

**WID3006 Machine Learning**

**Semester 2, 2020/2021**

**Final Report - Love Matching**

**Between Two Individuals Through First Date**

**Lecturer:Dr. Aznul Qalid Md Sabri**

**Group Members:**

| **No** | **Reg. No. (New)** | **Reg. No. (Old)** | **Name** | **Tutorial Group** |
| --- | --- | --- | --- | --- |
| 1 | 17206272/1 | WID190510 | Ng Yong Zhang | T2 |
| 2 | 17205051/1 | WID190008 | Chia Jing Hui | T2 |
| 3 | 17204964/1 | WID190006 | Chan Jia Wei | T2 |
| 4 | 17201811/1 | WID190057 | Tong Boon Onn | T2 |
| 5 | 17203068/1 | WID190003 | Alvin Ang Zee Qin | T2 |

**Table of Contents**

1. **Introduction to Problem**
2. **Project Objectives**
3. **Methodology**
4. **Theory**
5. **Source code**
6. **Elaboration on Data & Features Used**
7. **Results**
8. **Discussions & Observation**
9. **Suggestion for future works**
10. **Appendix**

**Introduction to Problem**

Love can be defined as an emotion that keeps people bonded and committed to one another. Being in love involves a feeling of sexual arousal and attraction between two individuals. From a more scientific perspective, much of love can be explained in chemistry. Each of the stages in a romantic love is very much associated with its own set of hormones stemming from the brain. For instance, being attracted to someone simply means that a high level of dopamine and norepinephrine are released that gives you this giddy, energetic and euphoric feeling. However, is there really a “formula” for love? Can we really determine whether two individuals are in love just by looking at statistics?

It has been a challenge to define a definite method of classifying love due to the subjectiveness of the topic. However, it is undeniable that we do have a basic standard when choosing mates. There might be some important qualities or characteristics that we are looking for in our life partner. Take for instance age, race, hobby, life habit, career and income. These are all essential information that we would gather to determine whether our partner is the “right one”.

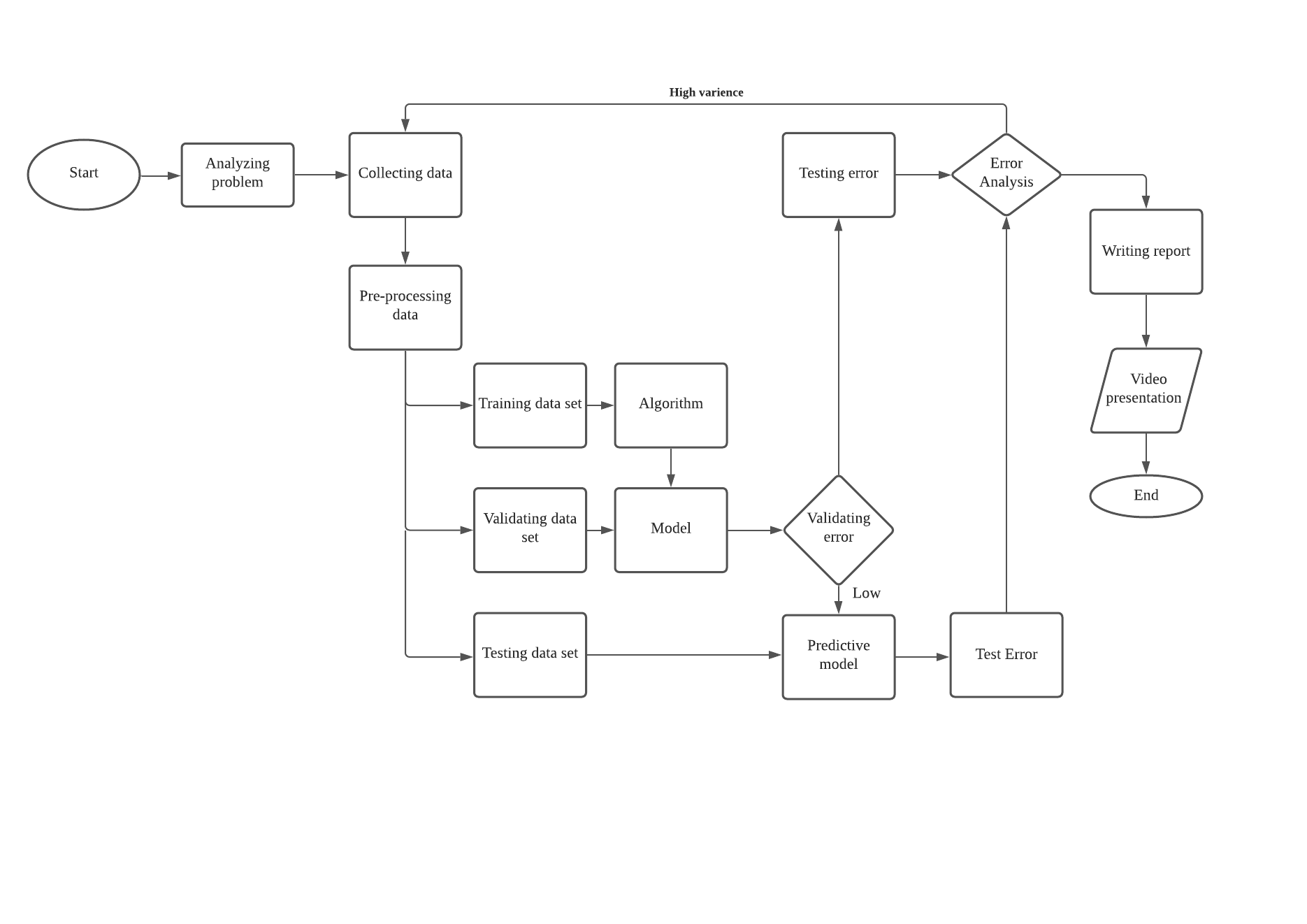
A [dataset compiled](http://www.stat.columbia.edu/~gelman/arm/examples/speed.dating/) by Columbia Business School professors Ray Fisman and Sheena Iyengar gathered the data from participants in an experimental speed dating event. At the end of the dating, participants were asked if they would like to see their date again. They were also asked to rate their date on some attributes. The result also includes whether they became a couple in the future.

Our problem statement is to develop a machine learning model to predict whether a pair of individuals would fall in love at the end using the data collected in their first date. The model would take the attributes as input and output a single classification of whether the two individuals are a match or not.

