Speed Dating

BAIS:6070 Data Science Final Project

Collin Harrell, Zoe Heimendinger, Hailey Hoffman, Andrew McQueen

Introduction

This project analyzes data collected by Ray Fisman and Sheena Iyengar from Columbia Business School during experimental speed dating events from 2002 to 2004. There were 21 waves in this two-year period which all had different participants and varying numbers of males and females within each round. We want to determine which features are most important in predicting compatibility between males and females in a relationship setting. The original dataset has 121 features with a binary target of a match (1) or not a match (0) achieved after the four minute "first date". All the participants were required to rank their date based on six features: attractiveness, sincerity, intelligence, fun, ambition, and shared interests. Because these are all subjective measurements, we are curious to see how strong of a prediction we can achieve using machine learning. The participants also completed a questionnaire regarding personal information such as demographics, interests, lifestyle, and subjective opinions on what they value most and least when seeking a partner.

Data Dictionary:

Variable	Description	Type
Match (target)	0 if didn't match, 1 if did match	nominal
Has_null	Whether or not the observation has any null values	nominal
Wave	Round of speed dating	numeric
Gender	Male or female	nominal
Age	How old the person is	numeric
Age_o	Age of partner	numeric
D_age	age	numeric
D_d_age	age binned	nominal

Race	Race of person	nominal
Race_o	Race of partner	nominal
samerace	Whether or not the person is the same race as their partner	nominal
Importance_same_race	Scale of 0-10 how important is it that the person and their partner are of the same race	numeric
Importance_same_religion	Scale of 0-10 how important is it that the person and their partner are of the same race	numeric
D_importance_same_race	Importance of same race binned	nominal
D_importance_same_religion	Importance of same religion binned	nominal
Field	Field of study	nominal
Pref_o_attractive	How important does the partner rate attractiveness	numeric
Pref_o_sincere	How important does the partner rate sincerity	numeric
Pref_o_intelligence	How important does the partner rate intelligence	numeric
Pref_o_funny	How important does the partner rate humor	numeric
Pref_o_ambitious	How important does the partner rate ambition	numeric
Pref_o_shared_interests	How important does the partner rate having shared interests	numeric
D_pref_o_attractive	Importance of attractiveness binned	nominal
D_pref_o_sincere	Importance of sincerity binned	nominal
D_pref_intelligence	Importance of intelligence binned	nominal
D_pref_o_funny	Importance of humor binned	nominal
D_pref_o_ambitious	Importance of ambition binned	nominal
D_pref_o_shared_interests binned	Importance of shared interests binned	nominal
Attractive_o	Rating by partner of attractiveness 0-10	numeric
Sincere_o	Rating by partner of sincerity 0-10	numeric
Intelligence_o	Rating by partner of intelligence 0-10	numeric
Funny_o	Rating by partner of humor 0-10	numeric
Ambitious_o	Rating by partner of ambition 0-10	numeric
Shared_interests_o	Rating by partner of shared interests 0-10	numeric
D_attractive_o	Attractiveness score given by partner binned	nominal
D_sincere_o	Sincerity score given by partner binned	nominal
D_intelligence_o	Intelligence score given by partner binned	nominal
D_funny_o	Humor score given by partner binned	nominal

D_ambitious_o	Ambition score given by partner binned	nominal
D_shared_interests_o	Shared interest score given by partner binned	nominal
Attractive_important	How important the person rates attractiveness	numeric
Sincere_important	How important the person rates sincerity	numeric
Funny_important	How important the person rates humor	numeric
Ambition_important	How important the person rates ambition	numeric
Shared_interest_important	How important the person rates shared interests	numeric
D_attractive_important	Attractiveness importance score binned	nominal
D_sincere_important	Sincerity importance score binned	nominal
D_intelligence_important	Intelligence importance score binned	nominal
D_funny_important	Humor importance score binned	nominal
D_ambition_important	Ambition importance score binned	nominal
D_shared_interests_important	Shared interest importance score binned	nominal
Attractive	Self-rating of attractiveness 0-10	numeric
Sincere	Self-rating of sincerity 0-10	numeric
Intelligence	Self-rating of intelligence 0-10	numeric
Funny	Self-rating of humor 0-10	numeric
Ambition	Self-rating of ambition 0-10	numeric
D_attractive	Self-rated attractiveness score binned	nominal
D_sincere	Self-rated sincerity score binned	nominal
D_intelligence	Self-rated intelligence score binned	nominal
D_funny	Self-rated humor score binned	nominal
D_ambition	Self-rated ambition score binned	nominal
Attractive_partner	Attractiveness rating of partner 0-10	numeric
Sincere_partner	Sincerity rating of partner 0-10	numeric
Intelligence_partner	Intelligence rating of partner 0-10	numeric
Funny_partner	Humor rating of partner 0-10	numeric
Ambition_partner	Ambition rating of partner 0-10	numeric
D_attractive_partner	Attractiveness rating of partner binned	nominal
D_sincere_partner	Sincerity rating of partner binned	nominal
D_intelligence_partner	Intelligence rating of partner binned	nominal
D_funny_partner	Humor rating of partner binned	nominal
D_ambition_partner	Ambition rating partner binned	nominal
D_shared_interests_partner	Shared interest rating of partner binned	nominal
Sports	How interested is the person in sports 0-10	numeric

Tvsports	How interested is the person in watching sports 0-10	numeric
Exercise	How interested is the person in exercise 0-10	numeric
Dining	How interested is the person in dining 0-10	numeric
Museums	How interested is the person in museum 0-10	numeric
Art	How interested is the person in art 0-10	numeric
Hiking	How interested is the person in hiking 0-10	numeric
Gaming	How interested is the person in gaming 0-10	numeric
Clubbing	How interested is the person in clubbing 0-10	numeric
Reading	How interested is the person in reading 0-10	numeric
TV	How interested is the person in TV 0-10	numeric
Theater	How interested is the person in theater 0-10	numeric
Movies	How interested is the person in movies 0-10	numeric
Shopping	How interested is the person in shopping 0-10	numeric
Yoga	How interested is the person in yoga 0-10	numeric
D_sports	Sports score binned	nominal
D_tvsports	Tv sports score binned	nominal
D_exercise	Exercise score binned	nominal
D_dining	Dining score binned	nominal
D_museums	Museum score binned	nominal
D_art	Art score binned	nominal
D_hiking	Hiking score binned	nominal
D_gaming	Gaming score binned	nominal
D_clubbing	Clubbing score binned	nominal
D_reading	Reading score binned	nominal
D_tv	Tv score binned	nominal
D_theater	Theater score binned	nominal
D_movies	Movie score binned	nominal
D_concerts	Concert score binned	nominal
D_music	Music score binned	nominal
D_shopping	Shopping score binned	nominal
D_yoga	yoga scores binned	nominal
Interests_correlate	Correlation between person and partner’s ratings of interests	numeric
D_interests_correlate	Correlation between person and partner’s ratings of interests binned	nominal

Expected_happy_with_sd_people	How happy does the person expect to be with the people met at the event 0-10	numeric
Expected_num_interested _In_me	Expected number of people interested in person	numeric
Expected_num_matches	Expected number of matches	numeric
D_expected_happy_with_sd_people	Happiness expectation score binned	nominal
D_num_interested_in_me	Number of people expected to be interested binned	nominal
D_expected_num_matches	Expected number of matches binned	nominal
Like	How much the person liked their partner 0-10	numeric
Guess_prob_liked	Guess of what partner rated liking them 0-10	numeric
D_like	Like score binned	nominal
D_guess_prob_liked	Guess of what partner rated them binned	nominal
Met	Met partner how many times before	numeric
Decision	Whether the person wants to match with their partner	nominal
Decision_o	Whether the partner wants to match with the person	nominal

Features Eliminated:

Attribute Eliminated	**Reason**
Has_null	Not useful for predicting, just indicates whether or no the row has any null values
Age	Information contained in combination variable, d_d_age
Age_o	Information contained in combination variable, d_d_age
D_age	Information contained in the binned version of this variable, d_d_age
Importance_same_race	Information contained in combination variable d_importance_same_race
Importance_same_religion	Information contained in combination variable d_importance_same_religion
Pref_o_attractive	Information contained in combination variable d_pref_o_attractive
Pref_o_sincere	Information contained in combination variable d_pref_o_sincere
Pref_o_intelligence	Information contained in combination variable d_pref_o_intelligence
Pref_o_funny	Information contained in combination variable d_pref_o_funny

Pref_o_ambitious	Information contained in combination variable d_pref_o_ambitious
Pref_o_shared_interest	Information contained in combination variable d_pref_o_shared_interest
Attractive_important	Information contained in combination variable d_attractive_important
Sincere_important	Information contained in combination variable d_sincere_important
Intelligence_important	Information contained in combination variable d_intelligence_important
Funny_important	Information contained in combination variable d_funny_important
Ambition_important	Information contained in combination variable d_ambition_important
Shared_interests_important	Information contained in combination variable d_shared_interests_important
Sports	Information factored into d_interests_correlate variable
Tvsports	Information factored into d_interests_correlate variable
Exercise	Information factored into d_interests_correlate variable
Dining	Information factored into d_interests_correlate variable
Museums	Information factored into d_interests_correlate variable
Art	Information factored into d_interests_correlate variable
Hiking	Information factored into d_interests_correlate variable
Gaming	Information factored into d_interests_correlate variable
Clubbing	Information factored into d_interests_correlate variable
Reading	Information factored into d_interests_correlate variable
Tv	Information factored into d_interests_correlate variable
Theater	Information factored into d_interests_correlate variable
Movies	Information factored into d_interests_correlate variable
Concerts	Information factored into d_interests_correlate variable
Music	Information factored into d_interests_correlate variable
Shopping	Information factored into d_interests_correlate variable
Yoga	Information factored into d_interests_correlate variable
D_sports	Information factored into d_interests_correlate variable
D_tvsports	Information factored into d_interests_correlate variable
D_exercise	Information factored into d_interests_correlate variable
D_dining	Information factored into d_interests_correlate variable
D_museums	Information factored into d_interests_correlate variable
D_art	Information factored into d_interests_correlate variable
D_hiking	Information factored into d_interests_correlate variable
D_gaming	Information factored into d_interests_correlate variable
D_clubbing	Information factored into d_interests_correlate variable
D_reading	Information factored into d_interests_correlate variable
D_tv	Information factored into d_interests_correlate variable
D_theater	Information factored into d_interests_correlate variable
D_movies	Information factored into d_interests_correlate variable
D_concerts	Information factored into d_interests_correlate variable
D_music	Information factored into d_interests_correlate variable
D_shopping	Information factored into d_interests_correlate variable
D_yoga	Information factored into d_interests_correlate variable

Interests_correlate	Information contained in binned version of vairable d_interests_correlate
Like	Information contained in combination variable d_like
Guess_prob_liked	Information contained in combination variable d_guess_prob_liked
Decision	Leaker variable
Decision_o	Leaker variable
Attractive_o	Information contained in combination variable d_attractive
Attractive	Information contained in combination variable d_attractive
Sincere_o	Information contained in combination variable d_sincere
Sincere	Information contained in combination variable d_sincere
Intelligence_o	Information contained in combination variable d_intelligence
Intelligence	Information contained in combination variable d_intelligence
Funny_o	Information contained in combination variable d_funny
Funny	Information contained in combination variable d_funny
Ambitious_o	Information contained in combination variable d_ambition
Ambition	Information contained in combination variable d_ambition
Attractive_partner	Information contained in combination variable d_attractive_partner
Sincere_partner	Information contained in combination variable d_sincere_partner
Funny_partner	Information contained in combination variable d_funny_partner
Ambition_partner	Information contained in combination variable d_ambition_partner
Shared_interests_partner	Information contained in combination variable d_shared_interests_partner
D_shared_interests_o	Information better represented by interests_correlate variable
D_shared_interests_partner	Information better represented by interests_correlate variable
Expected_happy_with_sd_people	Information contained in combination variable d_expected_happy_with_sd_people
Expected_num_interested_in_me	Information contained in combination variable d_expected_num_interested_in_me
Expected_num_matches	Information contained in combination variable d_expected_num_matches
Race_o	Information redundant to race variable
Wave	Used as a round identification variable, not a predictive feature

Features Kept:

Gender

Figure 1 indicates that there close to equivalent numbers of males and females in the dataset. Likewise, the probability of matching is close to equivalent regardless of gender. This is to be expected because this study only allowed for heterogeneity in matches, so the same number of males and females matched (Fisman et al., 2006).

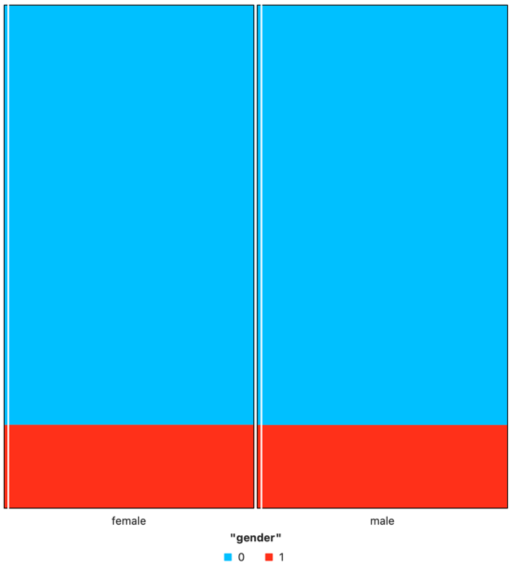


Figure 1: A mosaic of the gender feature segmented by whether the person matched with someone else.

Age Difference

As there is a greater age difference between the two people on the speed date, there is less of a chance for those two people to match (Figure 2). Additionally, while there are a similar number of couples in the data set with age differences in the lower 3 bins, there are fewer couples that had an age difference of 7 years or greater (Figure 2). This is due to the fact that most people in the dataset were of college age making smaller age gaps more likely (Fisman et al., 2006).

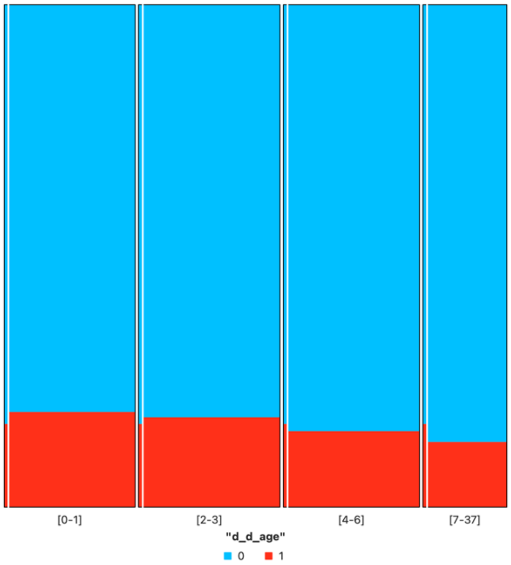


Figure 2: A mosaic of the d_d_age feature segmented by whether the person matched with someone else.

Race and Religion

Most people in the dataset are European/Caucasian-American with the second largest group being Asian/Pacific Islanders at half the amount of the European/Caucasian-American group (Figure 3). Black, Latino, and other categories represent about a fourth of the data (Figure 3). Individuals that are members of the smaller groupings (Black, Latino, and other) are more likely to have matched than those that are members of the larger groupings (European/Caucasian-American and Asian/Pacific Islander) potentially due to the lower level of representation in the dataset (Figure 3). Figure 4 shows that most of the couples were not of the same race and couples that were of the same race were slightly more likely to match than those that were not. Additionally, individuals that valued being with someone who was the same race as them less are more likely to match with someone (Figure 5). A similar pattern can be seen with religion. Individuals that place less value on people with someone of the same religion as them are more likely to match with someone (Figure 6). One reason for this could be that being open to being with someone of any race or religion opens the door to more people to potentially match with.

