# Determining Attractive Traits for Men and Women Through Predictive Modeling

Neva Lui and Tu Do

January 6, 2019

**Abstract**

In this experiment, we investigate the most desirable traits in men and women that make them more "attractive". Supervised machine learning algorithms (decision trees and random forests) will be used to approach this problem. The performance of these algorithms will be investigated along with each model's advantages and disadvantages. Learning models, as well as their accuracy, will be shown as a part of this experiment. Based on the characteristics and performance of these models, the most desirable traits in men and women are able to be determined.

## 1 Introduction

### 1.1 Overview

Depending on the occupation, different groups of people can analyze data in various ways. For example, psychologists may interpret data and make predictions based on their previously acquired knowledge about human behavior and cognitive tendencies. Similarly, Artificial intelligence scientists are concerned with analyzing data and leveraging this knowledge to predict future trends, but they utilize raw data that was previously collected on a subject.

In this experiment, we use the speed dating experiment data, gathered by Ray Fisman and Sheena Iyengar at Columbia University, to explore and compare the traits men and women find most desirable in a date and potentially in a partner (Fisman et al., 2006). The traits that we will mostly focus on are (physical) attractiveness, sincerity, intelligence, fun, ambition, and shared interests. By identifying the most important and least important traits, an individual could potentially increase his/her "attractiveness" in the dating market by applying the trait analysis in his/her self-improvement. More importantly, this analysis could be a significant contribution to social science studies in predicting human behaviors and values.

### 1.2 Dataset

The speed dating data was collected from multiple speed dating events between 2002-2004. The data set was composed of data from a pre-survey, score card evaluation (during event), and two post-event surveys. At each event, participants were

involved in four minute dates with other participants. Participants were asked to evaluate each partner's level of (physical) attractiveness, sincerity, intelligence, fun, ambition, and shared interests. The scale was from 1-10, where 1 is not important and 10 is extremely important. Each participant was also asked if he/she would be interested in seeing his/her partner again after the event.

For the purpose of this experiment, we assume that a person's "attractiveness" is higher in the dating market if his/her partner is interested in seeing him/her again. This assumption would enable us to use machine learning algorithms to determine which traits are most desirable by the opposite sex. Based on the results, individuals can identify traits they should enhance and suppress in order to increase their "attractiveness" in the dating market.

## 1.3 Algorithms

The problem above can be solved by applying Machine learning algorithms. Machine learning is a subfield of Artificial Intelligence that uses statistical methods to give a computer the ability to "learn" on its own without explicitly programming the patterns in the data. Explicitly telling the computer the patterns becomes increasingly difficult as the data set becomes larger. Therefore, machine learning algorithms help simplify big data analysis.

The machine learning algorithms that will be used in this experiment are supervised learning algorithms. Supervised learning consists of trying to predict the outcome of a target variable given a set of predictor variables (Ray, 2017). Supervised learning algorithms include, but are not limited to, regressions, decision trees, and KNN (k-nearest neighbors). These algorithms can be used to classify whether an individual is more "attractive" (target variable) based on a specific trait or a combination of traits (predictor variables). In this experiment, decision trees and random forests will be used to make this classification (details in section 3).

# 2 Literature Review

Machine learning algorithms are frequently used to predict patterns when analyzing data sets. Prediction tasks are crucial and prevalent in a myriad of industries and fields such as biology, economics, and astronomy. To keep up with the growing demand for prediction tasks, many learning models have been developed. As mentioned in the previous section, Decision Trees and Random Forests will be the algorithms investigated in this experiment. Both algorithms are based on the concept of tree classification. While Decision Trees are standalone trees, Random Forests are compositions of multiple Decision Trees or other tree classifiers.

## 2.1 Decision Trees

### 2.1.1 Overview

In classification problems, the dataset being analyzed includes several features (predictor variables) and a class variable (target variable). Given a set of features,

the goal is to build a model on a training sample that can accurately predict the class variable. Decision Trees are one among many classification models that are used to predict a class variable. Conceptually, decision trees recursively partition datasets based on predictor variables, one variable at a time (Loh, 2011). Details of building a decision tree are provided in section 4.1.2. There are numerous variations of decision trees (e.g. pruning); however, only classical decision trees will be discussed further (Patel and Upadhyay, 2012).