**Project Objectives**

* Build a model to classify whether a pair of individuals would fall in love based on their personal features
* Determine the most essential attributes that a male or female partner is looking for when choosing life partner
* Develop insights and hypothesis about the model predicting love and common approaches for future works

**Methodology**

The figure below is the protocol that we originally designed to solve the problem. The overall process can be divided into 3 main stages, data preprocessing, modelling and performance analysis

**Data Preprocessing:**

The data that we used is a [dataset](http://www.stat.columbia.edu/~gelman/arm/examples/speed.dating/) compiled by Columbia Business School professors Ray Fisman and Sheena Iyengar which is available on kaggle dataset website. Data was gathered from participants in experimental speed dating events from 2002-2004. During the events, the attendees would have first date with every other participant of the opposite sex. At the end of their dating, participants were asked if they would like to see their date again. They were also asked to rate their date on six attributes: Attractiveness, Sincerity, Intelligence, Fun, Ambition, and Shared Interests.The result also includes whether they became a couple in the future.

We then up-sample the no-match data using the Smote algorithm(Synthetic Minority Oversampling Technique) that creates synthetic samples from the minor class instead of creating copies. After the SMOTE operation, our training data had been well balanced. We over-sampled only on the training data, because by oversampling only on the training data, none of the information in the test data is being used to create synthetic observations, therefore, no information will bleed from test data into the model training. After that, we perform recursive feature elimination(RFE) to choose the best performing feature. This process is applied until all features are exhausted. The features with p-value > 0.05 will be filtered out.

**Modelling:**

After getting ready with the data, we split the data into 70% train set and 30% test set. We use a simple logistic regression classifier to obtain predictions and calculate the accuracy of the prediction.

**Performance Analysis:**

After modelling, we built up the analyzing codes. Confusion matrix is selected to be the evaluation matrix. The confusion matrix will show us how accurate our model is by determining the true positive, true negative, false positive and false negative cases. The y-axis represents the actual label while the x-axis represents the predicted label. Next, we used the confusion matrix to compute precision, recall, F-measure and support using the following formula:

The confusion matrix helps a lot in analyzing the performance of the models on the train set and also the test set to see whether it is predicting certain classes too frequently causing the model to have high precision and high accuracy. Besides, we also built up a correlation matrix to determine which attributes are correlated and unrelated with each other. For example, the first correlation matrix measures the correlation between six attributes that we collected in the “How do you think you measure up” section. After everything has been done and the model has been finalized, we export the model and start writing the report

**Theory**

**Logistic Regression**

In statistics, the logistic model is used to calculate the probability of a certain class or event existing. While in machine learning, logistic regression is an algorithm which is used to classify the data or observation to a discrete set of classes by considering outcome variables on extreme ends and tries to make a logarithmic line that distinguishes between them. Logistic regression is used when the dependent variable is categorical. Classification on whether an email is spam or not, tumor is malignant or benign, are the common examples of logistic regression.

Logistic Regression hypothesis expectation: 0 ≤ hθ(x) ≤ 1

y ∈ {0,1} where 0 is “Negative Class” and 1 is “Positive Class”

P( y = 0 | x;θ) + P( y = 1 | x;θ) = 1

Compared to linear regression, logistic regression implements a more complicated cost function, ‘sigmoid function’ to carry out classification. This is because the hypothesis of logistic regression tends to limit the cost function between 0 and 1, thus the hypothesis of linear regression is not suitable to be implemented in logistic regression as the hypothesis of linear regression may be greater than 1 or less than 0. Hence, in order to map the predicted values to possibilities, sigmoid function is used as it maps any real value into another value between 0 and 1.



Sigmoid Function Graph

Referring from the sigmoid function graph, if ‘t’ goes to positive infinity, y = 1 and if ‘t’ goes to negative infinity, y = 0.

**Hypothesis representation**

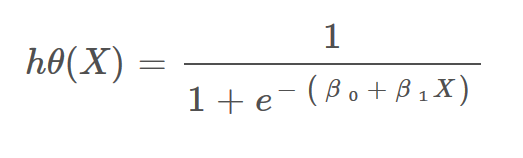
To predict and classify the input from the training data sets, a hypothesis function, hθ(x) is used to estimate the Y value when x=1 in logistic regression. Since we need the classifier to output values between 0 and 1, the sigmoid function is used.

Z = β₀ + β₁X

Sigmoid(z) = 1/(1 + e*-z*)

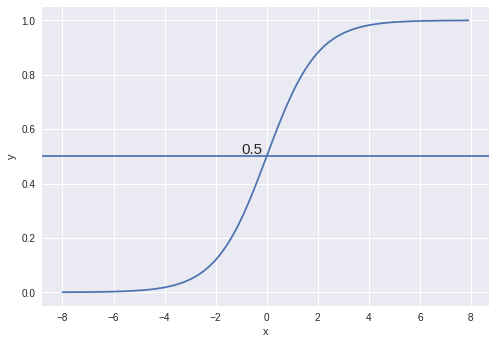
hθ(x) = sigmoid(Z)

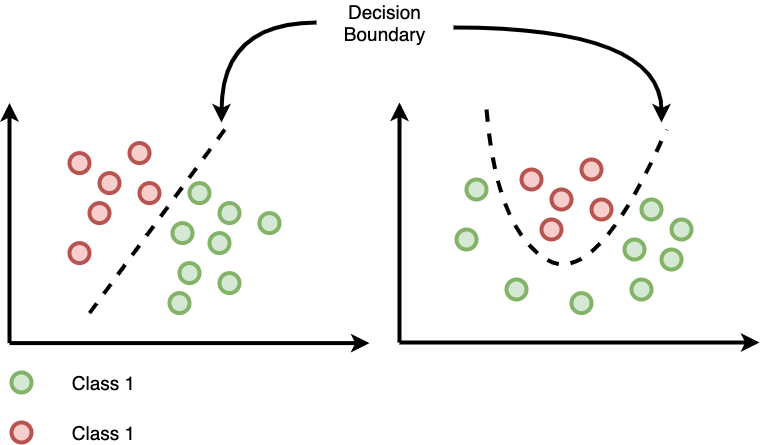
Hence,



**Decision Boundary**

To predict which class a data belongs to, a threshold can be set. Based upon this threshold, the obtained estimated probability is classified into classes. For example, the logistic regression model is used to classify whether an email is spam(1) or not a spam(0), z = β₀ + β₁X and threshold is 0.5. If z is above threshold(z ≥ 0.5), then the email is classified into spam else if z is below the threshold, the email will classify into not a spam. Decision boundary can be linear or non-linear. Polynomial order can be increased to get a complex decision boundary.