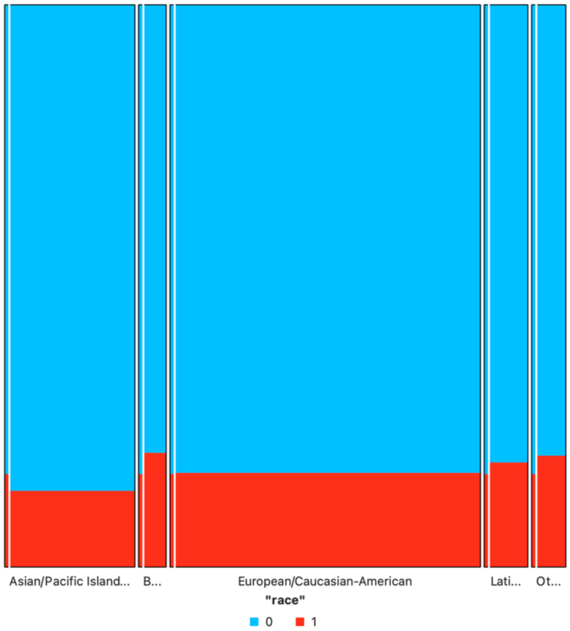
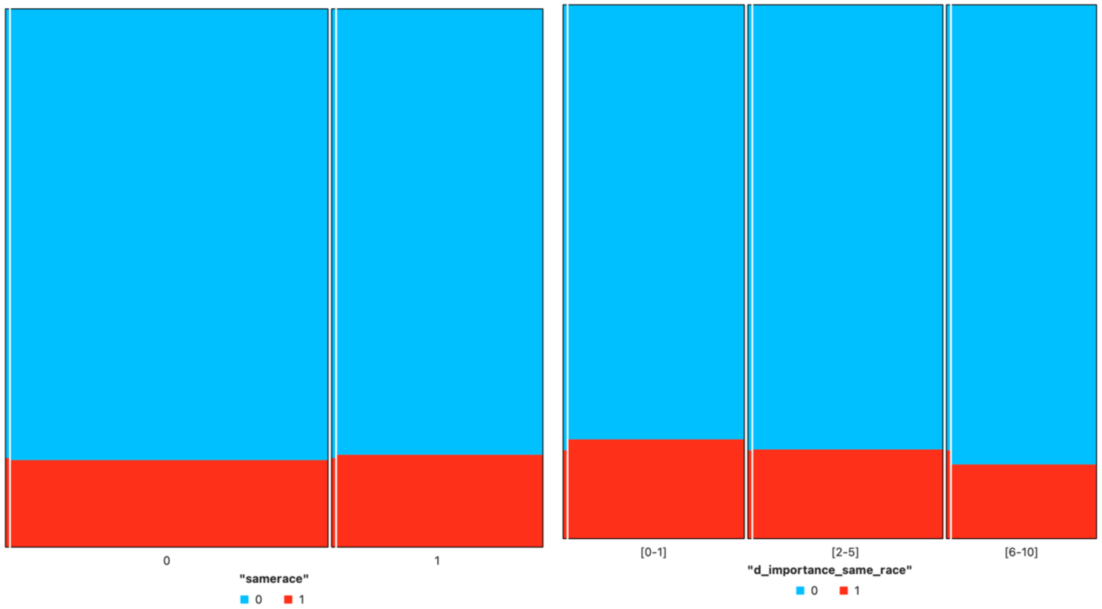


Figure 3: A mosaic of the race feature segmented by whether the person matched with someone else.



Figures 4 and 5: Mosaics of the samerace and d_importance_same_race features segmented by whether the person matched with someone else.

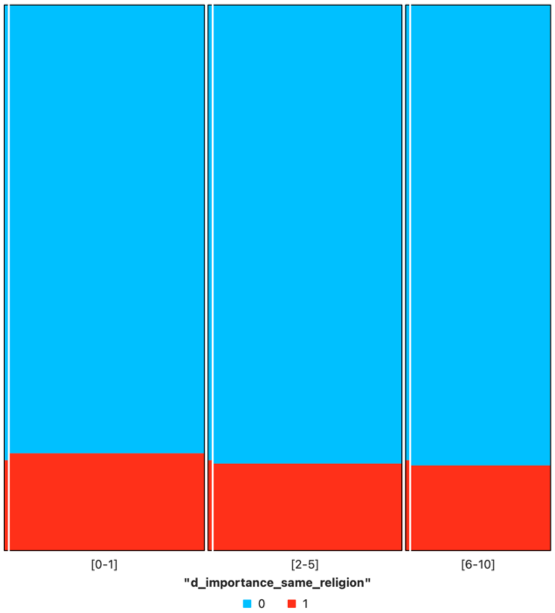


Figure 6: A mosaic of the d_importance_same_religion feature segmented by whether the person matched with someone else.

Quality Preferences

Figures 7-12 show the amount of weight the individuals put on each of the following qualities: attractiveness, sincerity, intelligence, humor, ambition, and shared interests. Each individual had 100 points to distribute to the importance of each of these qualities (Fisman et al., 2006). The overall trend for attractiveness, sincerity, and shared interests is as there is more weight put on that feature, the probability of matching decreases (Figures 7, 8, 12). This is expected as the less important features are, the easier it will be to find someone that matches the individual's desires. For intelligence, humor, and ambition, the trend is reversed so that putting more weight on each of those features has a higher probability of matching (Figures 9-11). As the weight of a quality increases past the value of weighting all attributes evenly, 16.67, there are fewer people that chose to attribute that level of weight to the quality in all qualities aside from attractiveness and intelligence (Figures 7-12). This finding is especially prevalent in ambition and shared interests where there are a lot of people who rate those qualities as less important and very few who rate that quality very important (Figures 11-12). Generally, people in this dataset seem to prefer attractiveness and intelligence and not value ambition or shared interests as much (Figures 7-12). These exact patterns are mirrored in the variables d_attractive_important, d_sincere_important, d_intelligence_important, d_funny_important, d_ambition_important, and d_shared interests_important, respectively. The only difference is the plotted group of variables represents the partner's weights while the previously listed variables represent the self-decided weights.

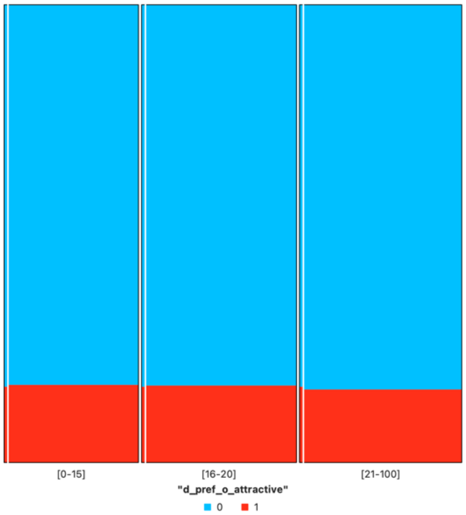


Figure 7

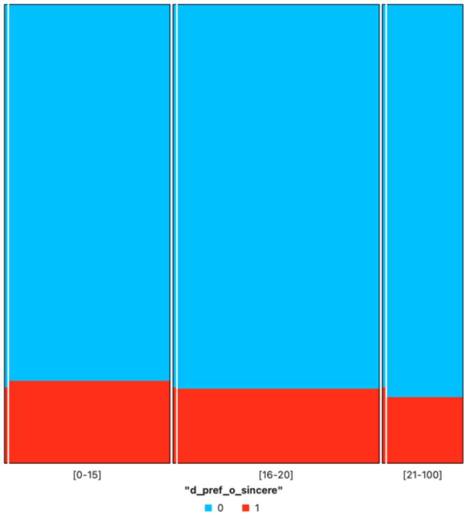
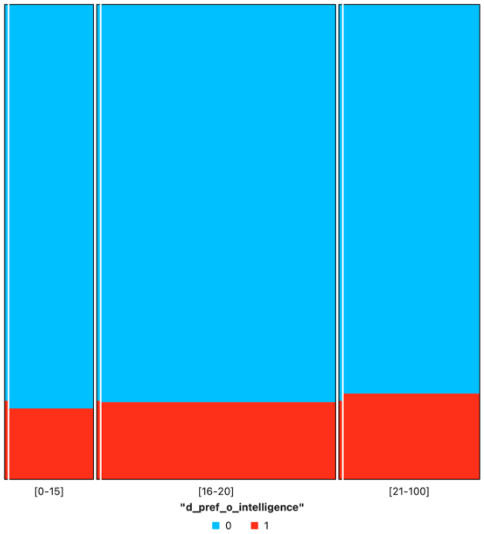


Figure 8



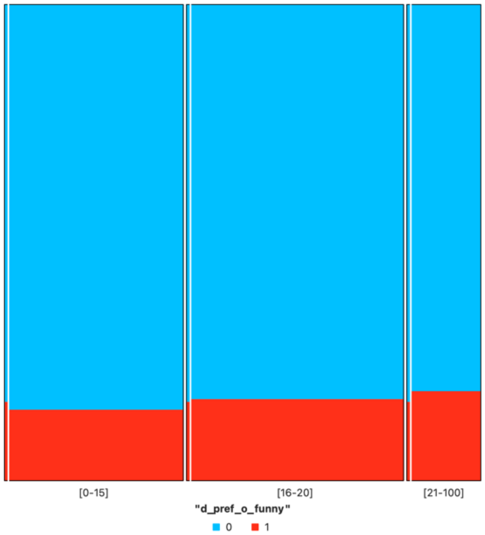
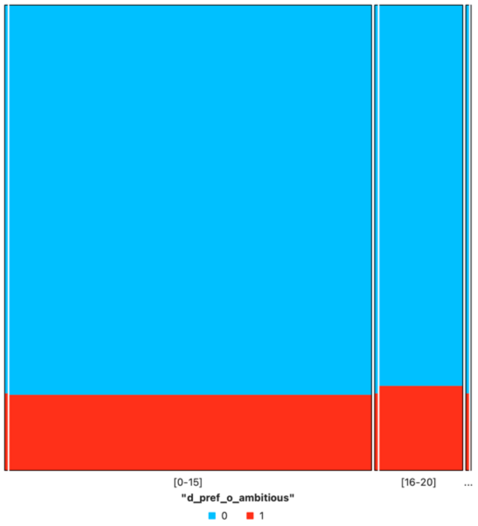


Figure 9 *Figure 10*



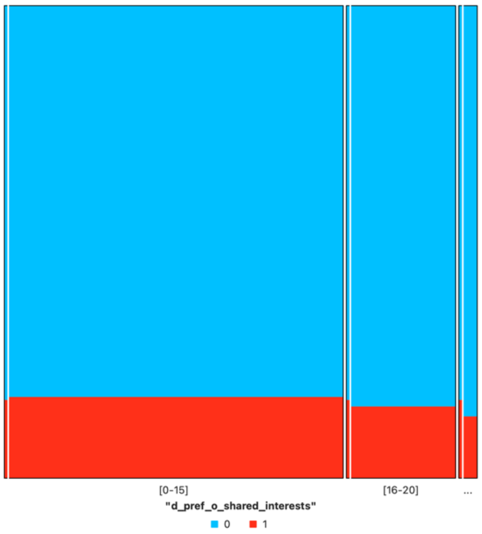


Figure 11 *Figure 12*

Figures 7-12: A mosaics of the d_pref_o_attractive, d_pref_o_sincere, d_pref_o_intelligence, d_pref_o_funny, d_pref_o_ambitious, and d_pref_o_shared_interests features segmented by whether the person matched with someone else.

Quality Scores Given to/Received by Partner

As expected, partners that were rated highly in each of these 5 areas are at a definite advantage probability wise when it comes to matching with someone (Figures 13-17). Also as expected, most individuals rated their partner in the middle range of between a 6 and 8 out of 10 (Figures 13-17). Attractiveness was the score with which people were harshest. Most people were scored in the lowest range and the least people in the highest range, while intelligence was the least harsh scoring with the fewest people in the lowest range and the most people in the highest range (Figures 13 and 15).

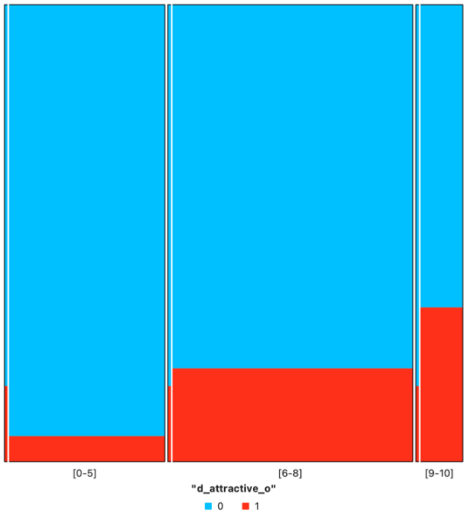


Figure 13

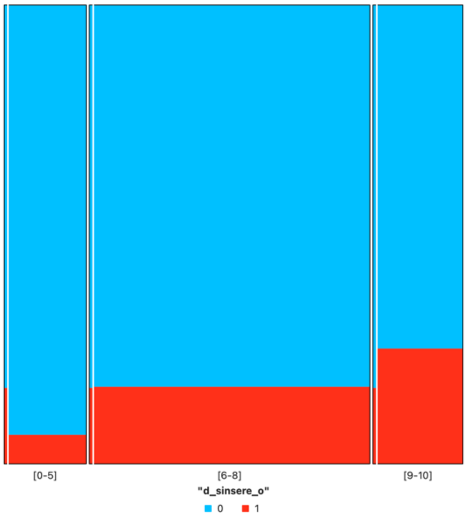


Figure 14

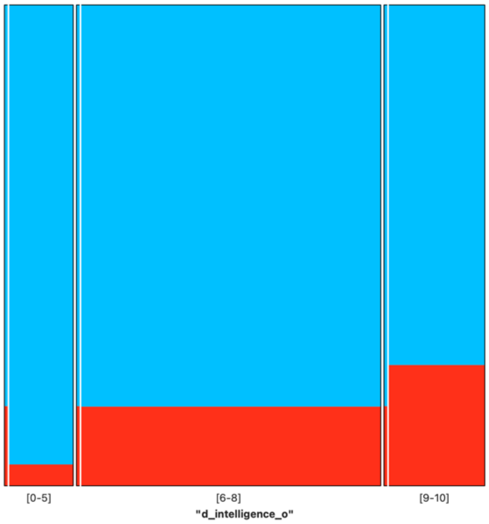


Figure 15

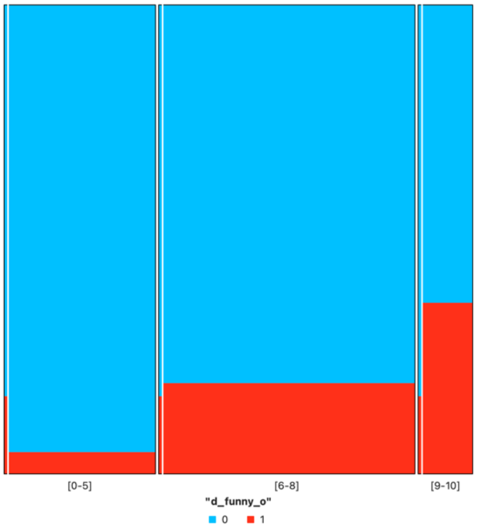


Figure 16

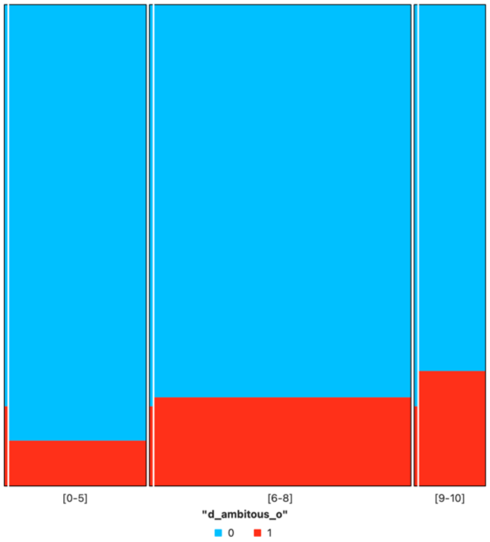


Figure 17

Figures 13-17: A mosaics of the d_attractive_o, d_sincere_o, d_intelligence_o, d_funny_o, and d_ambition_o, features segmented by whether the person matched with someone else.