### 2.1.2 Building Tree Model For Classification

Suppose we want to predict whether a person will go out for a run on a typical day. Some deciding factors may include weather, temperature, or humidity. These factors can be used to construct a set of predictions. A tree representation of this example is depicted in Figure 1. The leaf nodes of the tree represent the classification of the target variable based on the values of the predictor variables. The other nodes (internal nodes) are the predictor variables. Each internal node has two or more branches; the branches correspond to the values the predictor variables could have.

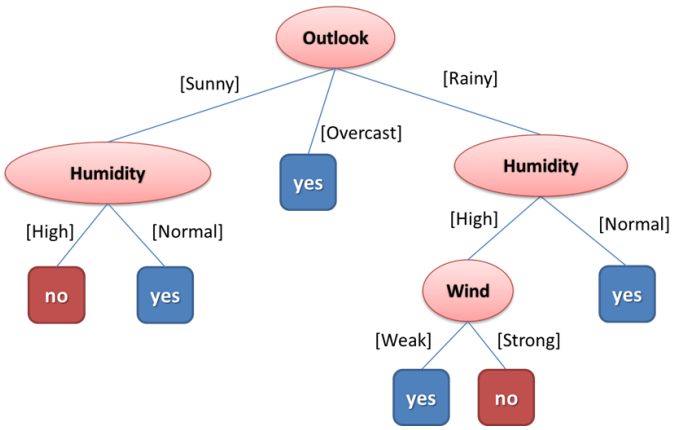



Figure 1: Example of a decision tree (Souza, 2012)

In real life, most large data sets include redundant or irrelevant features. To extract useful features, impurity measures (Gini Index in CART, entropy in ID3, GID3*, or C-SEP) are used to calculate the most important feature in the data set, which will be extracted from the remaining, less important features (Fayyad and Irani, 1992); the first extracted feature becomes the root of the tree. The algorithm iterates through the remaining features and extracts the next useful feature until there are no useful features remaining.

When using the decision tree model, tree complexity should be taken into consideration because it affects the model's ability to make accurate predictions and find patterns. In general, the size of the trees should be small, so the results are easier to understand (Bhargava et al., 2013). If the tree is too complex, the model tends to be over-fitted to the training set; the models have high accuracy in predicting training samples and low accuracy in predicting test samples. On the other hand, if the tree is too simple, the model could also suffer from under-fitting: the trees cannot capture patterns in the data.

## 2.2 Random Forests

### 2.2.1 Overview

Since decision trees have a tendency to be overfitted to the data, possible alternatives should be considered for classification problems. Random forest is a model that is derived from the decision tree model, but it tends to reduce overfitting the training set to the model. The algorithm utilizes multiple classification (decision) trees, adds a component of "randomness" by randomly selecting which predictor variables will be used in each classifier (different for each tree), and determine feature importance based on the decision trees included in the majority vote. The number of features chosen should optimize the strength of the individual trees and the correlation between trees. Correlation has a direct relation with error rate while strength has an inverse relationship with error rate (Breiman, 2001). Both of these properties increase as the number of features selected to build the model increases, so the number of features chosen should try to minimize correlation, but also maximize strength.

### 2.2.2 Random Forest in Classification

The random forest model is based off the method of bagging which will be discussed in Section 2.2.3. As mentioned in the overview, predictor variables are randomly chosen when building each classifier in the forest. This randomness helps decorrelate the trees and potentially reduce variance.

### 2.2.3 Bagging

Bagging (bootstrap aggregating) is an ensemble method used to help increase the accuracy of prediction when random features are fitted to a model by reducing the error in variance (Breiman, 1996). The process of bagging is outlined below:

1. Generate random samples from the training data set
2. Build a classifier for each sample
3. Average the results (regression) or take the majority vote (classification) from these classifiers to create an aggregated predictor for the final model

Bagging is useful when predictors are unstable, but if a predictor is stable to begin with, bagging can transform the classifier into a worse classifier (Breiman, 1996). Based on Breiman's instability study (results in Figure 2), classification was determined to be an unstable procedure; therefore, bagging can be an effective method of producing more optimal results (Breiman, 1996).