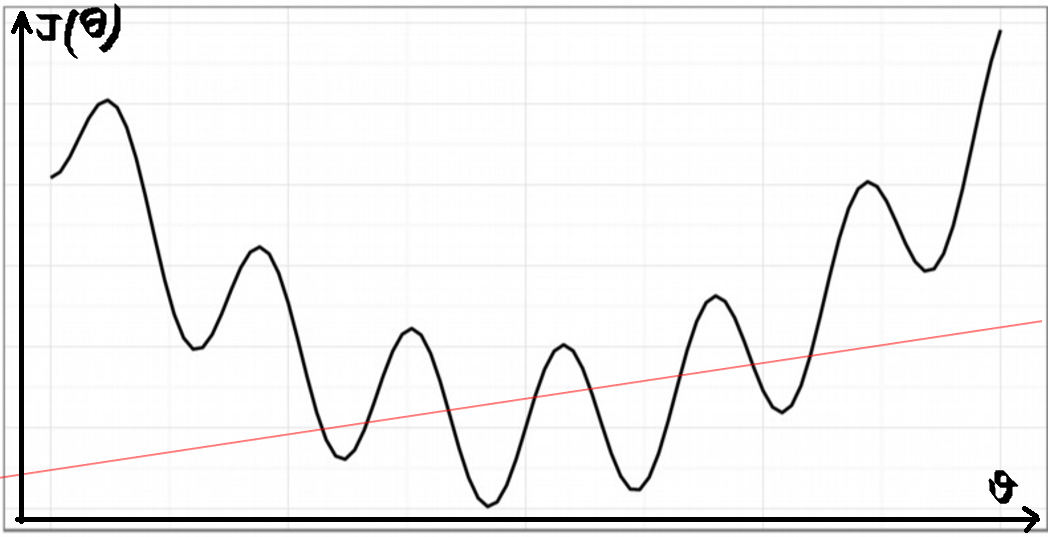
Example of linear and non-linear decision boundary

**Cost Function**

To create an accurate model with minimum error, a cost function, J(θ) is used to find the errors between actual output and predicted value. For linear regression, the cost function, J(θ) is calculated using mean squared error. It is a convex function and can be optimized easily using gradient descent.

The Cost function of Linear regression

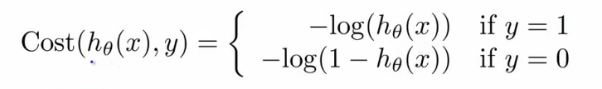
But, the hypothesis function of logistic regression has a non-linearity, where there is no direct relationship between an independent variable and a dependent variable. This is a complicated non-linear function which is different from linear regression. Thus, if we use the same cost function for the logistic regression, it would end up with a non-convex function with many local minimums, in which it would be very difficult to minimize the cost value and find the global minimum.

Non-convex function

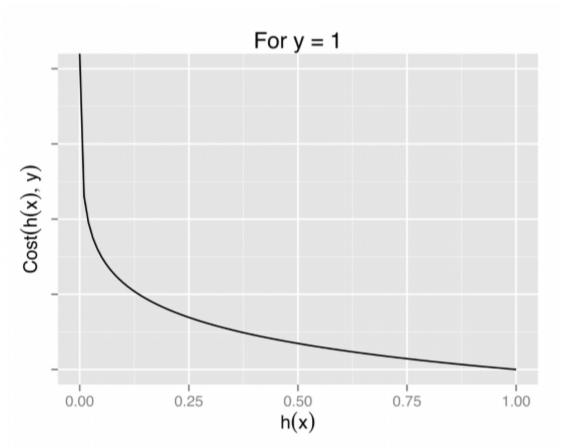
Hence, a different, convex cost function for logistic regression had come out to overcome this problem. Firstly, J(θ) is redefined as

where

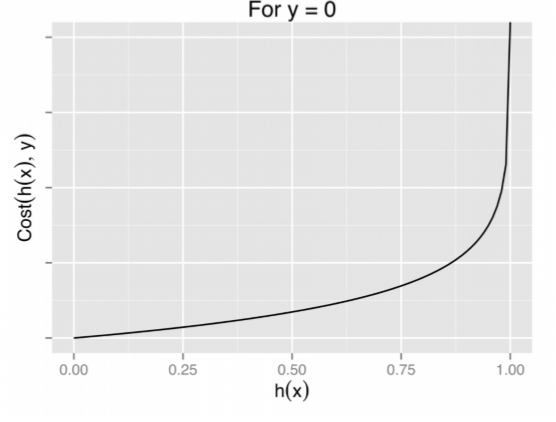
which is the sum of all the individual costs over the training data and same as linear regression. Then, the log function is used for the cost function of logistic regression.

Cost function of logistic regression

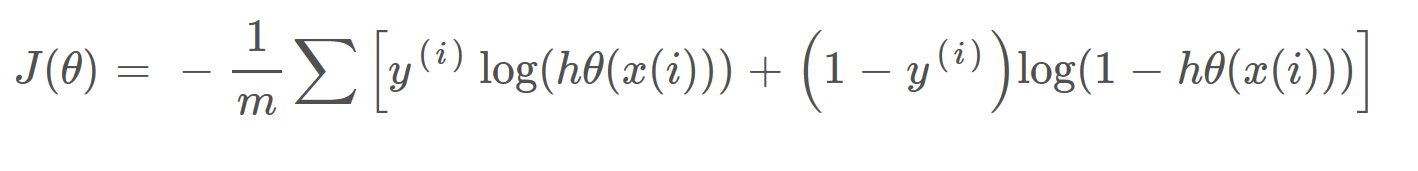
According to the formula above, when we plot the cost function when y = 1, the graph is shown below. When the hypothesis function(hθ(x)) predicts the output exactly as the actual output, the cost function is 0 which means there is no error between y\_actual and y\_predicted. But if the hθ(x) goes to 0, the cost will become infinity and penalize the learning algorithm with a massive cost.

Graph of cost function when y = 1

For y = 0, we inverse the cost function when y = 1. Hence, when y = 0 and hθ(x) goes to 1, the cost will become infinity and when y = 0 and hθ(x) goes to 0, the cost is 0. By applying these formulas, the cost function for logistic regression is going to be convex and avoid local minimum.