Self-Given Quality Scores

When comparing Figures 18-22 to Figures 13-17, the scores given by an individual to themselves tend to be higher than the scores given to their partner. There are fewer people who rate themselves in the lowest category for each of the qualities when compared to the number of people who rated their partner in the lowest category (Figures 13-22). For attractiveness and intelligence, people who scored themselves higher are at a higher probability of matching (Figures 18 and 20). For sincerity and humor, people who scored themselves lower have a higher probability of matching (Figures 19 and 21). For ambition, people are about as likely to match regardless of how they scored themselves (Figure 22).

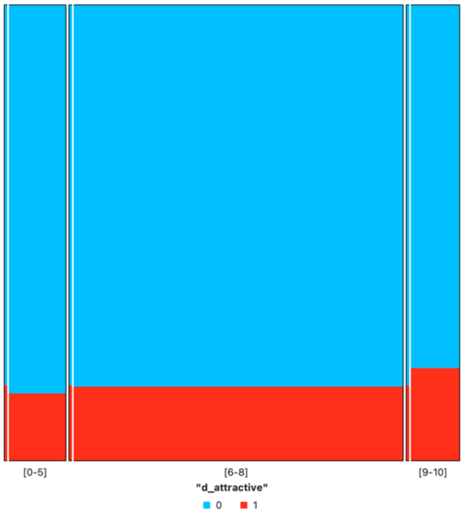


Figure 18

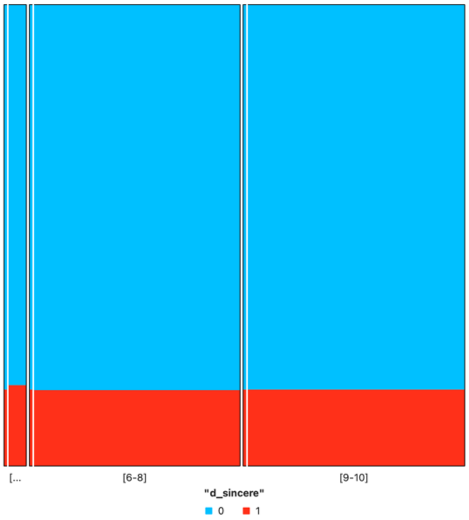


Figure 19

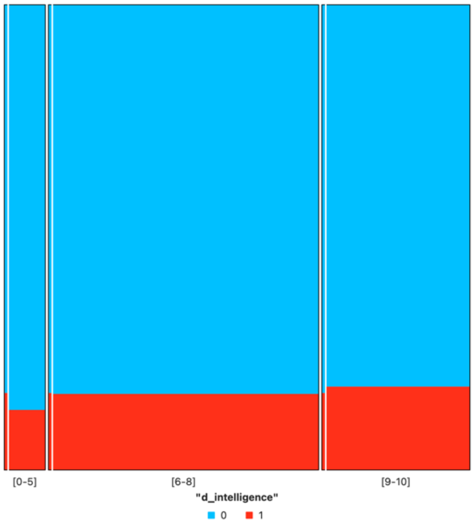


Figure 20

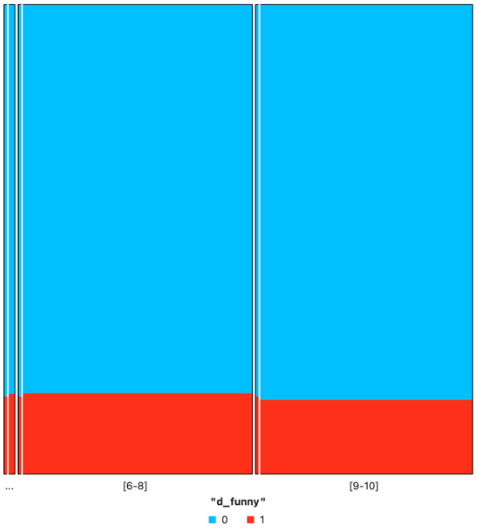


Figure 21

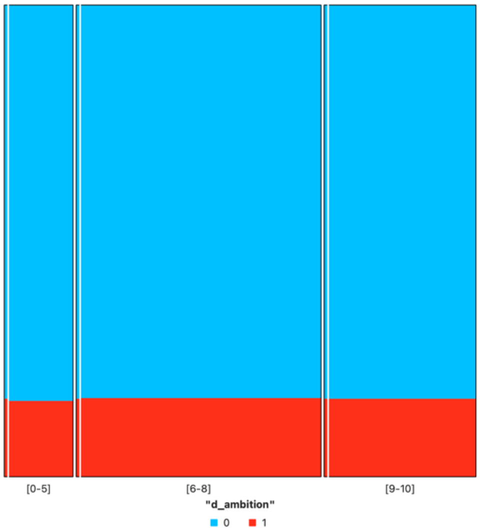


Figure 22

Figures 18-22: A mosaics of the d_attractive, d_sincere, d_intelligence, d_funny, and d_ambition, features segmented by whether the person matched with someone else.

Shared Interests

Figure 23 shows that there is close to an even distribution of people between each of the three groupings: different interests, some similar interests, and several similar interests. Those with different interests and only some similar interests have close to the same probability of matching (Figure 23). As expected, couples with several similar interests are more likely to match (Figure 23).

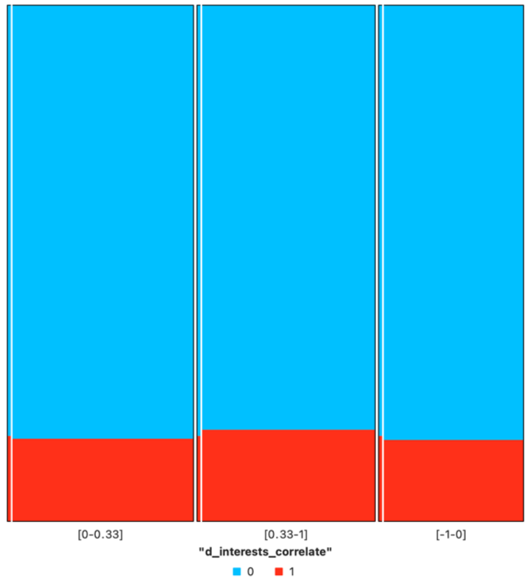


Figure 23: A mosaic of the d_inerests_correlate feature segmented by whether the person matched with someone else.

Expectations

People that had higher expectations regarding what they would get out of the speed dating experience were more likely to have matched with someone (Figures 24-26). Figure 24 shows how happy individuals thought they would be with people they met during the speed dating experience. Those who did not expect to be happy had a lower probability of matching with someone than those who expected to be happier (Figure 24). This is likely because those who expected to be happy with people they met took the event more seriously than those who did not expect to be happy with who they met. Also, those who thought that a higher number of people were interested in them were more likely to match with someone (Figure 25). Similarly, people who had a higher expectation regarding the number of matches they got had a higher chance of matching (Figure 26). This confidence regarding interest and matches likely carried over to the event, causing more people to want to match with those individuals.

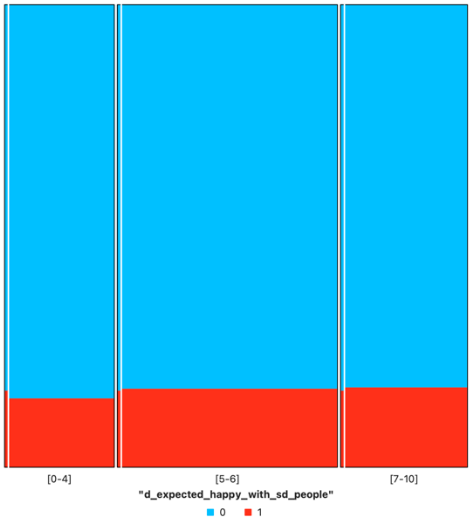


Figure 24

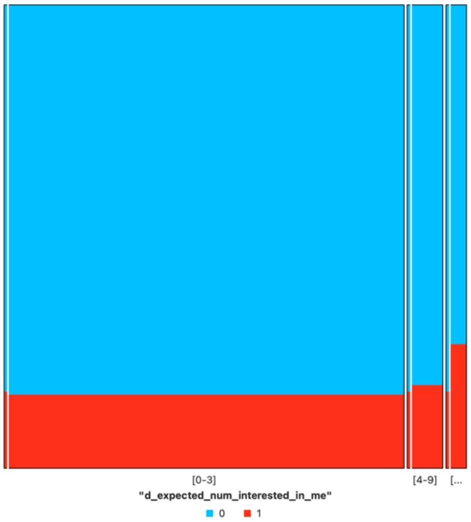


Figure 25

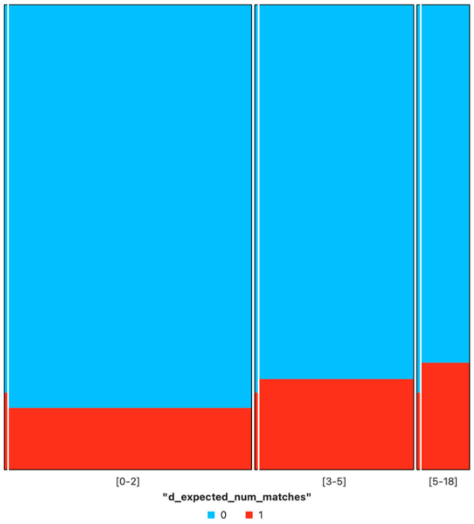


Figure 26

Figures 24-26: A mosaic of the d_expected_happy_with_sd_people, d_expected_num_interested_in_me, d_expected_num_matches features segmented by whether the person matched with someone else.

Ending Feelings

Figure 27 shows individuals who are more confident that their partner liked them are more likely to have matched with that person. Similarly, individuals who liked their partner more are more likely to have matched with their partner. This makes sense since if someone likes someone else and feels like someone else likes them, they are more likely to request a match.

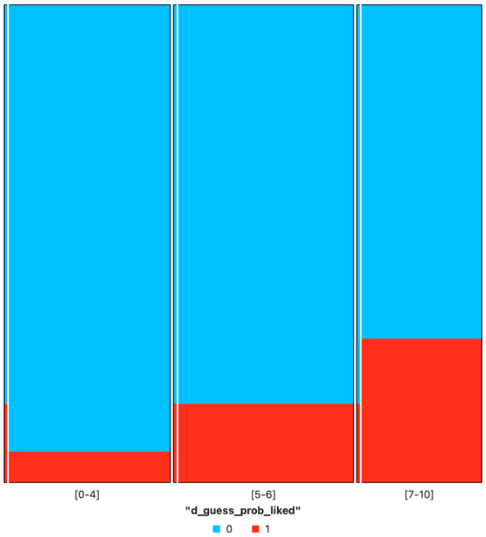


Figure 27

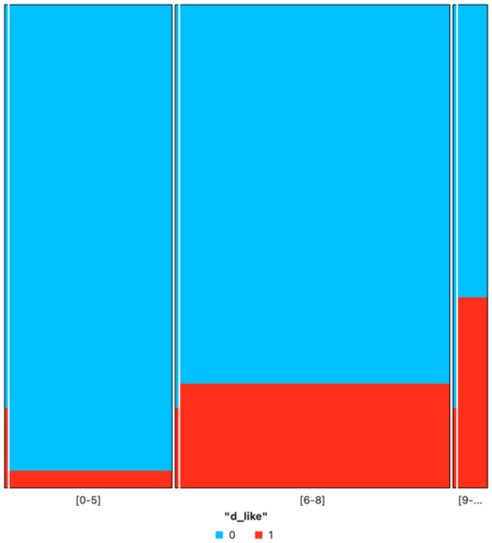


Figure 28

Figures 27 and 28: A mosaics of the d_guess_prob_liked and d_like features segmented by whether the person matched with someone else.

Past Meetings

According to Figure 29, most people had not met the person they were on a date with prior to the speed dating event. However, people that had met before were more likely to match than people who had not met (Figure 29). This is likely because people who have met already know some things about each other, leading to what could be a more natural and enjoyable speed date.

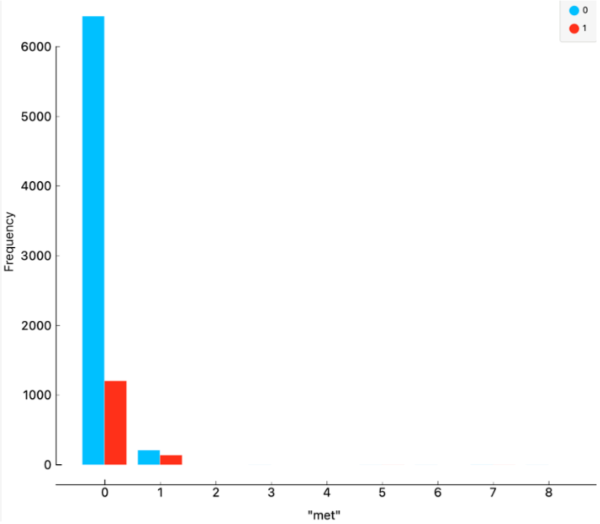


Figure 29: Histogram of the met feature broken up by whether the person matched with someone else.

Preprocessing

The features outlined in the “Features Eliminated” table were all dropped from the data set. The remaining numeric feature: met, was normalized to the [0,1] interval. All remaining categorical features: gender, d_d_age, race, samerace, d_importance_same_race, d_importance_same_religion, d_pref_o_attractive, d_pref_o_sincere, d_pref_o_intelligence, d_pref_o_funny, d_pref_o_ambitious, d_pref_o_shared_interests, d_attractive_o, d_sincere_o, d_intelligence_o, d_funny_o, d_ambitious_o, d_attractive, d_sincere, d_intelligence, d_ambition, d_attractive_partner, d_sincere_partner, d_intelligence_partner, d_funny_partner, d_ambition_partner, d_interests_correlated, d_expected_happy_with_sd_people, d_expected_num_interested_in_me, d_expected_num_matches, d_like, d_guess_prob_liked, d_attractive_important, d_sincere_important, d_intelligence_important, d_funny_important, d_ambition_important, and d_shared_interests_important, were continuized so the most frequent class was the base to reduce multicollinearity. The average value was imputed for any numeric feature and the mode was imputed for any categorical feature. The training set used in each of the models consisted of 80% of the data and the test set consisted of 20% of the data. All models will be evaluated via cross-validation with 10 folds and use the same training and testing set. The six models tested are decision trees, logistic regression, support vector machines (SVM), neural networks, adaptive boosting, and random forests.

Models

**Disclaimer*

All six of the models were split into groups of two to divide the workload among our group. Because of this separation, the three different sets of models were built on different samples of the data since it is not possible to set a random seed in Orange. Therefore, when we compiled all the models on one Orange file, the scores differed by a small margin of error. However, the ranking of the models' scores from worst to best remained the same, so we were confident that our parameters were still producing the optimal results for each model.

Decision Tree

The first of the six models we built was the decision tree. We built our first tree using the default parameters of a minimum of two instances in leaves, do not split subsets smaller than five, limit the maximal tree depth to 100, and stop when the majority reaches 95%. This preliminary decision tree was made of 1,239 nodes and 783 leaves, with no match (0) as the target. The first split was based on the d_like variable split into bins [0-5], [6-8], and [9-10]. The lower the score given for the d_like score, meaning the partner did not like their date after briefly speaking with them, the more likely the dater was to leave without matching with them. The next splits were made using the d_funny_o and d_attractive variables. The higher the partner rated their date based on how funny they found them, the more likely they were to leave with a match. The higher the attractiveness of a dater, the more likely they were to leave with a match as well. This decision tree resulted in a base case AUC of 0.558. Figure 30 below shows the first 4 levels of our preliminary tree to visualize the splits occurring. It is important to note that we decided not to induce a binary tree because many of our important variables have bins that fall into three categories rather than two. Therefore, our predictions will be most accurate if we do not force this constraint on the decision tree. Given this decision tree, a dater rated [0-5] for how much the partner liked them would be the most likely cause of leaving without a match, and someone rated high for likeability, attractiveness, and humor by their partner would be most likely to leave with a match.