| Data Set | $\bar{e}_S$ | $\bar{e}_B$ | Decrease |
|---|---|---|---|
| waveform | 29.1 | 19.3 | 34% |
| heart | 4.9 | 2.8 | 43% |
| breast cancer | 5.9 | 3.7 | 37% |
| ionosphere | 11.2 | 7.9 | 29% |
| diabetes | 25.3 | 23.9 | 6% |
| glass | 30.4 | 23.6 | 22% |
| soybean | 8.6 | 6.8 | 21% |

Figure 2: Misclassification rate comparison ($\bar{e}_B$ = average bagging misclass. rate, $\bar{e}_S$ = average misclass. rate without bagging) (Breiman, 1996)

#### 2.2.4 Individual Tree Construction

Suppose that a data set is composed of 15 predictor variables, and 10 of those 15 variables are randomly chosen (after each node split) to build individual trees in the forest. The goal is to train each tree with different predictor variables. Each of the 10 variables will be assigned a probability where the sum of all the probabilities is 1. Each node will correspond to one of the 10 variables, and the node will be split based on the probability assigned to that variable. If the variable's probability is high, the variable/node will be split based on the median of its values. If the probability is low, the value the variable/node will be split at is randomly chosen (Breiman, 2004).

# 3 Problem Approach

As described in the previous section (Section 2), decision trees and random forests are some of the most frequently used algorithms with classification problems. In order to understand why these two algorithms were chosen for this experiment, the advantages and disadvantages of each algorithm were explored.

## 3.1 Decision Tree Advantages and Disadvantages

### 3.1.1 Advantages

In data mining, decision trees are the most powerful method for knowledge discovery (Bhargava et al., 2013). Decision trees have the ability to learn from datasets of large sizes and extract the useful features to build decision trees. The model also provides insight about the rules used when creating the tree. Each path, starting from the root to a leaf of a tree, generates a rule. Other advantages of decision trees include the ability to handle various types of data such as nominal, numeric, and textual datasets (Bhargava et al., 2013). Decision trees are also able to perform will with small amounts of noise and outliers in the data. Moreover, decision trees are known to be an efficient algorithm for data mining. Libraries have been developed for building decision tree models where users can tune tree parameters in various platforms (Bhargava et al., 2013).

### 3.1.2 Disadvantages

Despite its advantages, decision trees also have many disadvantages. When the structure of the data set is too complicated, decision tree models have difficulty in classifying the target variable. The tree models also tend to be over-fitted. These problems can be addressed by using other classifiers such as Random Forests (discussed in section 2.2) or Gradient Boosted Trees. Furthermore, decision trees are not applicable in classifying data with a large number of irrelevant attributes. Attributes are considered irrelevant if they are either weakly associated with the target variable

or highly correlated with another attribute (Tan et al., 2005). Highly correlated variables are redundant, which could result in poorer classification performance; using related attributes creates extra noise in model training.

## 3.2 Random Forest Advantages and Disadvantages

### 3.2.1 Advantages

The advantages of using a random forest directly parallel the advantages of using decision trees: performs well (prediction accuracy) with large amount of variables (large dataset), identifies useful and important features, and handles small amounts of missing data (Breiman, 2004). Some additional advantages random forests provide are a smaller likelihood of overfitting the data and the ability to discern more complicated patterns within the data set.

### 3.2.2 Disadvantages

Unfortunately, the algorithm is not the easiest to analyze. Unlike the decision tree, the rules used in selecting the optimal number and combination of features are not visible to the user. Since random forests aggregate the parameters from the trees in the majority vote, the rules of each individual tree used to build the random forest are not available.

In addition to potential roadblocks with analyzing rules for each decision tree, more time and memory may be required to build the model. Time and memory used by the algorithm is heavily dependent on the number of trees and the size of the trees in the forest. A large number of trees/larger trees will typically increases prediction accuracy, but requires more time and memory. Conversely, smaller number of trees/smaller trees will reduce the amount of time and memory required to build the forest, but the prediction accuracy may be hindered from not having enough trees/features to find the "correct" patterns in the data (inaccurate majority vote) (Cinaroglu, 2016).