Graph of cost function when y = 0

The cost function of logistic regression can be compressed as

So when y = 1, 

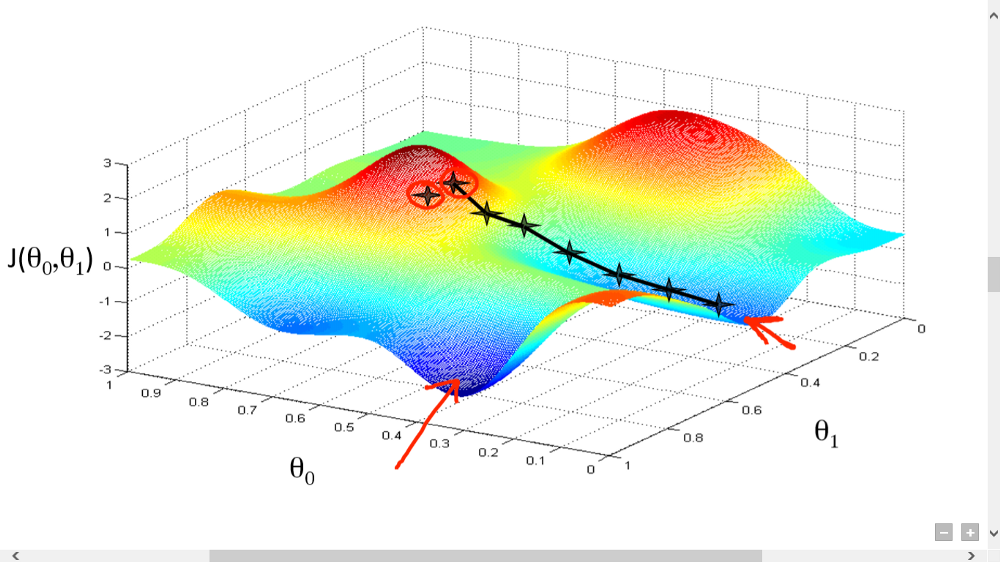
J(θ) = -log(hθ(x)) - (0)log(1 - hθ(x))

= -log(hθ(x))

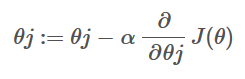
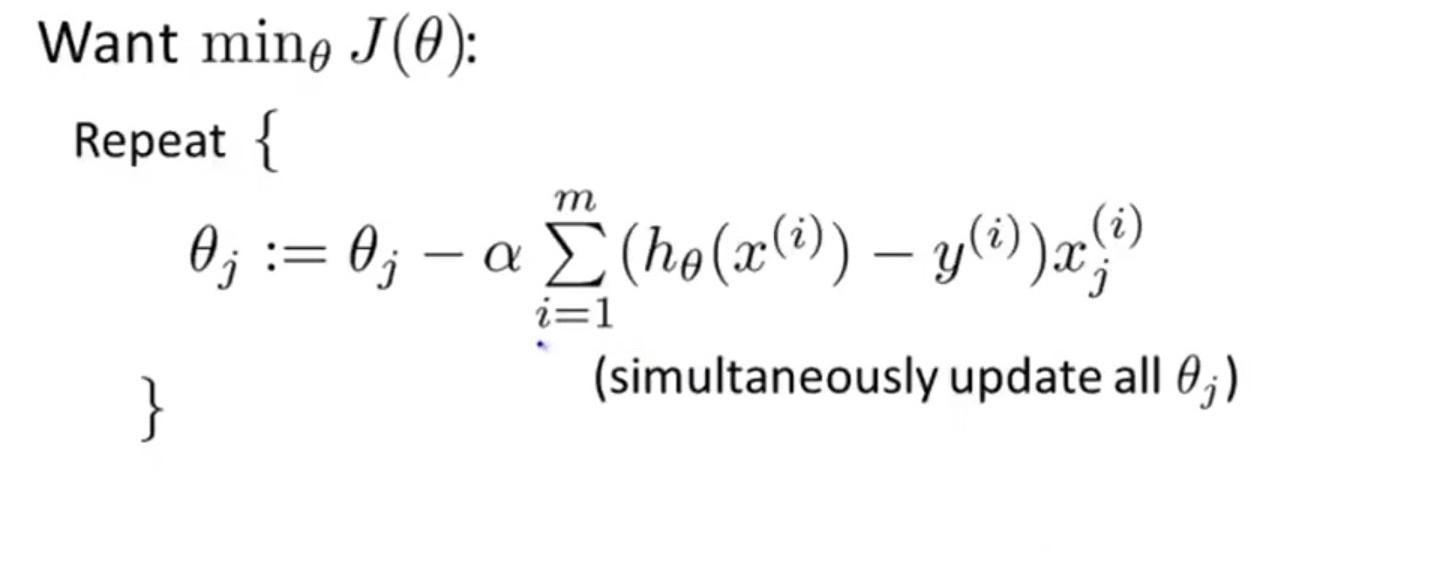
when y = 0,

J(θ) = -(0)log(hθ(x)) - (1)log(1 - hθ(x))

= - log(1 - hθ(x))

