "ugly" participants is not only tricky (attractiveness is very subjective!), but also may not be a very ethical experiment to conduct. Therefore we conclude that there is strong *association* between attractiveness and "like" score, but there is not necessarily a "causal relationship" between the two. Without loss of generality, this same argument can be applied to our multiple regression case where we declare an association between "like" score and received individual attribute ratings in addition to interaction terms and higher-order terms. That is to say, though we have sufficient statistical evidence to declare an association between our covariates and "like" score, we do not have sufficient evidence to declare causality.

## IV. Discussion

### A. Practical Uses and Refitting

In practical settings, we expect that our model would be used primarily for inference and not prediction. This is because our model makes use of a speed dating participant's ratings of their matched partners' personal attributes in order to predict the "like" score that the participant will assign that partner. However, since there is generally no objective standard of attractiveness, sincerity, intelligence, fun-ness, ambition, or shared interests/hobbies, and different peoples' ratings of one individuals' personal attributes can vary widely, there are virtually no practical settings for which the predictions that we *can* make from our model would be useful.

Though we regret that our model is not as practical as we would like on the prediction front, there are quite a few interesting inferential statements that can be made about what people primarily look for in a partner upon first meeting. For example, we know that the top attributes that most people look for are shared interests and fun-ness. These types of inferences can be used in decision-making when organizing a speed-dating event for which participants fill out an initial survey before the day of the actual event to help organizers determine which circle of people they would be most likely to click with. Similarly, these inferences could be used in building an online dating app with in-app questionnaires to gauge shared interests of two parties. We make note, however, that extrapolation of our model inferences too far beyond the settings of speed dating and online dating could potentially lead to very bad results. The primary reason for this is that the response variable of our model is a "like" score determined four minutes after a brief and shallow interaction with a potential partner. In particular, we note that these inferences about what brings two people together on a second date likely differ significantly from what long-term relationships thrive off of, and what someone looks for in a speed-dating or online-dating partner may not always match what they look for in a marriage partner. Determining inferences in those scenarios would require further study and analysis.

On the topic of these alternative scenarios, we note that our model may not stay relevant for a long period of time due to the rapid evolution of speed dating and online dating. In particular, we note that traditional speed dating is almost entirely outdated now with most "speed-dating" events now actually serving as an ice-breaker activity to find friends in new settings, and the uses of online dating have rapidly transformed within the last decade. Specifically, while online dating was formerly used primarily by busy middle-aged people looking for a serious relationship, the proliferation of dating apps such as Tinder and OkCupid have gained popularity amongst teens and college students as a means of obtaining hookups and casual relationships. We would hope that what people look for in a hookup or casual relationship differs, at least in part, from what people look for in a potential life partner. Hence, we would expect that our model would need to be refitted every 3-6 years, depending on how rapidly changes in the use of speed dating and online dating take place.

### B. Caveats and Limitations

People who would like to use our regression model should be aware of several nuances. Firstly, the original dataset contained vastly more columns than the ones we used in our model. We removed these columns either due to too many missing entries or if they displayed obvious collinearity. Furthermore, they should be aware that, by the nature of our data, many of the covariates in our model will be at least partially collinear. This is because many attributes, such as shared attributes, fun, and sincerity, are linked, and thus oftentimes all large or all small in score. As discussed above, this partial collinearity can lead to inflated variances of our estimated coefficients, and additionally, high sensitivity of our estimated coefficients to the design matrix.

Additionally, our model has a high tendency to overfit, especially since the training set and test set were imputed separately. We suggest imputing the entire dataset first, before splitting it into a training and test set to guarantee that the imputation process is the same for both sets of data. In addition, we did not set aside a holdout validation set for use during our model selection process. Because both the training and CV RMSEs we obtained and compared in selecting our best model were very close in magnitude, our final chosen model has a high likelihood of having overfit the training data. To add to this, we do not have access to standard metrics of assessing the degree of overfitting (namely, comparing train vs. test error), as non-independence of rows in our dataset have likely put a downward bias on our obtained test errors in addition to a whole slew of other issues.