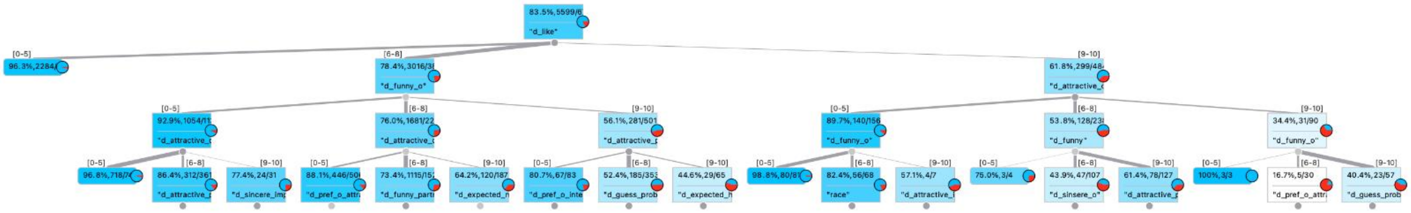
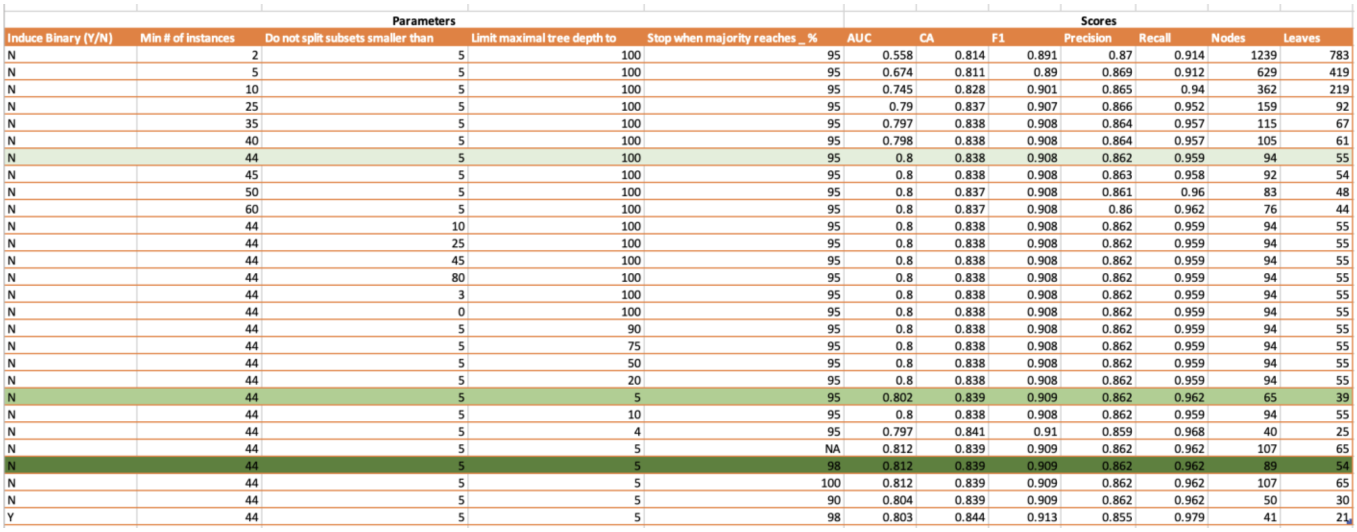


Figure 30: Preliminary decision tree using default parameters

Figure 31 below displays all the parameters changed individually to improve the AUC score of the decision tree. Once a parameter was optimized, it was highlighted in green to show which value would be used going forward when tuning the next parameter. The increasing darkness of the green corresponds to a higher AUC score. When the stop when majority reaches _% parameter was eliminated or changed to either 98% or 100%, the AUC score remained 0.812. We prioritized the parameter that resulted in fewer nodes and leaves to reduce complexity, which was the value of 98%. Given this information, the best decision tree had 44 minimum instances, did not split the subsets smaller than five, limited the maximal tree depth to five, and stopped when the majority reached 98%. This model produced an AUC score of 0.812.



		Parameters						Scores			
Induce Binary (Y/N)	Min # of instances	Do not split subsets smaller than	Limit maximal tree depth to	Stop when majority reaches _ %	AUC	CA	F1	Precision	Recall	Nodes	Leaves
N	2	5	100	95	0.558	0.814	0.891	0.87	0.914	1239	783
N	5	5	100	95	0.674	0.811	0.89	0.869	0.912	629	419
N	10	5	100	95	0.745	0.828	0.901	0.865	0.94	362	219
N	25	5	100	95	0.79	0.837	0.907	0.866	0.952	159	92
N	35	5	100	95	0.797	0.838	0.908	0.864	0.957	115	67
N	40	5	100	95	0.798	0.838	0.908	0.864	0.957	105	61
N	44	5	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	45	5	100	95	0.8	0.838	0.908	0.863	0.958	92	54
N	50	5	100	95	0.8	0.837	0.908	0.861	0.96	83	48
N	60	5	100	95	0.8	0.837	0.908	0.86	0.962	76	44
N	44	10	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	25	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	45	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	80	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	3	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	0	100	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	90	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	75	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	50	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	20	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	5	95	0.802	0.839	0.909	0.862	0.962	65	39
N	44	5	10	95	0.8	0.838	0.908	0.862	0.959	94	55
N	44	5	4	95	0.797	0.841	0.91	0.859	0.968	40	25
N	44	5	5	NA	0.812	0.839	0.909	0.862	0.962	107	65
N	44	5	5	98	0.812	0.839	0.909	0.862	0.962	89	54
N	44	5	5	100	0.812	0.839	0.909	0.862	0.962	107	65
N	44	5	5	90	0.804	0.839	0.909	0.862	0.962	50	30
Y	44	5	5	98	0.803	0.844	0.913	0.855	0.979	41	21

Figure 31: Table tuning the parameters of the decision tree and the corresponding scores

Comparing the final tree’s leaves to the preliminary tree’s leaves will show us the variables that end up being the most important. Figure 32 below shows the first four levels of the best-scoring decision tree. The first variable to split the data on is the d_funny_partner variable. When a dater did not rate their partner high in terms of their humor, that dater was less likely to match. The d_attractive variable again was an important factor in the matching decision. If a dater was rated low in terms of how attractive they are, they were less likely to leave the event with a match. The d_like variable provides binned scores for how a dater rated their partner on how much they liked them overall. This is a very subjective measure that incorporates a lot of different factors, making it difficult to predict how someone will be scored. The variable does behave as we would expect, because the lower a dater rated their partner in terms of likeability, the less likely they were to match. A person who was rated low for attractiveness and humor ([0-5]) would be 97.2% likely to leave the event without a match. Given the decision tree below, we identified humor, attractiveness, and overall likeability of a person to be the most important factors when determining if there would be a match between daters.

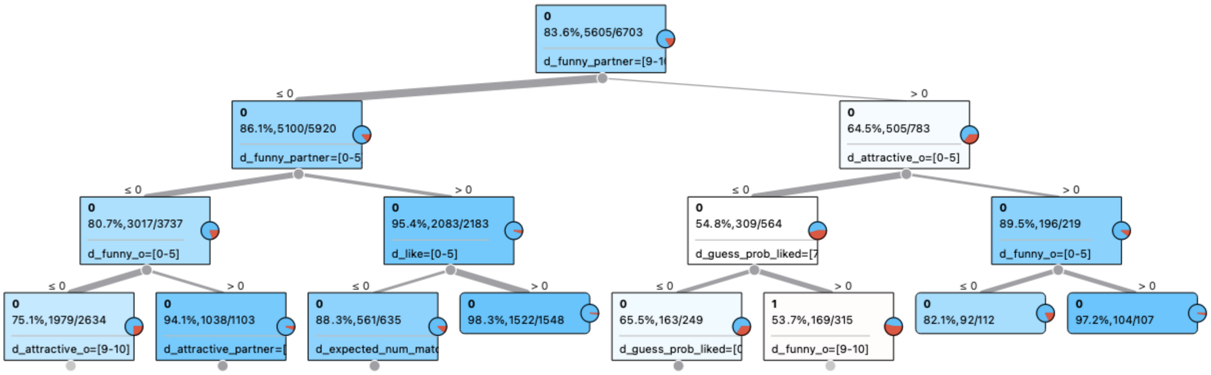
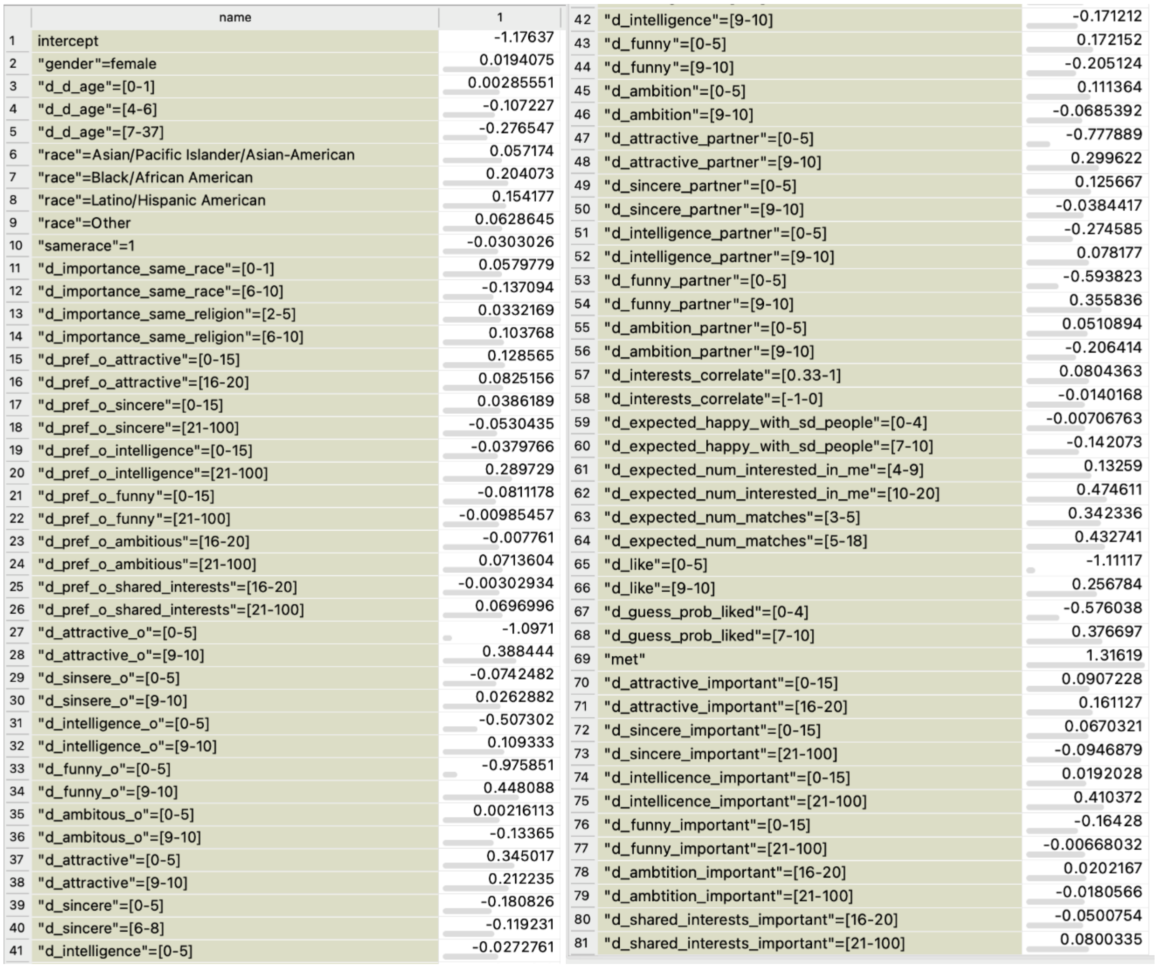


Figure 32: First 4 layers of the final decision tree

Logistic Regression

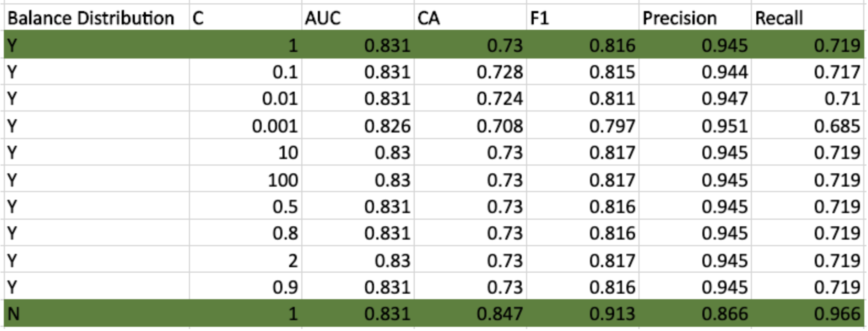
The next model constructed was our logistic regression model. We only used the Ridge (R2) regularization type for the logistic regression, because our model had a lot of large variables that we expected to impact its results. The logistic regression only requires the tuning of the C parameter within the range of 0.001 to 1,000. We began with a C value of 1, which produced the coefficients for the 80 variables shown below in Figure 33. We recognize having a low attractiveness rating of [0-5] and d_like [0-5] have the heaviest weightings of –1.097 and –1.11, respectively, for reducing the probability of leaving the event with a match. The d_funny_o [0-5] variable had the second highest weighting of –0.976, meaning that daters who were rated low in terms of their humor were more likely to leave without a match. On the contrary, the met variable had the highest positive impact on a couple matching of 1.316, which meant if the daters had met previously, then they were more likely to match. The d_expected_num_interested_in_me [10-20] had the next highest impact on leaving with a match with its positive coefficient of 0.475. From this we conclude that daters who were more confident that they would get matches were more likely to end up leaving with a match. The d_intelligence_important [21-100] coefficient of 0.410 shows that daters who saw value in how intelligent their date was, were more likely to leave with a match.