**Gradient Descent**

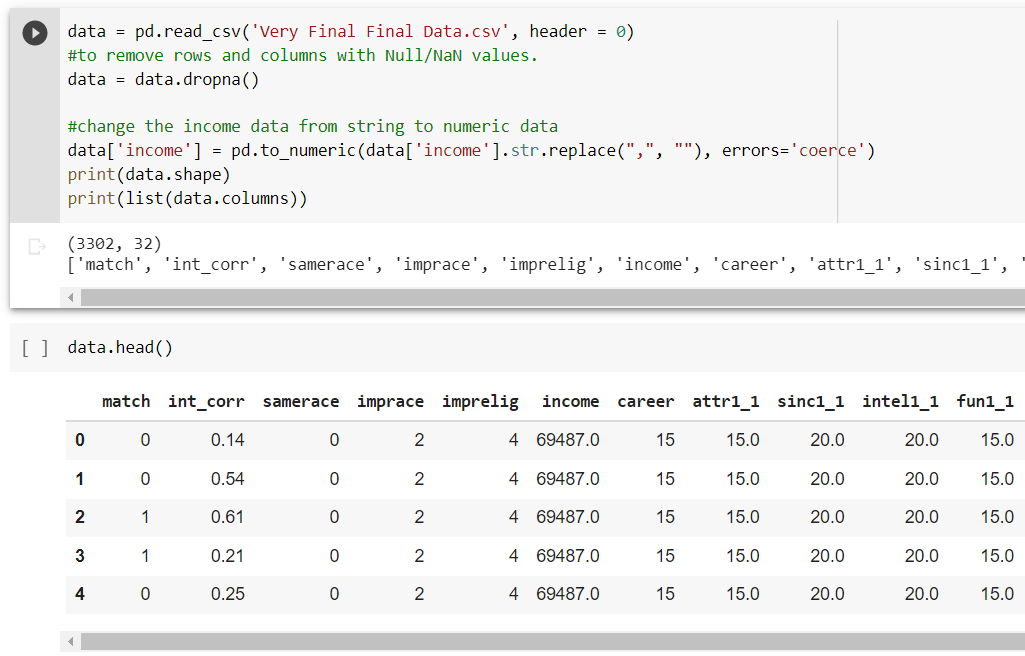
Gradient Descent Analogy

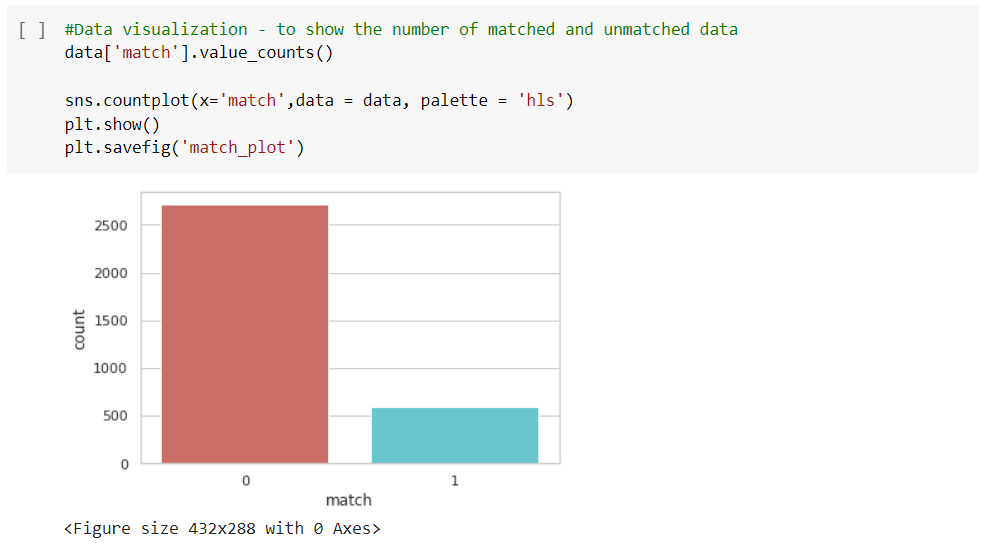
To increase the accuracy of hypothesis function, we need to carry out gradient descent. Gradient descent is an optimization algorithm which is used to minimise some function by iteratively moving in the direction of steepest descent as defined as the negative of the gradient. In machine learning, we use gradient descent to find the global minimum of the graph of cost function versus x features. In gradient descent, we will need to repeatedly update the parameters of the logistic regression model using a learning rate.The size of these steps is called the learning rate. With a high learning rate we can cover more ground each step, but we risk overshooting the minimum and fail to converge since the slope of the hill is constantly changing. With a very low learning rate, we can confidently move in the direction of the negative gradient since we are recalculating it so frequently. A low learning rate is more precise, but calculating the gradient is time-consuming, so it will take us a very long time to get to the bottom.