On the topic of non-independence of rows, we also note that the coefficient confidence intervals generated using clustered bootstrap are more accurate than those generated using general bootstrap methods, due to the larger standard errors of clustered bootstrap somewhat accounting for the dependence of rows in our data. Lastly, for multiple hypothesis testing, it is important to use the most practical test for our model. In particular, we noted that the Bonferonni correction is the most appropriate approach, even though it is generally considered conservative, since its results better matches our analysis in post-selection inference (with only seven covariates deemed significant on the new data).

### C. What We Would Do Differently

The data that we were provided has a significant portion of missing values. This is mostly due to the data collection process being inherently flawed, as it allows users to leave answers blank or fill in answers incorrectly. The data collection process could be improved by having users input their answers into a computer-generated survey, where each blank *requires* a response and automatically checks if the answer provided is valid. For example, some participants responded "UC, IRVINE!!!!!!!!!", or similar expressions of enthusiasm, when asked to input their undergraduate institution. Allowing these types of answers muddy the data, and require a significant amount of manual data cleaning. Instead, having users use a drop-down menu to select which undergraduate institution they attended would be much more organized and practical.

There are numerous columns in our original data that were not as individualized as we would like. For example, "SAT Score" was the median SAT score for the undergraduate institution that the participant attended, not his/her individual SAT score. Similarly, "income" was calculated as the median household income based on zip code using the Census Bureau website. Since SAT score and income can vary widely within a school or zip code, we believe that the data provided for these columns is not very meaningful. Rather, we would like to collect more personalized data, such as individual SAT score, average undergraduate GPA, or individual income bracket.

Additionally, our data was constructed so that each row of the training data represented one interaction pair between two participants. As a result, many of the rows in our data are correlated, easily leading to misinterpretations in the regression since it assumes independent rows. Although we attempted to account for this violation of assumption by performing clustered bootstrap in the Inference section, there were situations in which we were not able to account for the consequences of this non-independence between rows. For instance, there wasn't a way for us to adjust our obtained test error in order to account for the likely downward bias that occurred. If we could change the entire data collection process, we would collect data from more unique participants rather than collect many observation points from each participant, so that many of these issues could be avoided.

If we were to work with the exact same dataset again, there are four main things we would to differently. Firstly, we realized that our k-nearest-neighbors algorithm was applied to the training set and test set independently. This led to a larger-than-expected discrepancy between the regression models applied on the training set versus on the test set. We should have instead applied k-NN on the entire data, and then split the data accordingly. Secondly, we would have liked to further analyze the best k-NN approach for imputation of missing data. In this study, we chose $k = 10$, since this is often the standard. However, we believe our results would benefit from experimenting with different values for $k$. Thirdly, we would have set aside both a validation dataset and a test dataset, as opposed to just a test dataset. Having a holdout validation dataset would have helped tremendously with the model selection process of part 2 of this project. It not only gives us one additional metric to use during model selection, but also drastically decreases the likelihood of selecting a model that severely overfits the training data, especially in our case where RMSEs were quite similar across the board. And lastly, on the topic of model building and selection, we would not have looked at coefficient standard errors when selecting a best model if we had known we were only optimizing for low prediction error in part 2 or if we understood the gravity of post-selection inference issues we brought about by doing so.

```
Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         -0.5952755  0.1809686  -3.289 0.001009 **
gender               0.3993198  0.1417289   2.817 0.004854 **
attr                 0.1758982  0.0357915   4.915 9.11e-07 ***
sinc                 0.1452195  0.0138777  10.464  < 2e-16 ***
intel                0.2291802  0.0564419   4.060 4.95e-05 ***
fun                  0.3579695  0.0367427   9.743  < 2e-16 ***
shar                 0.2335156  0.0081239  28.744  < 2e-16 ***
log(imprace + 0.01) -0.1301849  0.0501151  -2.598 0.009405 **
I(attr^2)            0.0078951  0.0029871   2.643 0.008234 **
I(intel^2)          -0.0079797  0.0039402  -2.025 0.042887 *
I(fun^2)            -0.0109903  0.0029853  -3.681 0.000234 ***
I(imprace^2)         0.0150095  0.0056089   2.676 0.007469 **
I(imprace^3)        -0.0016283  0.0005005  -3.253 0.001146 **
gender:attr          0.0567724  0.0156703   3.623 0.000293 ***
gender:sinc         -0.0409981  0.0210314  -1.949 0.051292 .
gender:intel        -0.0648686  0.0234155  -2.770 0.005616 **
```