## 3.3 Algorithm Comparison

When building classification models, it is essential to consider the prediction accuracy (train and test) and runtime of the model. Since both algorithms are tree classifiers, some of their algorithm properties are similar. When the properties differ, one algorithm's strength is the other algorithm's weakness. Fitting the data to both models will enable us to make a direct comparison of each algorithm's performance with classifying the data. Prediction accuracy will be the main measure for evaluating the algorithm's performance. More details about the design of the experiment and algorithm evaluation is described below (Section 4).

# 4 Experiment

## 4.1 Design

### 4.1.1 Data Preprocessing

Data preprocessing will be composed of both manual and automatic processing of the dataset. Attributes that are not human traits such as income, career or religious belief will be removed (manually) from the data set because they go beyond the scope of the experiment. The data set also includes survey questions at three different intervals for each event (mentioned in section 1.2), but the only features that were pertinent to the scope of the experiment are the initial importance of each trait (pre-event survey) and the evaluation of the traits during the event (score card). Outliers, missing values, and duplicate data should be handled to improve data quality. Since the data set is quite large, it is better to eliminate these types of data entries and attributes because extra noise in the data can inhibit the performance (speed and accuracy) of our algorithms.

After the model is fitted to the preprocessed data (building the model: section 4.1.2), dimensions could be further reduced to remove attributes that are highly correlated to each other (covariance). If dimensions are reduced based on a positive or negative relationship between attributes, the combination of attributes will be considered when analyzing the final results of the model prediction. Specifically for tree classifiers (decision trees, random forests, etc), feature importances can be used as another method of dimension reduction because less important attributes (extra noise) can be removed, and more "accurate" predictions can be produced. Since the focus of the experiment is to compare the desired traits for males and females, the dataset was divided into two sub-datasets: one for male data and one for female data. The preprocessing steps mentioned above will be performed for each of the sub-datasets.

### 4.1.2 Building the Learning Model

Before fitting the sub-datasets to each of the models/algorithms (decision tree and random forest), each sub-dataset will be divided into a training set and a test set. The training set is used to help the algorithm learn information about the existing domain while the test set is used to evaluate the performance of the trained model (Tan et al., 2005). Choosing the split ratio between the training set and the test set is important because a smaller training set (larger test set) typically produces higher variance in the model results while a larger training set (smaller test set) usually leads to model evaluation with higher inaccuracy. Several versions of each model/algorithm will be compared based on varying training set-test set ratios (%) of the original data set: 90-10, 80-20, and 70-30.

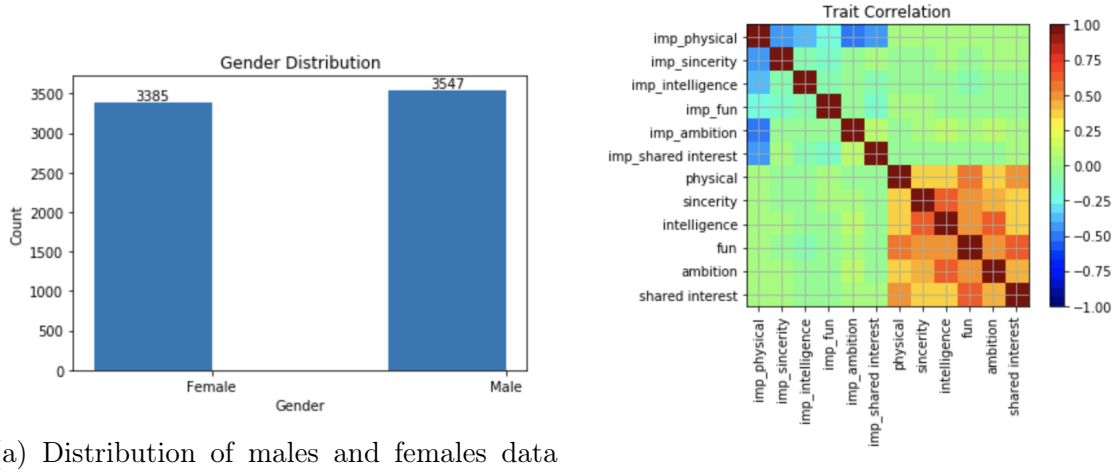
## 4.2 Solution Evaluation

The models will be evaluated based on each model's accuracy, runtime, and distribution of classification (true positives, true negatives, false positives, false negatives). A confusion matrix will be used to visualize the distribution of predicted