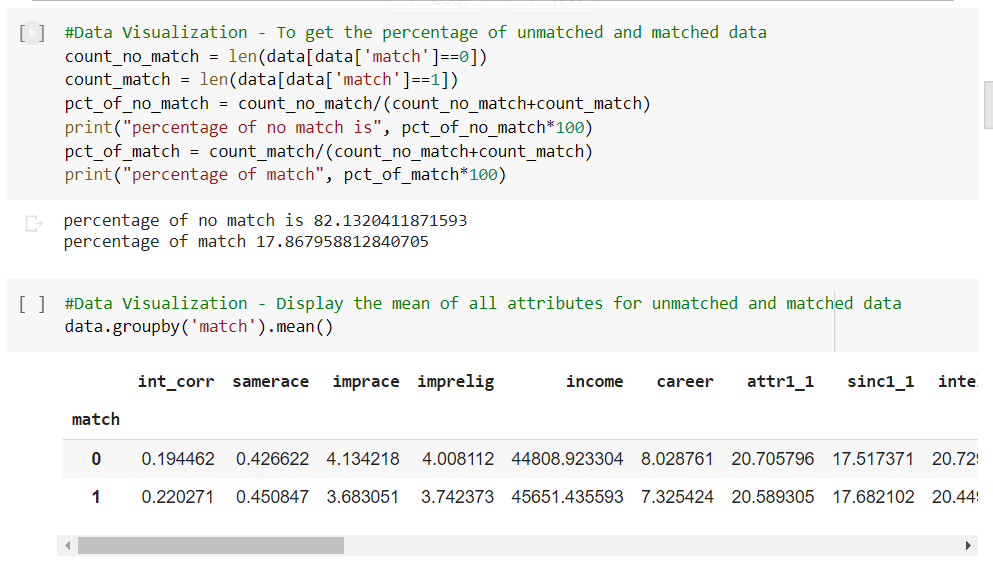
Simplified Gradient Descent

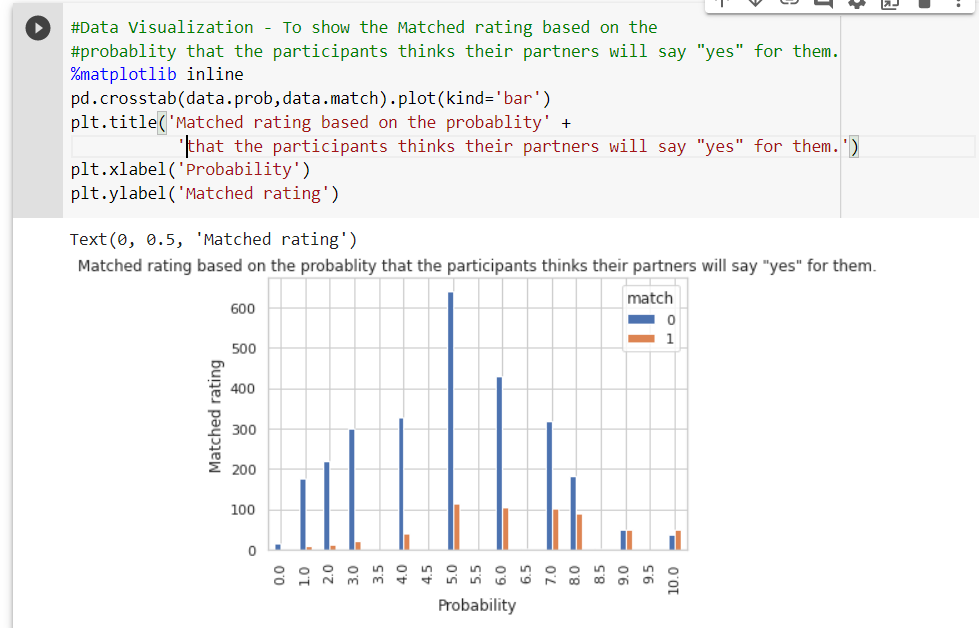
**Source Code**

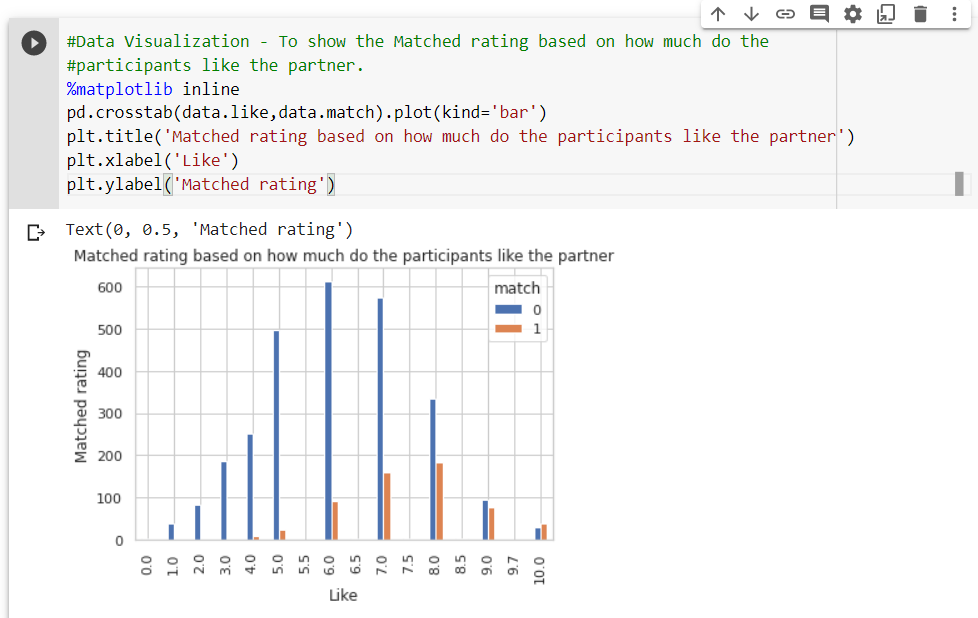
****

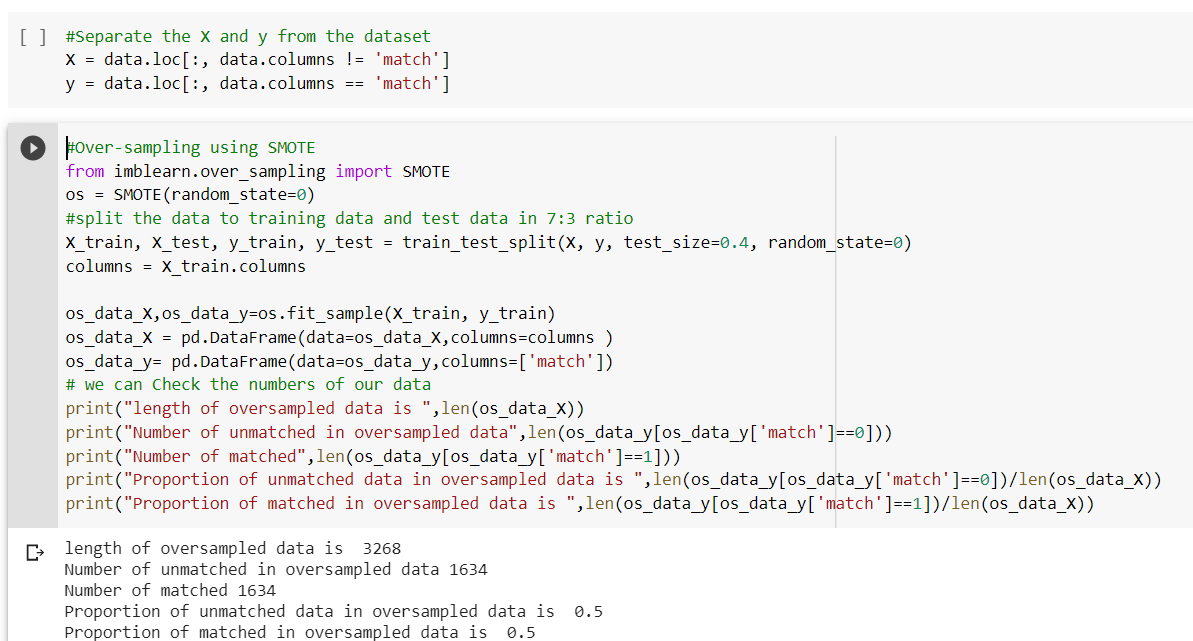


****

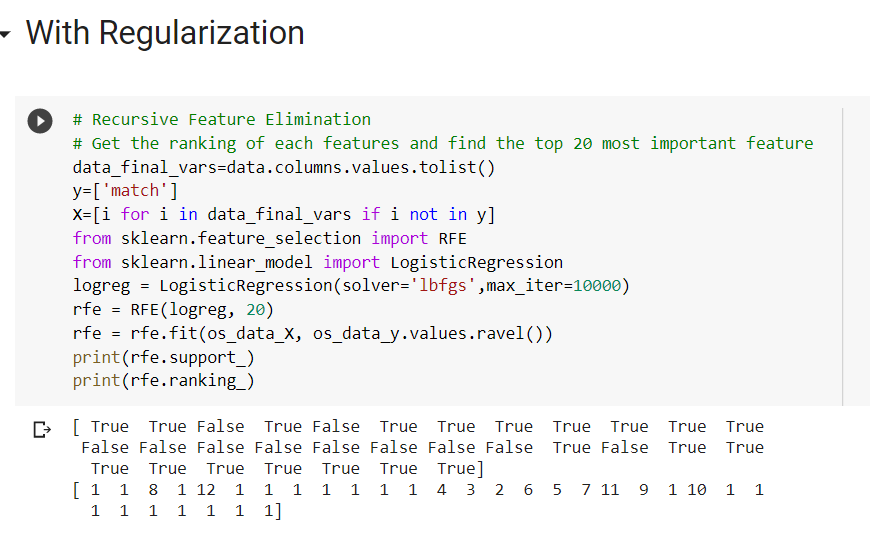
****

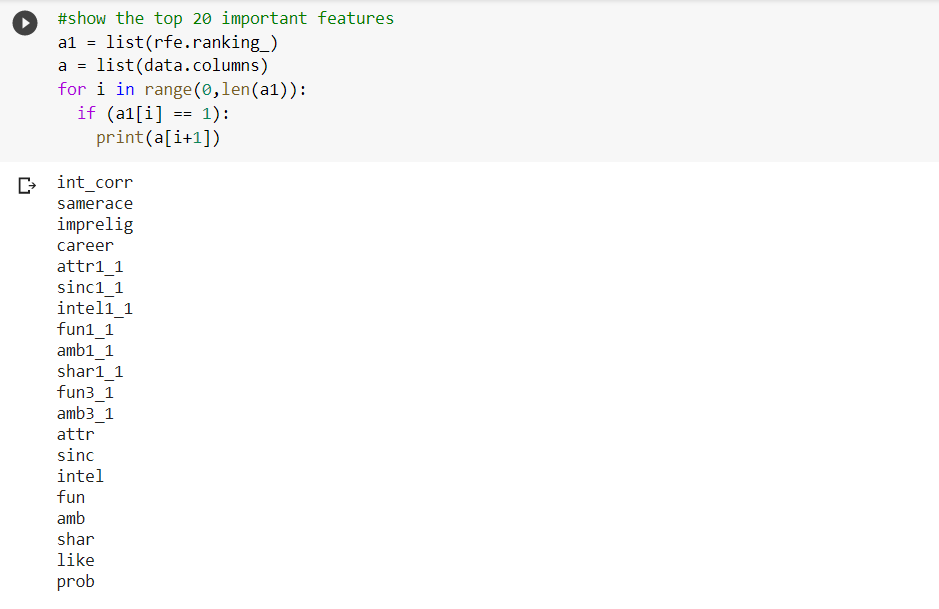
****

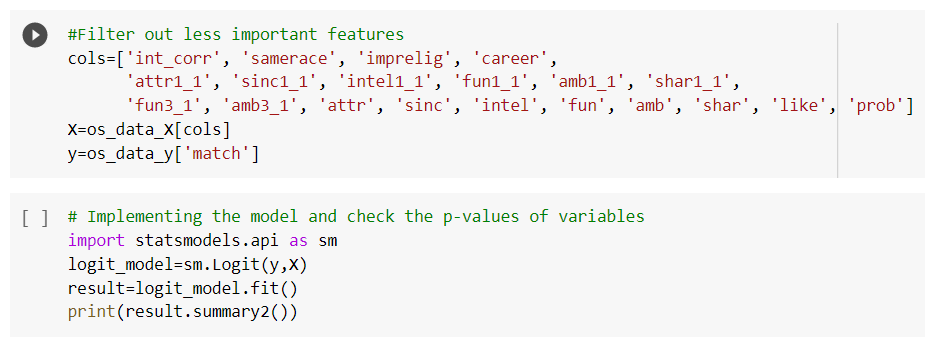
****

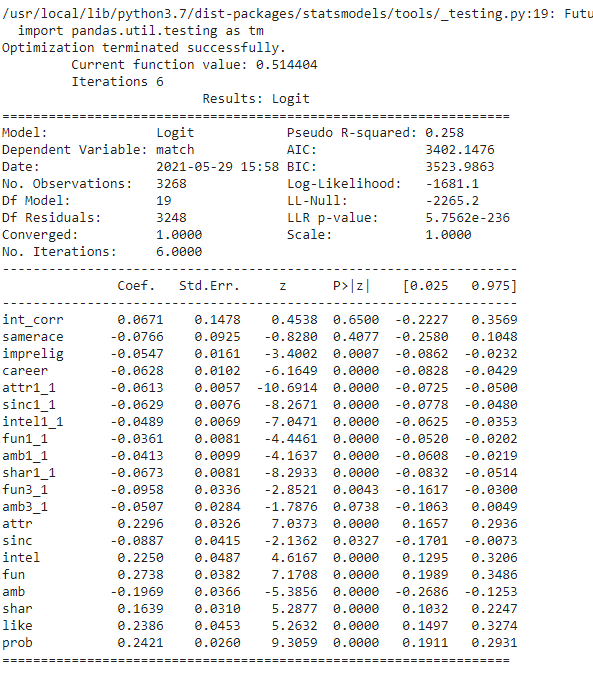
****

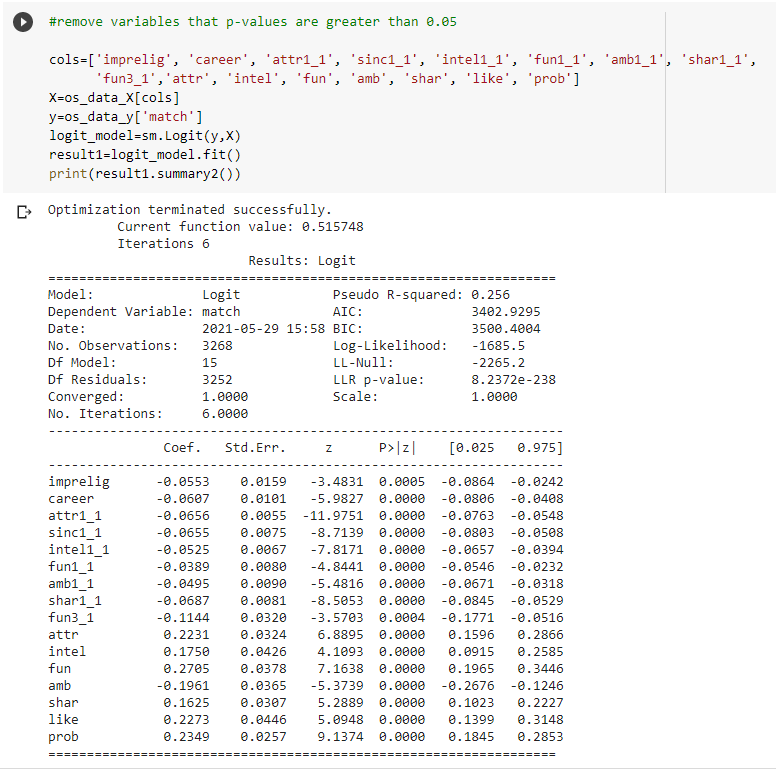
****

****

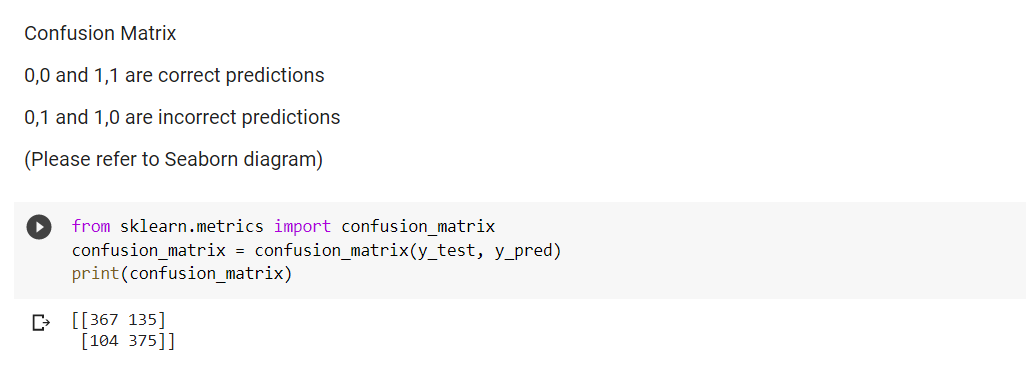
****

****

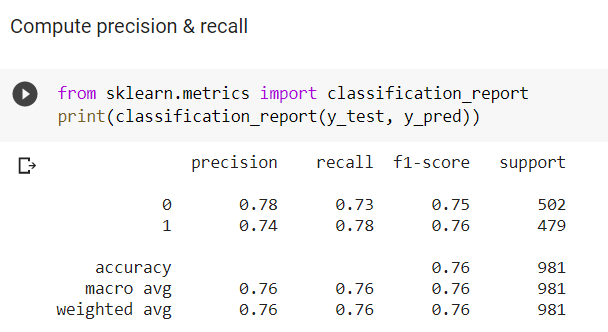
****

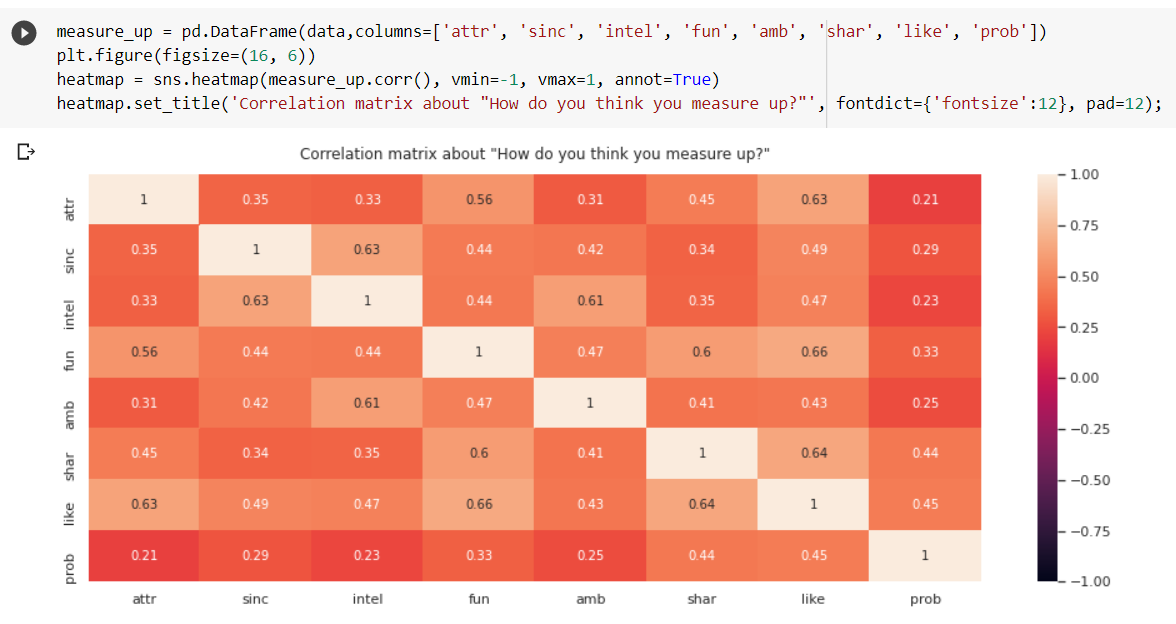
****

****

****

****





**Elaboration on Data & Features Used**

What influences love at first sight? (Or, at least, love in the first four minutes?) This [dataset](http://www.stat.columbia.edu/~gelman/arm/examples/speed.dating/) was compiled by Columbia Business School professors Ray Fisman and Sheena Iyengar which is available on kaggle dataset website: <https://www.kaggle.com/annavictoria/speed-dating-experiment>.