	name	1
1	intercept	-1.17637
2	"gender"=female	0.0194075
3	"d_d_age"=[0-1]	0.00285551
4	"d_d_age"=[4-6]	-0.107227
5	"d_d_age"=[7-37]	-0.276547
6	"race"=Asian/Pacific Islander/Asian-American	0.057174
7	"race"=Black/African American	0.204073
8	"race"=Latino/Hispanic American	0.154177
9	"race"=Other	0.0628645
10	"samerace"=1	-0.0303026
11	"d_importance_same_race"=[0-1]	0.0579779
12	"d_importance_same_race"=[6-10]	-0.137094
13	"d_importance_same_religion"=[2-5]	0.0332169
14	"d_importance_same_religion"=[6-10]	0.103768
15	"d_pref_o_attractive"=[0-15]	0.128565
16	"d_pref_o_attractive"=[16-20]	0.0825156
17	"d_pref_o_sincere"=[0-15]	0.0386189
18	"d_pref_o_sincere"=[21-100]	-0.0530435
19	"d_pref_o_intelligence"=[0-15]	-0.0379766
20	"d_pref_o_intelligence"=[21-100]	0.289729
21	"d_pref_o_funny"=[0-15]	-0.0811178
22	"d_pref_o_funny"=[21-100]	-0.00985457
23	"d_pref_o_ambitious"=[16-20]	-0.007761
24	"d_pref_o_ambitious"=[21-100]	0.0713604
25	"d_pref_o_shared_interests"=[16-20]	-0.00302934
26	"d_pref_o_shared_interests"=[21-100]	0.0696996
27	"d_attractive_o"=[0-5]	-1.0971
28	"d_attractive_o"=[9-10]	0.388444
29	"d_sinsere_o"=[0-5]	-0.0742482
30	"d_sinsere_o"=[9-10]	0.0262882
31	"d_intelligence_o"=[0-5]	-0.507302
32	"d_intelligence_o"=[9-10]	0.109333
33	"d_funny_o"=[0-5]	-0.975851
34	"d_funny_o"=[9-10]	0.448088
35	"d_ambitous_o"=[0-5]	0.00216113
36	"d_ambitous_o"=[9-10]	-0.13365
37	"d_attractive"=[0-5]	0.345017
38	"d_attractive"=[9-10]	0.212235
39	"d_sincere"=[0-5]	-0.180826
40	"d_sincere"=[6-8]	-0.119231
41	"d_intelligence"=[0-5]	-0.0272761
42	"d_intelligence"=[9-10]	-0.171212
43	"d_funny"=[0-5]	0.172152
44	"d_funny"=[9-10]	-0.205124
45	"d_ambition"=[0-5]	0.111364
46	"d_ambition"=[9-10]	-0.0685392
47	"d_attractive_partner"=[0-5]	-0.777889
48	"d_attractive_partner"=[9-10]	0.299622
49	"d_sincere_partner"=[0-5]	0.125667
50	"d_sincere_partner"=[9-10]	-0.0384417
51	"d_intelligence_partner"=[0-5]	-0.274585
52	"d_intelligence_partner"=[9-10]	0.078177
53	"d_funny_partner"=[0-5]	-0.593823
54	"d_funny_partner"=[9-10]	0.355836
55	"d_ambition_partner"=[0-5]	0.0510894
56	"d_ambition_partner"=[9-10]	-0.206414
57	"d_interests_correlate"=[0.33-1]	0.0804363
58	"d_interests_correlate"=[-1-0]	-0.0140168
59	"d_expected_happy_with_sd_people"=[0-4]	-0.00706763
60	"d_expected_happy_with_sd_people"=[7-10]	-0.142073
61	"d_expected_num_interested_in_me"=[4-9]	0.13259
62	"d_expected_num_interested_in_me"=[10-20]	0.474611
63	"d_expected_num_matches"=[3-5]	0.342336
64	"d_expected_num_matches"=[5-18]	0.432741
65	"d_like"=[0-5]	-1.11117
66	"d_like"=[9-10]	0.256784
67	"d_guess_prob_liked"=[0-4]	-0.576038
68	"d_guess_prob_liked"=[7-10]	0.376697
69	"met"	1.31619
70	"d_attractive_important"=[0-15]	0.0907228
71	"d_attractive_important"=[16-20]	0.161127
72	"d_sincere_important"=[0-15]	0.0670321
73	"d_sincere_important"=[21-100]	-0.0946879
74	"d_intellicence_important"=[0-15]	0.0192028
75	"d_intellicence_important"=[21-100]	0.410372
76	"d_funny_important"=[0-15]	-0.16428
77	"d_funny_important"=[21-100]	-0.00668032
78	"d_ambtition_important"=[16-20]	0.0202167
79	"d_ambtition_important"=[21-100]	-0.0180566
80	"d_shared_interests_important"=[16-20]	-0.0500754
81	"d_shared_interests_important"=[21-100]	0.0800335

Figure 33: Coefficients of the preliminary logistic regression using C =1

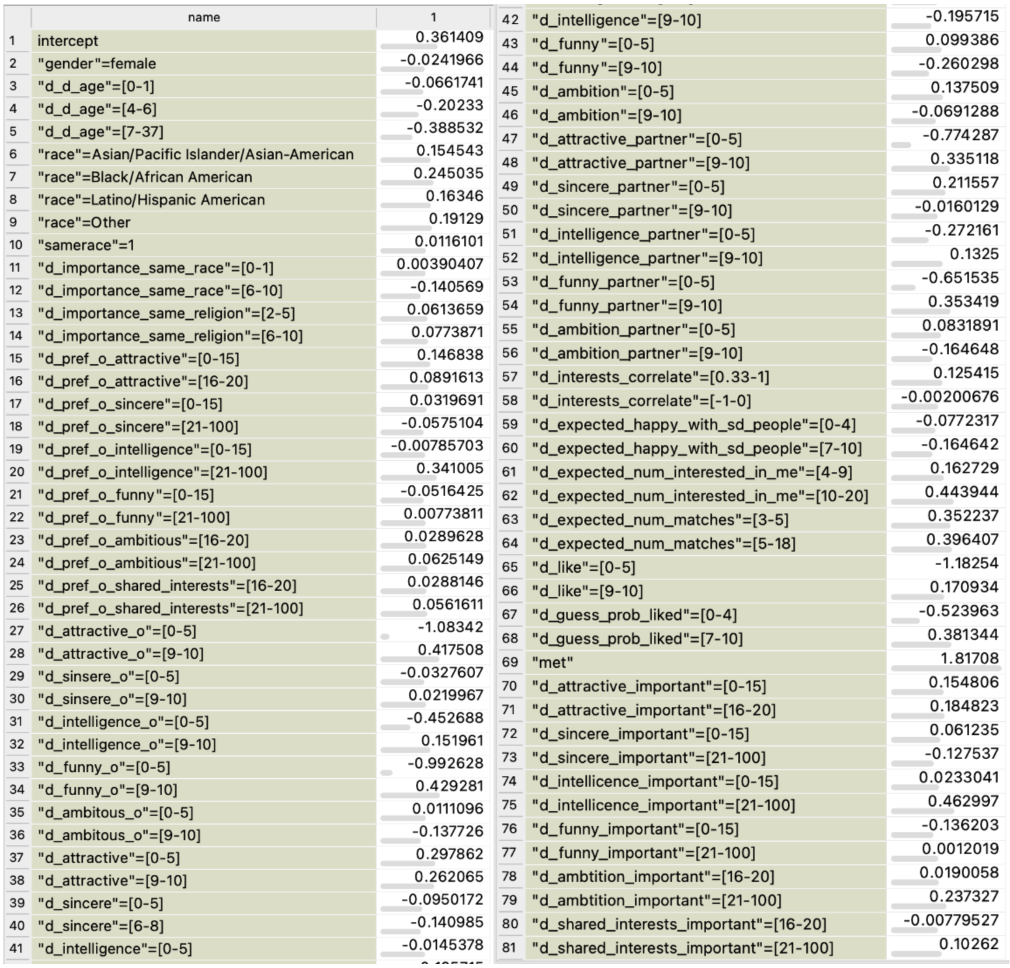
The table in Figure 34 shows the different AUC scores achieved by changing the value of C. The AUC scores did not change much, with a score of 0.831 being the highest score achieved. Orange provides the option to balance the distribution, which means that it would under sample the negatives because our dataset has more leave instances (0) than match instances (1). Balancing the distribution with a C of 1 resulted in the highest AUC score of 0.831 and highest F1 score of 0.913. Therefore, we chose these two parameters to create our final logistic regression model.



Balance Distribution	C	AUC	CA	F1	Precision	Recall
Y	1	0.831	0.73	0.816	0.945	0.719
Y	0.1	0.831	0.728	0.815	0.944	0.717
Y	0.01	0.831	0.724	0.811	0.947	0.71
Y	0.001	0.826	0.708	0.797	0.951	0.685
Y	10	0.83	0.73	0.817	0.945	0.719
Y	100	0.83	0.73	0.817	0.945	0.719
Y	0.5	0.831	0.73	0.816	0.945	0.719
Y	0.8	0.831	0.73	0.816	0.945	0.719
Y	2	0.83	0.73	0.817	0.945	0.719
Y	0.9	0.831	0.73	0.816	0.945	0.719
N	1	0.831	0.847	0.913	0.866	0.966

Figure 34: Table tuning the parameters of the logistic regression and the corresponding score

Figure 35 below displays the different coefficient results of the final logistic regression built. Similar to the preliminary logistic regression, d_attractive_o [0-5], d_like [0-5], and d_funny_o [0-5] had the largest negative impact on whether a dater would leave the event with a match. This model also recognized d_intelligence_important [21-100], met, and d_attractive_o [9-10] as being the most heavily weighted coefficients that lead to a higher probability of daters matching at the event. Overall, logistic regression places importance on some of the same variables that the decision tree did, but the results are more easily interpretable and its AUC score was slightly higher. Therefore, we prefer the logistic regression over the decision tree thus far in our model building.

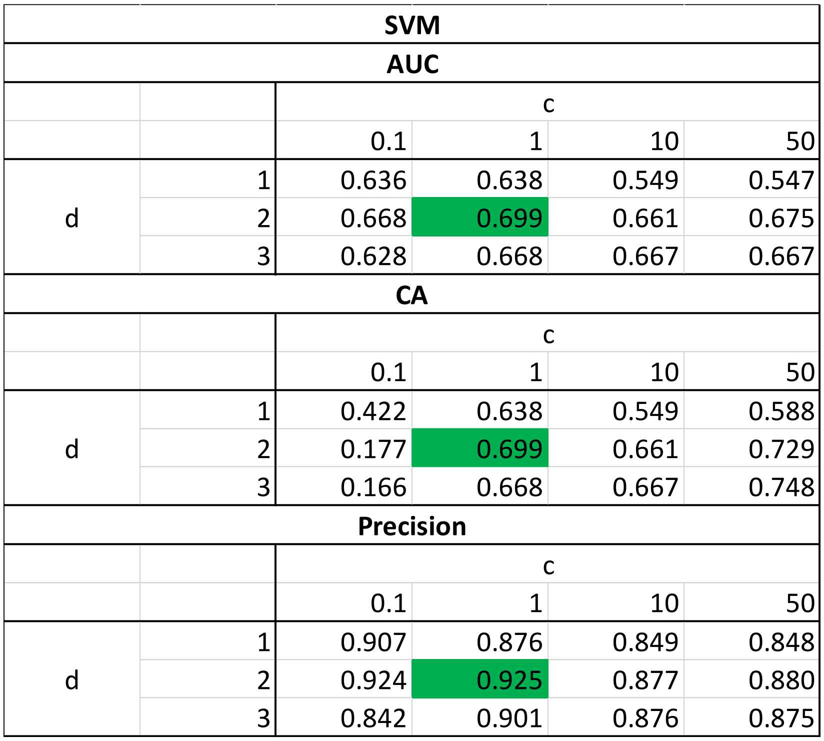


	name	1
1	intercept	0.361409
2	"gender"=female	-0.0241966
3	"d_d_age"=[0-1]	-0.0661741
4	"d_d_age"=[4-6]	-0.20233
5	"d_d_age"=[7-37]	-0.388532
6	"race"=Asian/Pacific Islander/Asian-American	0.154543
7	"race"=Black/African American	0.245035
8	"race"=Latino/Hispanic American	0.16346
9	"race"=Other	0.19129
10	"samerace"=1	0.0116101
11	"d_importance_same_race"=[0-1]	0.00390407
12	"d_importance_same_race"=[6-10]	-0.140569
13	"d_importance_same_religion"=[2-5]	0.0613659
14	"d_importance_same_religion"=[6-10]	0.0773871
15	"d_pref_o_attractive"=[0-15]	0.146838
16	"d_pref_o_attractive"=[16-20]	0.0891613
17	"d_pref_o_sincere"=[0-15]	0.0319691
18	"d_pref_o_sincere"=[21-100]	-0.0575104
19	"d_pref_o_intelligence"=[0-15]	-0.00785703
20	"d_pref_o_intelligence"=[21-100]	0.341005
21	"d_pref_o_funny"=[0-15]	-0.0516425
22	"d_pref_o_funny"=[21-100]	0.00773811
23	"d_pref_o_ambitious"=[16-20]	0.0289628
24	"d_pref_o_ambitious"=[21-100]	0.0625149
25	"d_pref_o_shared_interests"=[16-20]	0.0288146
26	"d_pref_o_shared_interests"=[21-100]	0.0561611
27	"d_attractive_o"=[0-5]	-1.08342
28	"d_attractive_o"=[9-10]	0.417508
29	"d_sinsere_o"=[0-5]	-0.0327607
30	"d_sinsere_o"=[9-10]	0.0219967
31	"d_intelligence_o"=[0-5]	-0.452688
32	"d_intelligence_o"=[9-10]	0.151961
33	"d_funny_o"=[0-5]	-0.992628
34	"d_funny_o"=[9-10]	0.429281
35	"d_ambitous_o"=[0-5]	0.0111096
36	"d_ambitous_o"=[9-10]	-0.137726
37	"d_attractive"=[0-5]	0.297862
38	"d_attractive"=[9-10]	0.262065
39	"d_sincere"=[0-5]	-0.0950172
40	"d_sincere"=[6-8]	-0.140985
41	"d_intelligence"=[0-5]	-0.0145378
42	"d_intelligence"=[9-10]	-0.195715
43	"d_funny"=[0-5]	0.099386
44	"d_funny"=[9-10]	-0.260298
45	"d_ambition"=[0-5]	0.137509
46	"d_ambition"=[9-10]	-0.0691288
47	"d_attractive_partner"=[0-5]	-0.774287
48	"d_attractive_partner"=[9-10]	0.335118
49	"d_sincere_partner"=[0-5]	0.211557
50	"d_sincere_partner"=[9-10]	-0.0160129
51	"d_intelligence_partner"=[0-5]	-0.272161
52	"d_intelligence_partner"=[9-10]	0.1325
53	"d_funny_partner"=[0-5]	-0.651535
54	"d_funny_partner"=[9-10]	0.353419
55	"d_ambition_partner"=[0-5]	0.0831891
56	"d_ambition_partner"=[9-10]	-0.164648
57	"d_interests_correlate"=[0.33-1]	0.125415
58	"d_interests_correlate"=[-1-0]	-0.00200676
59	"d_expected_happy_with_sd_people"=[0-4]	-0.0772317
60	"d_expected_happy_with_sd_people"=[7-10]	-0.164642
61	"d_expected_num_interested_in_me"=[4-9]	0.162729
62	"d_expected_num_interested_in_me"=[10-20]	0.443944
63	"d_expected_num_matches"=[3-5]	0.352237
64	"d_expected_num_matches"=[5-18]	0.396407
65	"d_like"=[0-5]	-1.18254
66	"d_like"=[9-10]	0.170934
67	"d_guess_prob_liked"=[0-4]	-0.523963
68	"d_guess_prob_liked"=[7-10]	0.381344
69	"met"	1.81708
70	"d_attractive_important"=[0-15]	0.154806
71	"d_attractive_important"=[16-20]	0.184823
72	"d_sincere_important"=[0-15]	0.061235
73	"d_sincere_important"=[21-100]	-0.127537
74	"d_intellicence_important"=[0-15]	0.0233041
75	"d_intellicence_important"=[21-100]	0.462997
76	"d_funny_important"=[0-15]	-0.136203
77	"d_funny_important"=[21-100]	0.0012019
78	"d_ambtition_important"=[16-20]	0.0190058
79	"d_ambtition_important"=[21-100]	0.237327
80	"d_shared_interests_important"=[16-20]	-0.00779527
81	"d_shared_interests_important"=[21-100]	0.10262

Figure 35: Coefficients of the finalized logistic regression using C = 1 with balanced class distribution

Support Vector Machine

The third model we built was a support vector machine. To find the best model, we adjusted the cost and the kernel function. We began this search with a cost of 0.1 and a linear kernel function, then tested a range of values for these parameters. Focusing on maximizing the area under the ROC curve, these values ranged from 0.1 to 50 for the cost and up to a degree of three for the polynomial kernel function. After adjusting the complexity of the model, we had our final result. By testing each different degree on different cost values, we found that the AUC was maximized with a cost of one and a degree of two. Using a Radius Basis Function did not improve the AUC score. The resulting areas under the ROC curve, classification accuracies, and precisions—for each of the twelve models tested—are shown in Figure 36.