classifications from evaluating the test set. Accuracy, which is derived from the confusion matrix, will be measured by determining what percent of the test data predictions are correctly classified using the trained model. Sometimes the results may produce nearly 100% accuracy, which indicates that the model is likely over-fitted. To help reduce the over-fitting problem, dimension reduction as well as a comparison of the performance of the different versions of the same model (training-test ratios) will be used. In addition to the metrics described above, the run times of each algorithm for training the model and evaluating the test set will be compared. There is a possibility that there will be a trade off between run time of the algorithms and accuracy. However, for the purpose of this experiment, the algorithm that produce the best accuracy will be deemed the best algorithm for classifying the dataset.

## 4.3 Results

Based on Figure 3a, there is a fairly similar distribution of males and females that participated in the speed dating experiment. Figure 3b expresses that the twelve features that remain in the dataset are not heavily correlated.




(a) Distribution of males and females data points

(b) Correlation between traits

Figure 3: Characteristics of the speed dating dataset

### 4.3.1 Train-Test Split

The random forest classifier was used to compare the prediction accuracies for males and females based on the train-test split.

| 70-30 | Male | | Female | |
|---|---|---|---|---|
| Trial | Train | Test | Train | Test |
| 1 | 0.99 | 0.81 | 0.99 | 0.76 |
| 2 | 0.99 | 0.78 | 0.99 | 0.77 |
| 3 | 0.99 | 0.78 | 0.99 | 0.78 |
| 4 | 0.99 | 0.80 | 0.99 | 0.77 |
| 5 | 0.99 | 0.79 | 0.98 | 0.77 |

Table 1: Train-Test model prediction accuracies for 70-30

| 80-20 | Male | | Female | |
|---|---|---|---|---|
| Trial | Train | Test | Train | Test |
| 1 | 0.99 | 0.77 | 0.98 | 0.78 |
| 2 | 0.99 | 0.80 | 0.98 | 0.76 |
| 3 | 0.99 | 0.80 | 0.99 | 0.77 |
| 4 | 0.99 | 0.78 | 0.99 | 0.76 |
| 5 | 0.99 | 0.79 | 0.98 | 0.78 |

Table 2: Train-Test model prediction accuracies for 80-20

| 90-10 | Male | | Female | |
|---|---|---|---|---|
| Trial | Train | Test | Train | Test |
| 1 | 0.99 | 0.79 | 0.99 | 0.79 |
| 2 | 0.99 | 0.82 | 0.99 | 0.80 |
| 3 | 0.99 | 0.78 | 0.98 | 0.79 |
| 4 | 0.99 | 0.79 | 0.98 | 0.79 |
| 5 | 0.99 | 0.80 | 0.99 | 0.76 |

Table 3: Train-Test model prediction accuracies for 90-10

When comparing the train and test accuracies for each train-test split, the results were similar across splits for a particular gender. The results between genders were fairly similar, but the test accuracy for males was slightly higher than the test accuracy for females.

#### 4.3.2 Model Performance Comparison

Overall, the train-test split accuracies were essentially the same, so the 80-20 train set to test set split was arbitrarily chosen between the other splits for the model evaluation. Table 4 and 5 show the performance of each algorithm with respect to prediction accuracy (train and test) and runtime (time to build the model) for each gender dataset. Table 6 compares the overall performance between the two algorithms.

| | Male | | | Female | | |
|---|---|---|---|---|---|---|
| Trial | Train | Test | Time (s) | Train | Test | Time (s) |
| 1 | 1.00 | 0.75 | 0.01593 | 1.00 | 0.70 | 0.02096 |
| 2 | 1.00 | 0.75 | 0.01427 | 1.00 | 0.70 | 0.02114 |
| 3 | 1.00 | 0.76 | 0.02062 | 1.00 | 0.69 | 0.02090 |
| 4 | 1.00 | 0.74 | 0.01354 | 1.00 | 0.73 | 0.02018 |
| 5 | 1.00 | 0.73 | 0.01381 | 1.00 | 0.71 | 0.02041 |