Fig. 7. Coefficients for chosen regression model applied to training set.

```
Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         -0.817774   0.394347  -2.074  0.03826 *
gender               0.785545   0.282441   2.781  0.00548 **
attr                 0.175524   0.074287   2.363  0.01826 *
sinc                 0.025449   0.027840   0.914  0.36079
intel                0.336662   0.121956   2.761  0.00584 **
fun                  0.366817   0.073921   4.962 7.70e-07 ***
shar                 0.242700   0.016030  15.140  < 2e-16 ***
log(imprace + 0.01) -0.012077   0.104519  -0.116  0.90803
I(attr^2)            0.006709   0.006164   1.088  0.27656
I(intel^2)          -0.006918   0.008392  -0.824  0.40984
I(fun^2)            -0.008876   0.005966  -1.488  0.13701
I(imprace^2)         0.011343   0.011479   0.988  0.32322
I(imprace^3)        -0.001575   0.001010  -1.559  0.11919
gender:attr          0.034457   0.031020   1.111  0.26682
gender:sinc          0.069535   0.040980   1.697  0.08993 .
gender:intel        -0.202874   0.047650  -4.258 2.19e-05 ***
```

Fig. 8. Coefficients for chosen regression model applied to test set.

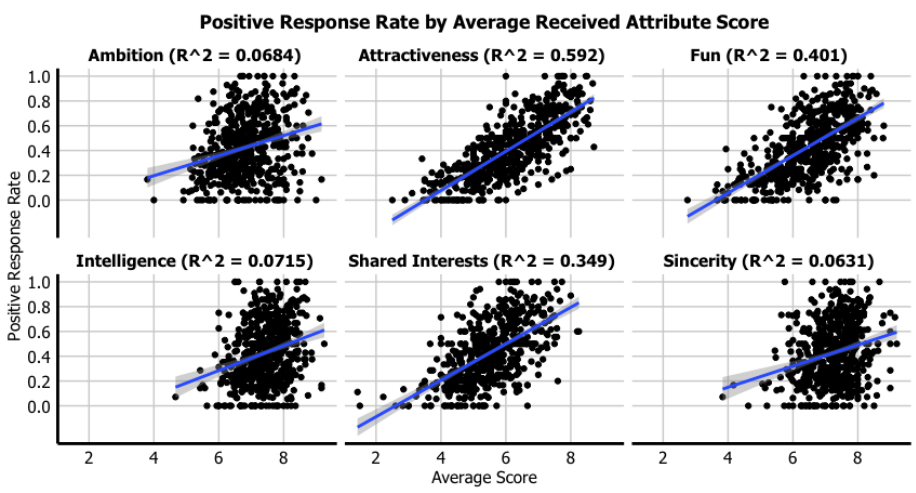


Fig. 9. Positive Response Rate vs. Received Attribute Score

## REFERENCES

[1] Fisman, Ray and Sheena Iyengar. (2004). Gender Differences in Mate Selection: Evidence From a Speed Dating Experiment. Retrieved from https://www.kaggle.com/annavictoria/speed-dating-experiment

[2] Fisman, R., Iyengar, S. S., Kamenica, E., & Simonson, I. (2008). Racial preferences in dating. The Review of Economic Studies, 75(1), 117-132.

[3] Willis, J., & Todorov, A. (2006). First Impressions: Making Up Your Mind After a 100-Ms Exposure to a Face. Psychological Science, 17(7), 592598. https://doi.org/10.1111/j.1467-9280.2006.01750.x