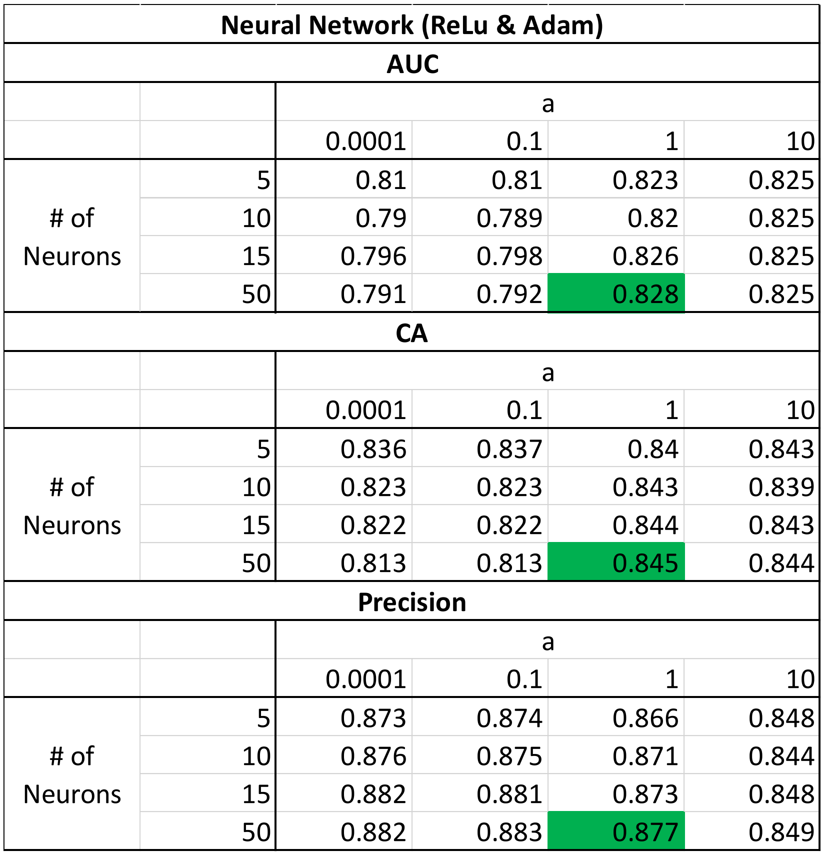


SVM					
AUC					
		c			
		0.1	1	10	50
d	1	0.636	0.638	0.549	0.547
	2	0.668	0.699	0.661	0.675
	3	0.628	0.668	0.667	0.667
CA					
		c			
		0.1	1	10	50
d	1	0.422	0.638	0.549	0.588
	2	0.177	0.699	0.661	0.729
	3	0.166	0.668	0.667	0.748
Precision					
		c			
		0.1	1	10	50
d	1	0.907	0.876	0.849	0.848
	2	0.924	0.925	0.877	0.880
	3	0.842	0.901	0.876	0.875

Figure 36: AUCs, classification accuracies, and precision values for the support vector machine models.

Neural Network

Our fourth model was a neural network. After testing many models, we found that using the ReLu activation function and the Adam solver consistently produced models with higher AUC values. The regularization parameter was tested over a range of values, and we determined that one was most suitable for maximizing the area under the ROC curve. Models with multiple layers did not improve the AUC, so we used the simplest model, which had 50 neurons and only one layer. The resulting areas under the ROC curve, classification accuracies, and precisions are shown in Figure 37.

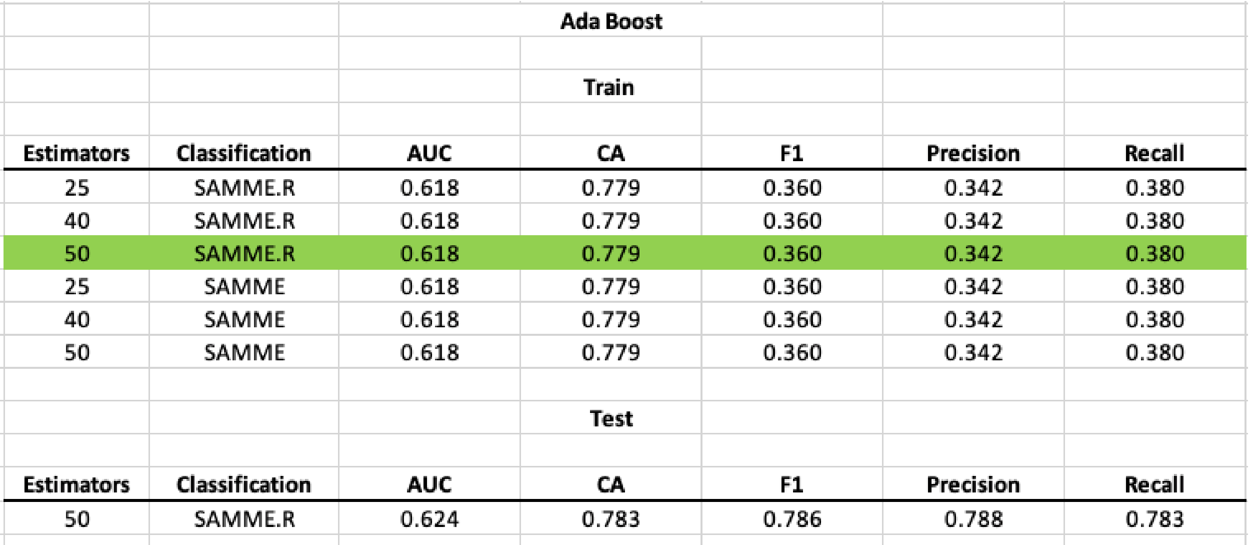


Neural Network (ReLu & Adam)					
AUC					
		a			
		0.0001	0.1	1	10
# of Neurons	5	0.81	0.81	0.823	0.825
	10	0.79	0.789	0.82	0.825
	15	0.796	0.798	0.826	0.825
	50	0.791	0.792	0.828	0.825
CA					
		a			
		0.0001	0.1	1	10
# of Neurons	5	0.836	0.837	0.84	0.843
	10	0.823	0.823	0.843	0.839
	15	0.822	0.822	0.844	0.843
	50	0.813	0.813	0.845	0.844
Precision					
		a			
		0.0001	0.1	1	10
# of Neurons	5	0.873	0.874	0.866	0.848
	10	0.876	0.875	0.871	0.844
	15	0.882	0.881	0.873	0.848
	50	0.882	0.883	0.877	0.849

Figure 37: AUCs, classification accuracies, and precision values for neural network models.

Adaptive Boosting

Our ADA Boost model was one of our lowest models in terms of AUC. We tried many parameters on our training set from 25-50 estimators, but all ended up resulting with the same scores. We tested the classification parameters as well, switching from SAMME to SAMME.R. After changing both the estimator and classification parameters, we were unable to find any results outside of what are posted below. We decided to use 50 estimators as suggested in class, with the SAMME.R classification. Our ADA Boost training model produced an AUC of 0.618 in our training set and 0.624 in our test set.



Ada Boost

Train

Estimators	Classification	AUC	CA	F1	Precision	Recall
25	SAMME.R	0.618	0.779	0.360	0.342	0.380
40	SAMME.R	0.618	0.779	0.360	0.342	0.380
50	SAMME.R	0.618	0.779	0.360	0.342	0.380
25	SAMME	0.618	0.779	0.360	0.342	0.380
40	SAMME	0.618	0.779	0.360	0.342	0.380
50	SAMME	0.618	0.779	0.360	0.342	0.380

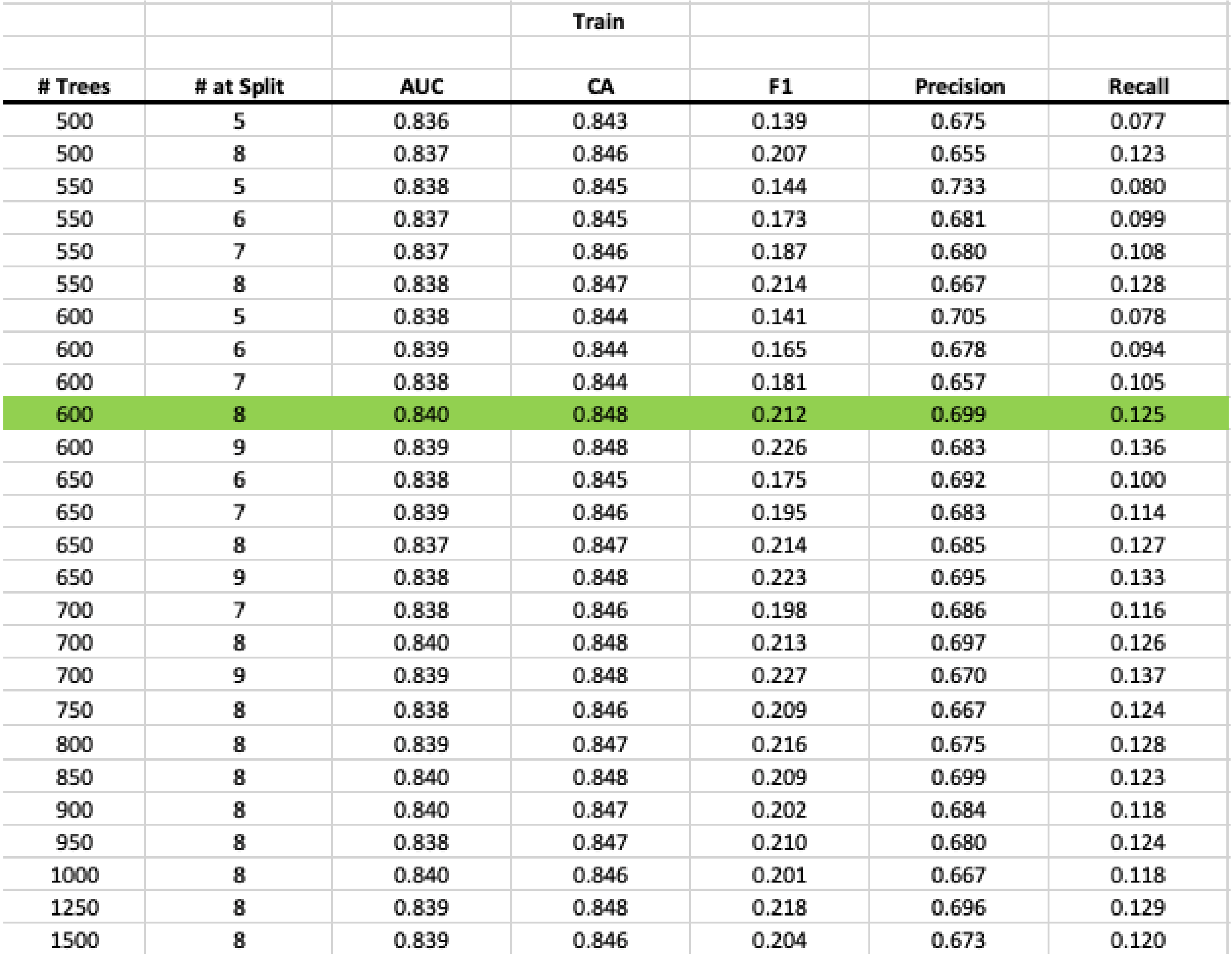
Test

Estimators	Classification	AUC	CA	F1	Precision	Recall
50	SAMME.R	0.624	0.783	0.786	0.788	0.783

Figure 38: This figure shows the Ada Boost training and testing results found on Orange.

Random Forest

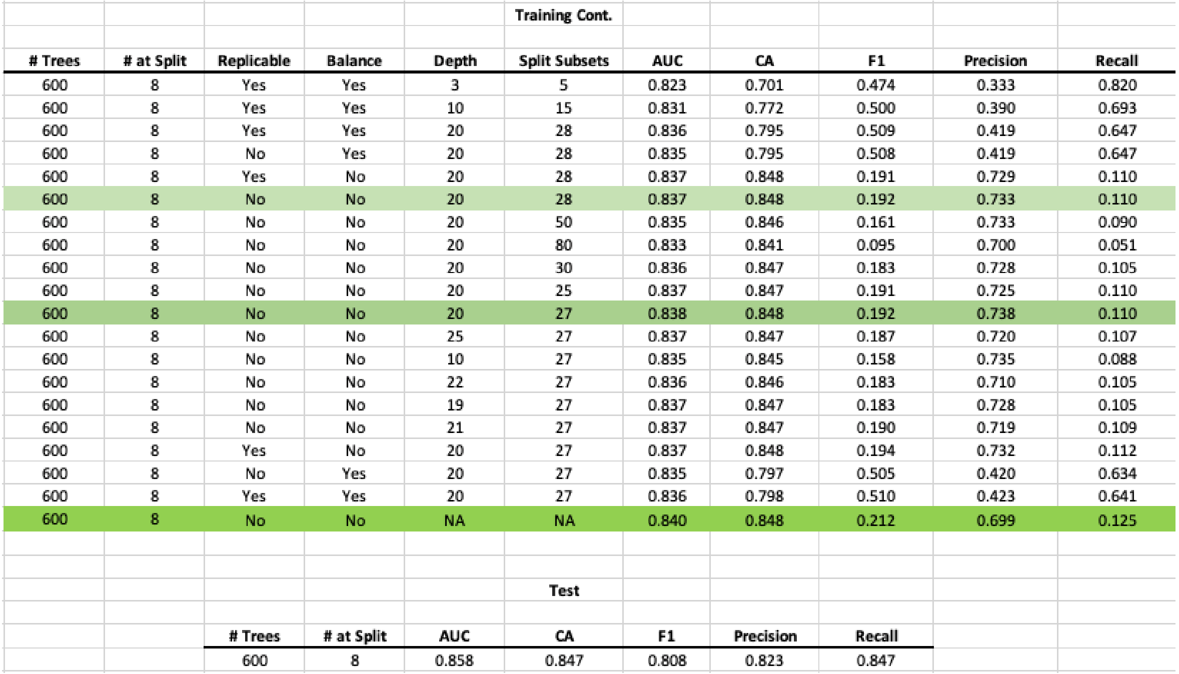
The last model we needed to test was the random forest. We built our first random forest testing 500 trees with our number of attributes considered at each split set to five. We kept the other filters off to begin, to find the optimal number of trees and splits before enhancing our search further. We found that the AUC was staying consistently similar, whether we were testing on 500 trees or 1500. The biggest obstacle we were facing was that the AUC values were fluctuating back and forth. It seemed that we had found a peak, but once we saw values dropping off on each side of this peak, we would see another jump further into testing. As you can see in the results below, AUC increases as it approaches a peak at 600 trees and eight attributes at split and decreases as it moves further away. We thought this would be our optimal peak but found that AUC increased once again at 700 trees with eight attributes at split. We kept the eight attributes considered at split found at both peaks, then tested them on the 750-1500 tree models. An AUC of 0.840 was found at multiple points, but our CA, F1, Precision, and Recall were all at their highest when the model used 600 trees and considered eight attributes at each split. We then decided to test on other parameters going forward with this 600/8 model.



			Train			
# Trees	**# at Split**	**AUC**	**CA**	**F1**	**Precision**	**Recall**
500	5	0.836	0.843	0.139	0.675	0.077
500	8	0.837	0.846	0.207	0.655	0.123
550	5	0.838	0.845	0.144	0.733	0.080
550	6	0.837	0.845	0.173	0.681	0.099
550	7	0.837	0.846	0.187	0.680	0.108
550	8	0.838	0.847	0.214	0.667	0.128
600	5	0.838	0.844	0.141	0.705	0.078
600	6	0.839	0.844	0.165	0.678	0.094
600	7	0.838	0.844	0.181	0.657	0.105
600	8	0.840	0.848	0.212	0.699	0.125
600	9	0.839	0.848	0.226	0.683	0.136
650	6	0.838	0.845	0.175	0.692	0.100
650	7	0.839	0.846	0.195	0.683	0.114
650	8	0.837	0.847	0.214	0.685	0.127
650	9	0.838	0.848	0.223	0.695	0.133
700	7	0.838	0.846	0.198	0.686	0.116
700	8	0.840	0.848	0.213	0.697	0.126
700	9	0.839	0.848	0.227	0.670	0.137
750	8	0.838	0.846	0.209	0.667	0.124
800	8	0.839	0.847	0.216	0.675	0.128
850	8	0.840	0.848	0.209	0.699	0.123
900	8	0.840	0.847	0.202	0.684	0.118
950	8	0.838	0.847	0.210	0.680	0.124
1000	8	0.840	0.846	0.201	0.667	0.118
1250	8	0.839	0.848	0.218	0.696	0.129
1500	8	0.839	0.846	0.204	0.673	0.120

Figure 39: This figure shows the initial training results on Random Forest in Orange, finding the optimal number of trees and attributes considered at each split.