Table 4: Decision Tree Model Performance

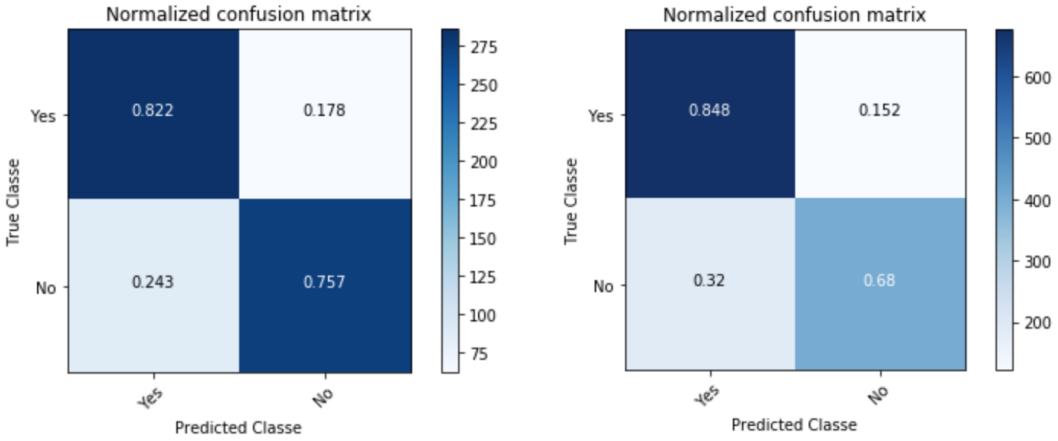
| | Male | | | Female | | |
|---|---|---|---|---|---|---|
| Trial | Train | Test | Time (s) | Train | Test | Time (s) |
| 1 | 0.99 | 0.80 | 0.05732 | 0.98 | 0.78 | 0.05270 |
| 2 | 0.99 | 0.81 | 0.06329 | 0.98 | 0.76 | 0.04962 |
| 3 | 0.99 | 0.79 | 0.07934 | 0.99 | 0.77 | 0.06271 |
| 4 | 0.99 | 0.80 | 0.05914 | 0.99 | 0.76 | 0.04907 |
| 5 | 0.99 | 0.80 | 0.05236 | 0.98 | 0.78 | 0.05021 |

Table 5: Random Forest Model Performance

| | Male | | | Female | | |
|---|---|---|---|---|---|---|
| Model | Train | Test | Time (s) | Train | Test | Time (s) |
| Decision Tree | 1.00 | 0.75 | 0.01563 | 1.00 | 0.71 | 0.02072 |
| Random Forest | 0.99 | 0.80 | 0.06229 | 0.98 | 0.77 | 0.05286 |

Table 6: Average Model Performance Comparison

Figure 4 depicts the distribution of "decision" predictions of the test set for each gender for the random forest classifier. "yes" represents the participant being attracted to his/her partner while "no" suggests the participant was not attracted to his/her partner. True positives are the top left square, false positives are bottom left square, true negatives are bottom right square, and false negatives are top right square.




(a) Male dataset

(b) Female dataset

Figure 4: Sample confusion matrices for each gender dataset

### 4.3.3 Feature Importance

Feature importance is a property of the tree classifier that creates visibility for which features are most critical for the model to produce more accurate predictions.

| | Ranking | |
|---|---|---|
| Feature | Male | Female |
| (Physical) Attractiveness (importance) | 7 | 4 |
| Sincerity (importance) | 6 | 6 |
| Intelligence (importance) | 4 | 7 |
| Fun (importance) | 8 | 10 |
| Ambition (importance) | 11 | 8 |
| Shared Interest (importance) | 5 | 5 |
| (Physical) Attractiveness | 1 | 1 |
| Sincerity | 9 | 9 |
| Intelligence | 10 | 12 |
| Fun | 3 | 3 |
| Ambition | 12 | 11 |
| Shared Interest | 2 | 2 |

Table 7: Feature Importance Ranking (1 = most important, 12 = least important)

## 4.4 Analysis

### 4.4.1 Analysis of Different Splits

Tables 1 - 3 communicated that train and test accuracies for both gender datasets were relatively invariant from different train-test splits (70-30, 80-20, and 90-10). This suggests that increasing the number of training instances (decreasing the number of test instances) does not increase train accuracy or test accuracy. In the scope of this paper, we only experimented with three different splits. However, a more thorough exploration of train-test splits could produce higher accuracy scores. For example, a split of 50-50 could show different a different accuracy range compared to the splits that were used in this experiment.

### 4.4.2 Classification Model Comparison

Overall, decision trees were a decent classifier because the prediction accuracies for both datasets were fairly high. In every trial for each gender, the train accuracy was 1.00. The test accuracies ranged from 0.69 to 0.76. Having a fairly large discrepancy between the train and test accuracies (2.4-3.9% difference) communicates that the decision tree is likely overfitted to the training set. When comparing the accuracy measures overall between males and females, the decision tree model produced similar prediction results. In addition to having a decent prediction accuracy, decision trees did not require much time to build. Although there are some differences between the subdatsets in the time it took to build models, most models were built in less than 0.03 seconds.

Since the decision tree algorithm is likely overfitted, the random forest algorithm may be a better option for classifying the dataset. The results support this claim because the random forest algorithm alleviated the overfitting problem observed in the decision tree algorithm. When comparing the test accuracy for the decision tree and random forest models, the test accuracy of random forest is 4-8% higher than

the test accuracy of the decision trees.

Since the decision tree algorithm is likely overfitted, the random forest algorithm may be a better option for classifying the dataset. The results support this claim because the random forest algorithm alleviated the overfitting problem observed in the decision tree algorithm. Even though the random forest models had a lower train accuracy than the decision trees, the train and test accuracy for the random forest (1.8-2.3% difference) was closer than the accuracies for the decision tree (2.5-3.1% difference). The train accuracies for both algorithms were relatively the same (1.00), but the test accuracy for random forest was 3-9% higher than the test accuracy of the decision trees. Unfortunately, random forest models require more time to build. Table 6 and 7 indicate that the time required to build random forest model is twice as much as the time required to build decision tree model.

### 4.4.3 Confusion matrix

Confusion matrices were used to further describe the prediction accuracy of the random forest model. As mentioned in Figure 4, the top left corner of the matrix is the true positive rate (model successfully predicts participant attracted to partner) while the bottom right is the true negative rate (model successfully predicts participant is not attracted to partner). The other two corners correspond to predicting attracted when actually not attracted (false positive: bottom left) and predicting not attracted when actually attracted (false negative: top right).

Even though the prediction/classification percentages (numbers in each square in the confusion matrices) varied, the following patterns were reflected in both of the gender datasets. In Figure 4, the random forest model was the most successful with correctly predicting the participant found his/her partner attractive (true positive). The model performed a little worse when predicting if the participant found his/her partner unattractive (true negative). Therefore, the random forest model performed better with identifying "yes" (attracted) decisions as opposed to "no" (not attracted) decisions. In other words, the true positive rate was slightly higher than the true negative rate.

### 4.4.4 Important attributes of female and male datasets

Although we divided the original dataset into a female sub-dataset and male sub-dataset, the results in both sub-datasets were remarkably similar. Table 7 suggests that the top three important features for both genders were consistently (physical) attractiveness, shared interest and fun. It can be concluded that these three traits are essential when attracting the opposite sex, regardless of gender.

For the other features, different runs produced different rankings, so we cannot completely deem any derived conclusions as meaningful. Although the rankings don't completely align between the two genders, ambition is ranked fairly low for both genders, so we can loosely conclude that this trait may not be that important when attracting the opposite sex.

# 5 Conclusion

## 5.1 Summary

Even though there have been many studies that investigate the traits that make a person attractive, this experiment utilized machine learning algorithms (decision trees and random forest) to analyze the data gathered from various speed dating events to find traits that men and women found attractive in the opposite sex. Out of the myriad features collected in the dataset, only features that are strongly related to a person's human qualities were considered when building the models ((physical) attractiveness, sincerity, intelligence, fun, ambition, and shared interest). As a result, the random forest model performed better than the decision tree model with more accurately predicting the data in the test set. The random forest model suggested that the top 3 traits that both men and women found attractive in the opposite sex were the same and had the same order of attractiveness ((physical) attractiveness, shared interest, then fun).