The remaining parameters we needed to test on were the replicable trainings, balance of class distributions, limit on the depth of individual trees, and limit on our subset splits. We kept the 600 trees and eight attributes considered at the split for the remainder of our testing, as this was our optimal result found above. We decided to turn on the replicable and balance parameters to begin, with the default of depth limit and subset limits set to three and five, respectively. Our AUC dropped much lower, so we decided to increase the depth and split limits. Once we got up to an AUC of 0.836, we wanted to start switching our replicability and balance to see the effects it had. We found that turning both the replicability and balance off had positive effects on AUC, Precision, and Recall, so we kept those off for the remainder of the testing (lightest green). We continued to test our subset splits alone until we found the highest AUC, which we reached at 27 splits (medium green). We then tested different depths of our random forest model but were unable to find a depth that was greater than our previously found result of 0.838. Overall, we concluded that our best model was found (most green) at 600 trees, 8 attributes considered at each split, no replicable training, no balance in class distribution, and no growth control effects used (no limit on the depth of individual trees or splitting of subsets).



Training Cont.

# Trees	# at Split	Replicable	Balance	Depth	Split Subsets	AUC	CA	F1	Precision	Recall
600	8	Yes	Yes	3	5	0.823	0.701	0.474	0.333	0.820
600	8	Yes	Yes	10	15	0.831	0.772	0.500	0.390	0.693
600	8	Yes	Yes	20	28	0.836	0.795	0.509	0.419	0.647
600	8	No	Yes	20	28	0.835	0.795	0.508	0.419	0.647
600	8	Yes	No	20	28	0.837	0.848	0.191	0.729	0.110
600	8	No	No	20	28	0.837	0.848	0.192	0.733	0.110
600	8	No	No	20	50	0.835	0.846	0.161	0.733	0.090
600	8	No	No	20	80	0.833	0.841	0.095	0.700	0.051
600	8	No	No	20	30	0.836	0.847	0.183	0.728	0.105
600	8	No	No	20	25	0.837	0.847	0.191	0.725	0.110
600	8	No	No	20	27	0.838	0.848	0.192	0.738	0.110
600	8	No	No	25	27	0.837	0.847	0.187	0.720	0.107
600	8	No	No	10	27	0.835	0.845	0.158	0.735	0.088
600	8	No	No	22	27	0.836	0.846	0.183	0.710	0.105
600	8	No	No	19	27	0.837	0.847	0.183	0.728	0.105
600	8	No	No	21	27	0.837	0.847	0.190	0.719	0.109
600	8	Yes	No	20	27	0.837	0.848	0.194	0.732	0.112
600	8	No	Yes	20	27	0.835	0.797	0.505	0.420	0.634
600	8	Yes	Yes	20	27	0.836	0.798	0.510	0.423	0.641
600	8	No	No	NA	NA	0.840	0.848	0.212	0.699	0.125

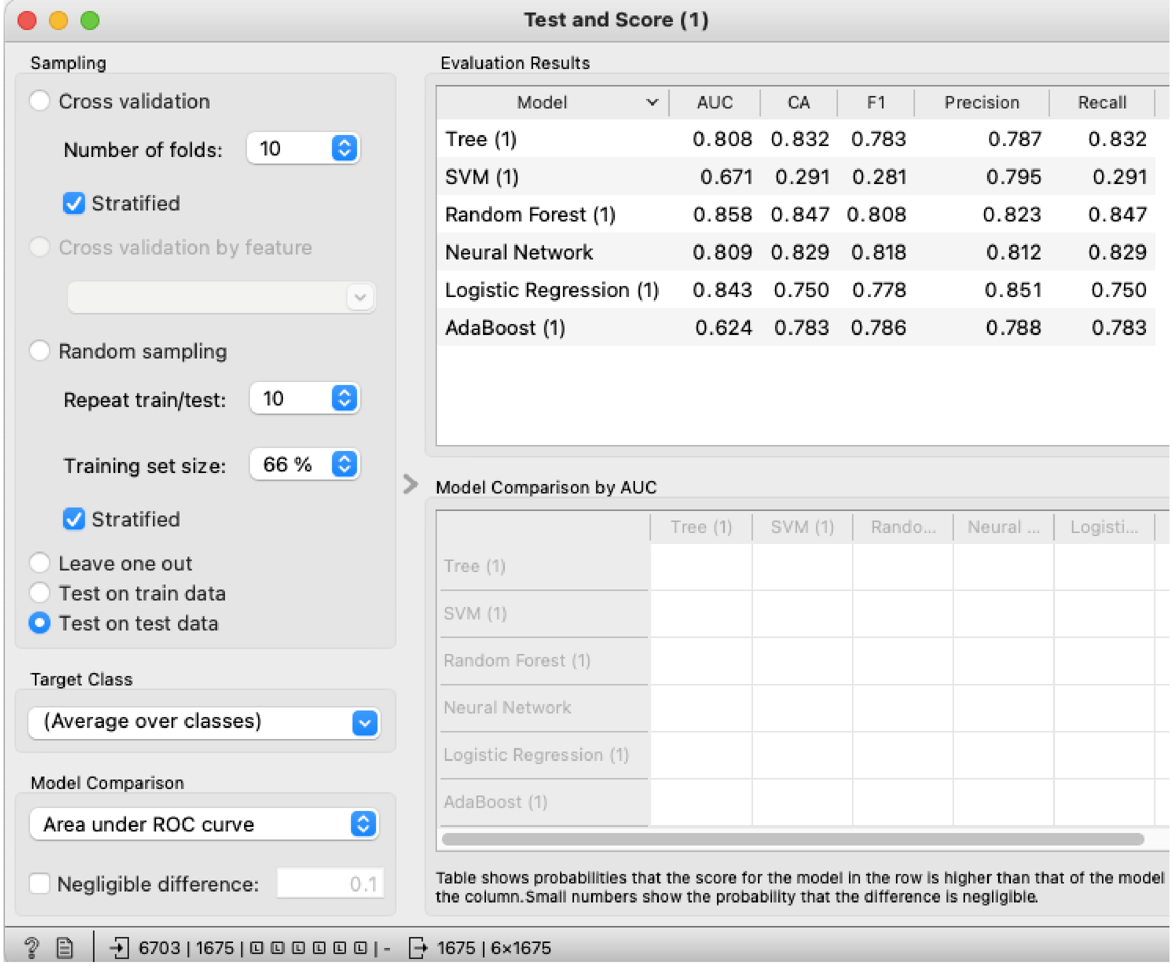
Test

# Trees	# at Split	AUC	CA	F1	Precision	Recall
600	8	0.858	0.847	0.808	0.823	0.847

Figure 40: This figure shows the second set of training results with Random Forest on Orange with all parameters being tested.

Comparison

Based on the AUC values of each of the 6 models, the random forest is the best predictor of which couples will match as its AUC value is highest (Figure 41). The random forest also has the best ROC curve as shown in Figure 42.



Test and Score (1)

Sampling

○ Cross validation

Number of folds: 10

☑ Stratified

○ Cross validation by feature

○ Random sampling

Repeat train/test: 10

Training set size: 66 %

☑ Stratified

○ Leave one out

○ Test on train data

● Test on test data

Target Class

(Average over classes)

Model Comparison

Area under ROC curve

□ Negligible difference: 0.1

Evaluation Results

Model	AUC	CA	F1	Precision	Recall
Tree (1)	0.808	0.832	0.783	0.787	0.832
SVM (1)	0.671	0.291	0.281	0.795	0.291
Random Forest (1)	0.858	0.847	0.808	0.823	0.847
Neural Network	0.809	0.829	0.818	0.812	0.829
Logistic Regression (1)	0.843	0.750	0.778	0.851	0.750
AdaBoost (1)	0.624	0.783	0.786	0.788	0.783

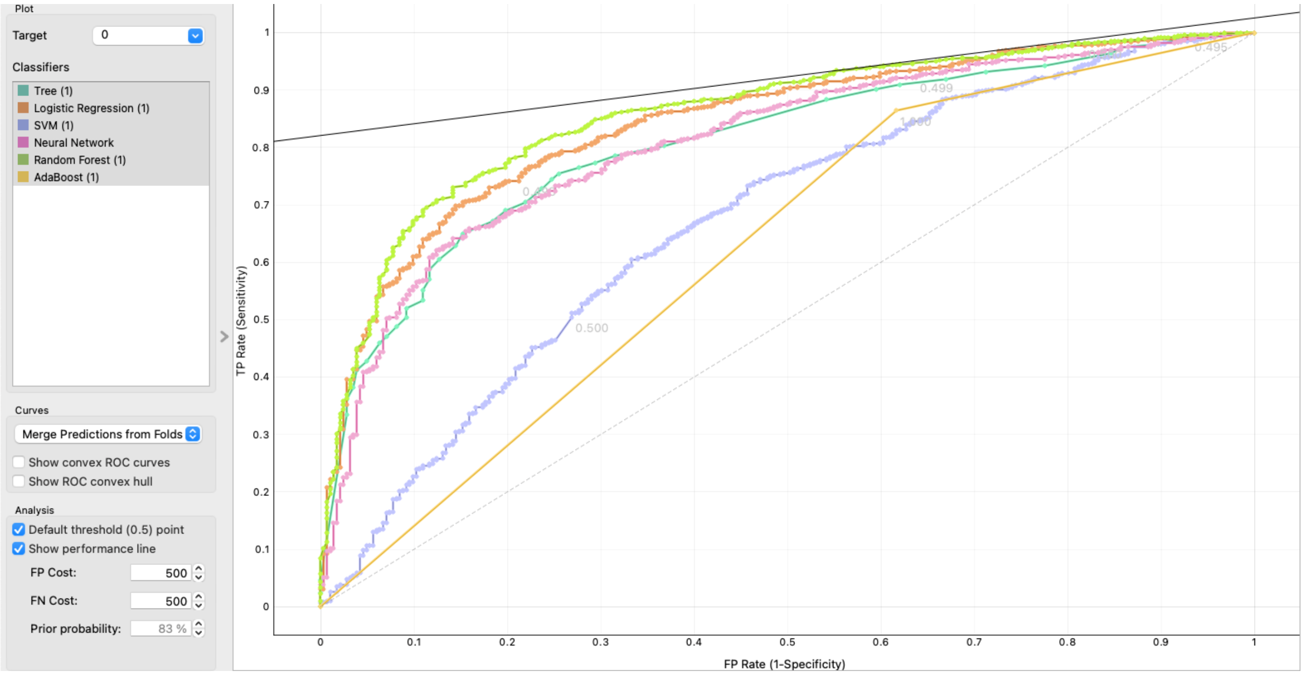
Model Comparison by AUC

	Tree (1)	SVM (1)	Rando...	Neural ...	Logisti...
Tree (1)					
SVM (1)					
Random Forest (1)					
Neural Network					
Logistic Regression (1)					
AdaBoost (1)					

Table shows probabilities that the score for the model in the row is higher than that of the model the column. Small numbers show the probability that the difference is negligible.

6703 | 1675 | □ □ □ □ □ □ | - 1675 | 6×1675

Figure 41: Final scores of all models.


Plot

Target: 0

Classifiers
- Tree (1)
- Logistic Regression (1)
- SVM (1)
- Neural Network
- Random Forest (1)
- AdaBoost (1)

Curves

Merge Predictions from Folds

□ Show convex ROC curves

□ Show ROC convex hull

Analysis

☑ Default threshold (0.5) point

☑ Show performance line

FP Cost: 500

FN Cost: 500

Prior probability: 83 %

Figure 42: ROC curves for all models.

Test Set Performance (of Random Forest)

After running the final six models through our testing set, we were able to find our completed test and score results. As stated in the disclaimer, some of our results were a bit different from when we tested them on our individual Orange platforms. The random data sampler chose different splits for each model, so when we came together to put our final models into Orange, some of our results in the test and score varied scarcely. Our models remained in the same order in terms of AUC, so we decided that our results were as accurate as could be. The two images above show that the random forest outperformed the other models in terms of AUC. Our random forest outperformed the other models with logistic regression following close behind. The neural network and decision tree performed well with very mere margins splitting the two. Our two lowest models were SVM and adaptive boosting with a large margin separating them from the other four models in terms of AUC. We noticed that our logistic regression and random forest models improved in our testing set as well, both increasing AUC values by 0.1 or above. Overall, we were happy with the performance of our final random forest and chose this as our best model to represent this data set.

Interpretation (of Random Forest)

Since Orange lacks the ability to see how the random forest it generates weights the different features in terms of importance, a random forest with the same parameters as the highest AUC random forest (600 trees, eight features at split) was created in Python to simulate the random forest created in Orange. The importance of each of the features in the Python random forest was then examined to approximate the most important features in the random forest created by Orange. According to the Python random forest, the most important features in determining whether or not a couple will match are: if the person rates their partner 5 or less out of 10 on attractiveness, if the like score given by the other person is 5 or less out of 10, if the person rates their partner 5 or less out of 10 on humor, if their partner rates them 5 or less out of 10 on attractiveness, and if the person guessed that the other person liked them (Figure 43). This means that attractiveness, humor, and feelings of likeness toward the other person are the most important factors according to the random forest. If either person's like score is low, it has a large, presumably negative, impact on their ability to match. Likewise, if someone's humor score is low, it also has a large, presumably negative, impact on their ability to match. If someone does not like their partner, they are unlikely to want to match with them. Conversely, if someone feels their partner likes them, they are more likely to want to match with them. Approximated by the Python random forest, the least important factor in determining whether or not two people match are whether the person and their partner feel that being ambitious is important, which means feelings about ambition do not matter much when predicting matches (Figure 43).