## 5.2 Future Expansion

If more time was allotted for this experiment, we would try to enhance the model's prediction accuracy and potentially discover more patterns within the data. The scope of our problem was limited with only considering 12 features (2 for each trait: importance and evaluation) out of the multitude of columns. As mentioned in the analysis of different splits section (4.4.1), additional train-test splits for building the model would've been explored to see if there was a combination that optimizes train accuracy and test accuracy for this dataset. Performing dimension reduction (utilize feature importance results and/or correlation matrix) on the previously built random forest models would also potentially improve the prediction accuracy by removing noisy data.

There are primarily two methods with finding more patterns within the overall dataset: analyzing the current dataset from a different angle and expanding the scope of the data used to build the models. When using the same data, the importance of traits (pre-survey) and the evaluation of the traits (score card) could be compared to identify how closely perception (pre-survey) and reality (score card) of the attractiveness of traits align. In terms of expanding the scope of the data used, adding some of the features that were originally removed from the dataset (race, religious beliefs, occupation, etc.) could potentially produce more patterns that were not captured with only analyzing 6 traits.

An expansion of the experiment that could potentially improve both accuracy and discovering additional patterns is to continue to collect data by organizing more speed dating events. The existing dataset suggests that men and women view (physical) attractiveness, shared interests, and fun as the most attractive traits in a partner. If the data was collected over a larger time frame, the patterns suggested by the model could align better with society's perception that men and women are attracted to different qualities in the opposite sex.

# 6 Writing Distribution

| Section | Description | Person in charge |
|---|---|---|
| 1.1-1.2 | Overview, Dataset | Tu Do |
| 1.3 | Algorithms | Neva Lui |
| 2.1 | Decision Trees | Tu Do |
| 2.2 | Random Forest | Neva Lui |
| 3 | Problem Approach | Tu Do, Neva Lui |
| 4.1.1 | Data Preprocessing | Tu Do |
| 4.1.2-4.2 | Building the Learning Model, Solution Evaluation | Neva Lui |
| 4.3 | Results | Tu Do, Neva Lui |
| 4.4 | Analysis | Tu Do, Neva Lui |
| 5 | Conclusion | Tu Do, Neva Lui |

# References


Bhargava, N., G. Sharma, R. Bhargava, and M. Mathuria
2013. Decision tree analysis on j48 algorithm for data mining. *Proceedings of International Journal of Advanced Research in Computer Science and Software Engineering*, 3(6).

Breiman, L.
1996. Bagging Predictors. *Machine Learning*, 24:123–140.

Breiman, L.
2001. Random Forests. *Machine Learning*, 45-1:5–32.

Breiman, L.
2004. Consistency for a Simple Model of Random Forests. Technical report, University of California at Berkeley.

Cinaroglu, S.
2016. Comparison of performance of decision tree algorithms and random forest: An application on oecd countries health expenditures. *International Journal of Computer Applications*, 138(1):37–41.

Fayyad, U. M. and K. B. Irani
1992. The attribute selection problem in decision tree generation. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI'92, Pp. 104–110. AAAI Press.

Fisman, R., S. S. Iyengar, E. Kamenica, and I. Simonson
2006. Gender Differences in Mate Selection: Evidence From a Speed Dating Experiment*. *The Quarterly Journal of Economics*, 121(2):673–697. data retrieved from Gender Differences in Mate Selection, http://faculty.chicagobooth.edu/emir.kamenica/documents/genderDifferences.pdf.

Loh, W.
2011. Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):14–23.

Patel, N. and S. Upadhyay
2012. Study of various decision tree pruning methods with their empirical comparison in WEKA. *International journal of computer applications*, 60(12).

Ray, S.
2017. Essentials of Machine Learning Algorithms (with Python and R Codes). https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/.

Souza, C.
2012. Decision trees in c sharp.

Tan, P.-N., M. Steinbach, V. Kumar, and Z. Tang
2005. Introduction to Data Mining.