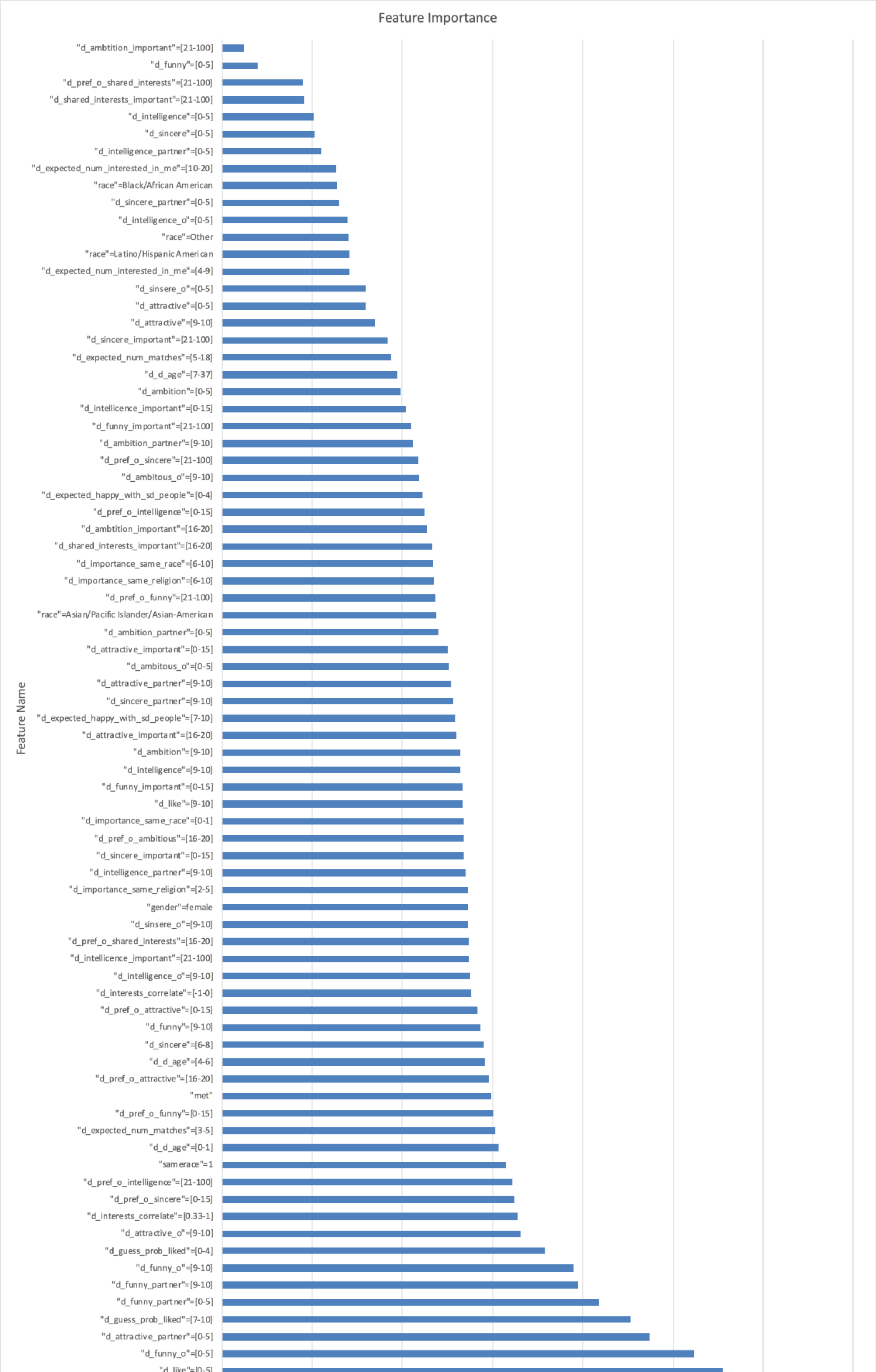

Feature Importance
Feature Name
"d_ambtition_important"=[21-100]
"d_funny"=[0-5]
"d_pref_o_shared_interests"=[21-100]
"d_shared_interests_important"=[21-100]
"d_intelligence"=[0-5]
"d_sincere"=[0-5]
"d_intelligence_partner"=[0-5]
"d_expected_num_interested_in_me"=[10-20]
"race"=Black/African American
"d_sincere_partner"=[0-5]
"d_intelligence_o"=[0-5]
"race"=Other
"race"=Latino/Hispanic American
"d_expected_num_interested_in_me"=[4-9]
"d_sinsere_o"=[0-5]
"d_attractive"=[0-5]
"d_attractive"=[9-10]
"d_sincere_important"=[21-100]
"d_expected_num_matches"=[5-18]
"d_d_age"=[7-37]
"d_ambition"=[0-5]
"d_intellicence_important"=[0-15]
"d_funny_important"=[21-100]
"d_ambition_partner"=[9-10]
"d_pref_o_sincere"=[21-100]
"d_ambitous_o"=[9-10]
"d_expected_happy_with_sd_people"=[0-4]
"d_pref_o_intelligence"=[0-15]
"d_ambtition_important"=[16-20]
"d_shared_interests_important"=[16-20]
"d_importance_same_race"=[6-10]
"d_importance_same_religion"=[6-10]
"d_pref_o_funny"=[21-100]
"race"=Asian/Pacific Islander/Asian-American
"d_ambition_partner"=[0-5]
"d_attractive_important"=[0-15]
"d_ambitous_o"=[0-5]
"d_attractive_partner"=[9-10]
"d_sincere_partner"=[9-10]
"d_expected_happy_with_sd_people"=[7-10]
"d_attractive_important"=[16-20]
"d_ambition"=[9-10]
"d_intelligence"=[9-10]
"d_funny_important"=[0-15]
"d_like"=[9-10]
"d_importance_same_race"=[0-1]
"d_pref_o_ambitious"=[16-20]
"d_sincere_important"=[0-15]
"d_intelligence_partner"=[9-10]
"d_importance_same_religion"=[2-5]
"gender"=female
"d_sinsere_o"=[9-10]
"d_pref_o_shared_interests"=[16-20]
"d_intellicence_important"=[21-100]
"d_intelligence_o"=[9-10]
"d_interests_correlate"=[-1-0]
"d_pref_o_attractive"=[0-15]
"d_funny"=[9-10]
"d_sincere"=[6-8]
"d_d_age"=[4-6]
"d_pref_o_attractive"=[16-20]
"met"
"d_pref_o_funny"=[0-15]
"d_expected_num_matches"=[3-5]
"d_d_age"=[0-1]
"samerace"=1
"d_pref_o_intelligence"=[21-100]
"d_pref_o_sincere"=[0-15]
"d_interests_correlate"=[0.33-1]
"d_attractive_o"=[9-10]
"d_guess_prob_liked"=[0-4]
"d_funny_o"=[9-10]
"d_funny_partner"=[9-10]
"d_funny_partner"=[0-5]
"d_guess_prob_liked"=[7-10]
"d_attractive_partner"=[0-5]
"d_funny_o"=[0-5]
"d_like"=[0-5]

Figure 43: Bar graph showing the importance of each of the features in the data set as determined by the random forest generated in Python.

Cost Structure

We chose to assume the position of a restaurant or bar that would be hosting the speed dating event to construct the profit curve of our final model. Many speed dating events are held in restaurants and bars because they are an environment where daters can eat, drink, and socialize. Our restaurant has an interest in whether the daters will match and meet again in the future because we will encourage the couple to come back to our establishment by providing a gift card. Couples tend to have loyalty to the place they first met, so we would expect these couples to become lifelong patrons of our establishment.

We decided to provide a $10 gift card to each dater that we expected to leave with a match at the end of the event. We settled on this dollar amount allocation after building several simulations of profit curves given our data from our random forest model. On average, American couples spend $100 total ($50 per person) for a meal at a nice restaurant. Therefore, if we predicted a match correctly, we expect the dater to return to the restaurant and earn us a profit of $40, calculated by taking the total meal price of $50 minus the total gift card value of $10. However, if we predicted a match incorrectly, we record a cost of $10. Because there is one gift card given per person, we anticipate each of them to return to the restaurant alone. It is difficult to estimate how much that individual will spend, so we are being conservative in our estimates and predict they will only spend the amount of the $10 gift card on a drink or appetizer. This expectation will result in a $10 loss for our establishment because we gave away a free $10 gift card and they did not spend more than that amount at our restaurant upon returning.

Figure 46 below pinpoints the maximum value achieved on our profit curve. The maximum profit achieved is $6,060, which occurs at the 509th instance in our data. This maximum profit means that we will provide gift cards for 509 of the total 1,675 daters at the event. We provide the gift card to anyone that we are 22.83% sure will leave the event with a match. Figure 45 gives a visual representation of our restaurant's profit at various instances. We predict a dater's likelihood of a match, given the importance of the features shown in the random forest results, in Figure 43. Some of these factors are observable, such as attractiveness, and self-reported data from surveys done before a dater attends the event would give us the ability to predict how likely we think someone is to leave with a match.

The lift curve in Figure 44 means that providing the gift cards to the 509 daters we predict to match will allow us to earn approximately 2.8 times more profit than if we randomly selected daters to give the gift cards to. This is more evidence that our model is a strong predictor of compatibility between daters based on our model's features of importance. It is important to notice that the curve does drop down to 1, meaning our model would be no better than randomly guessing, but this only occurs if gift cards are given to all 1,675 of the daters at the event.

Profit Curve Images

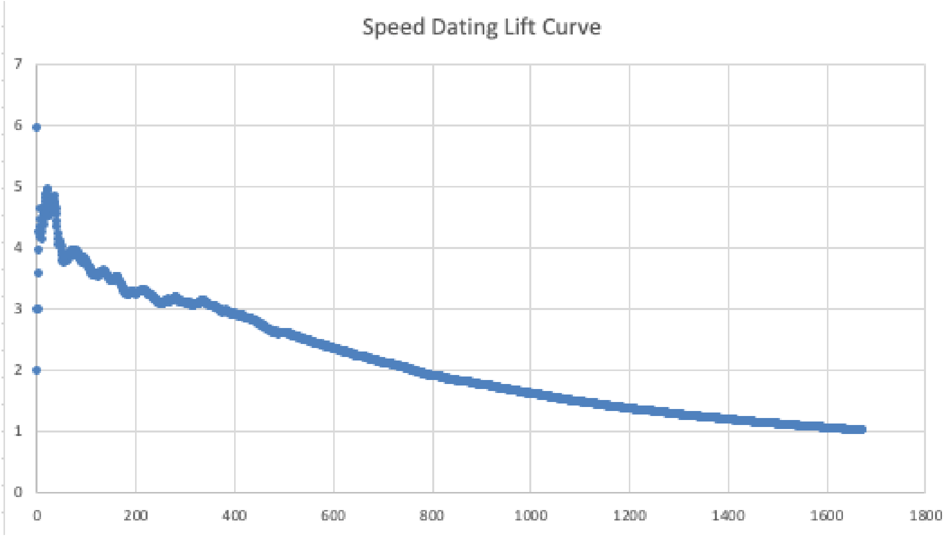


Figure 44: This figure represents the Lift Curve produced by our final Random Forest Model in Excel

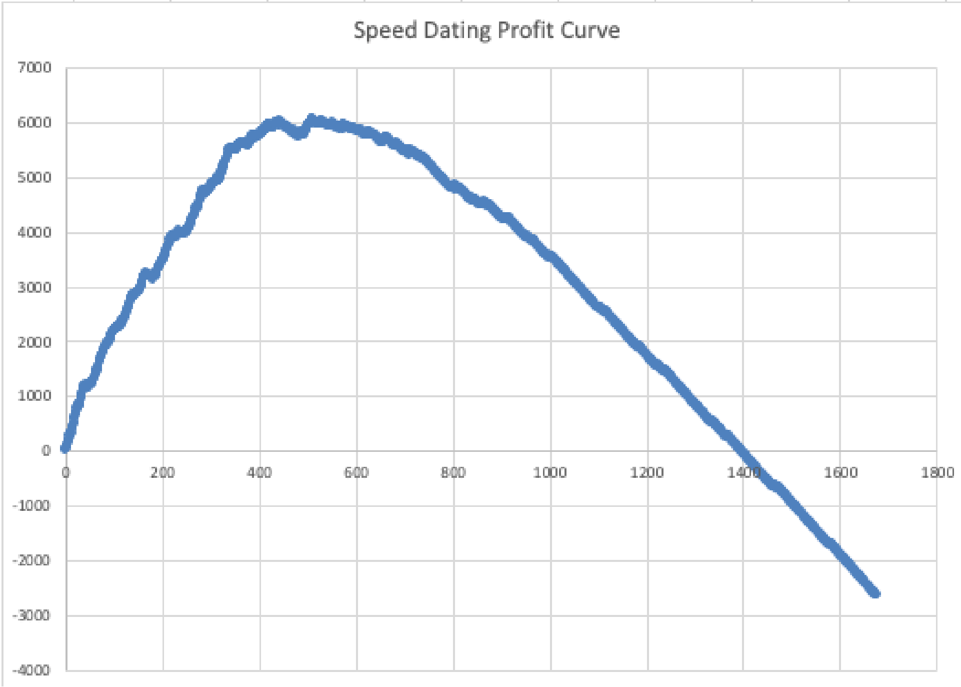
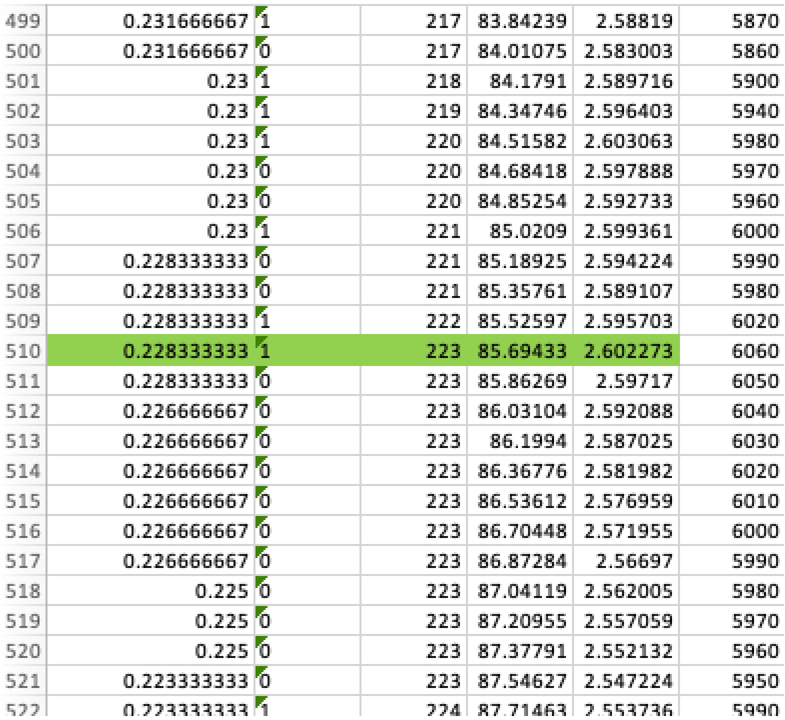


Figure 45: This figure represents the Profit Curve produced by our final Random Forest Model in Excel



499	0.231666667	1		217	83.84239	2.58819	5870
500	0.231666667	0		217	84.01075	2.583003	5860
501	0.23	1		218	84.1791	2.589716	5900
502	0.23	1		219	84.34746	2.596403	5940
503	0.23	1		220	84.51582	2.603063	5980
504	0.23	0		220	84.68418	2.597888	5970
505	0.23	0		220	84.85254	2.592733	5960
506	0.23	1		221	85.0209	2.599361	6000
507	0.228333333	0		221	85.18925	2.594224	5990
508	0.228333333	0		221	85.35761	2.589107	5980
509	0.228333333	1		222	85.52597	2.595703	6020
510	0.228333333	1		223	85.69433	2.602273	6060
511	0.228333333	0		223	85.86269	2.59717	6050
512	0.226666667	0		223	86.03104	2.592088	6040
513	0.226666667	0		223	86.1994	2.587025	6030
514	0.226666667	0		223	86.36776	2.581982	6020
515	0.226666667	0		223	86.53612	2.576959	6010
516	0.226666667	0		223	86.70448	2.571955	6000
517	0.226666667	0		223	86.87284	2.56697	5990
518	0.225	0		223	87.04119	2.562005	5980
519	0.225	0		223	87.20955	2.557059	5970
520	0.225	0		223	87.37791	2.552132	5960
521	0.223333333	0		223	87.54627	2.547224	5950
522	0.223333333	1		224	87.71463	2.553736	5990

Figure 46: This figure represents the maximum value in our profit curve, where we plan to set our limit for gift card allocation.

Recommendations

By using this random forest model, we have found a way to maximize the restaurant's profits with a gift card as an incentive. To further increase these profits, we recommend that the restaurant provides a preliminary survey to all its participants. We will construct this survey using only variables that scored above 0.015 on the importance index of the random forest model. Using this threshold, daters will be more likely to fill out the condensed survey and our establishment will not be overwhelmed with too many features for each dater. In its current state, the restaurant is pairing individuals randomly. With the condensed preliminary survey, these pairings could be made based on participants' similar interests and desires. We also recommend extending the length of the speed dates. We believe that longer dates between participants with similar interests will increase the total number of matches. Those who match will then receive their gift card and return to the restaurant for another date, increasing the profits of the restaurant.

We believe that these simple changes could make speed dating events and the restaurant’s primary business practices more successful.

Works Cited

Raymond Fisman; Sheena S. Iyengar; Emir Kamenica; Itamar Simonson. Gender Differences in Mate Selection: Evidence From a Speed Dating Experiment. The Quarterly Journal of Economics, Volume 121, Issue 2, 1 May 2006, Pages 673–697, https://doi.org/10.1162/qjec.2006.121.2.673 obtained from OpenML