Data was gathered from participants in experimental speed dating events from 2002-2004. During the events, the attendees would have first date with every other participant of the opposite sex. At the end of their dating, participants were asked if they would like to see their date again. They were also asked to rate their date on six attributes: Attractiveness, Sincerity, Intelligence, Fun, Ambition, and Shared Interests.The result also includes whether they became a couple in the future.

The dataset also includes questionnaire data gathered from participants at different points in the process. These fields include: demographics, dating habits, self-perception across key attributes, beliefs on what others find valuable in a mate, and lifestyle information.

The idea of this project is we want to determine what are the least desirable attributes in a male or female partner. For example, ethnic background and religious background may be important attributes for being a couple. Hence, the attributes that are included for our machine learning model training are listed and explained below.

**Features in dataset (**highlight is the key in the dataset)

**Result: match 1=yes, 0=no**

**Q1: samerace: the participant and the partner were the same race. 1= yes, 0=no**

**Q2: How important is it to you (on a scale of 1-10) that a person you date be of the same racial/ethnic background?**

**imprace**

**Q3: How important is it to you (on a scale of 1-10) that a person you date be of the same religious background?**

**imprelig**

**Q4: How much is your annual income? (in RM)**

**Income**

**Q5: What is your career? （refer to career id below)**

**Career\_**

**Q6:**

| **On which attributes that you care more in finding ideal partners (scale 1-10)** | |
| --- | --- |
| **Attributes** | **Scale(1-10)** |
| Attractive (attr1\_1 ) |  |
| Sincere (sinc1\_1) |  |
| Intelligent (int1\_1) |  |
| Fun (fun1\_1) |  |
| Ambitious (amb1\_1) |  |
| Has shared interests/hobbies (shar1\_1) |  |

**Q7:**

| **How does the respondent think on what opposite sex looks for and his/her own measure up?(scale 1-10)** | |
| --- | --- |
| **Attributes** | **Scale(1-10)** |
| Attractive (attr2\_1 ) |  |
| Sincere (sinc2\_1) |  |
| Intelligent (int2\_1) |  |
| Fun (fun2\_1) |  |
| Ambitious (amb2\_1) |  |
| Has shared interests/hobbies (shar2\_1) |  |

**Q8:**

| **Give points on yourself from scale 1 to 10.** | |
| --- | --- |
| **Attributes** | **Scale(1-10)** |
| Attractive (attr3\_1 ) |  |
| Sincere (sinc3\_1) |  |
| Intelligent (int3\_1) |  |
| Fun (fun3\_1) |  |
| Ambitious (amb3\_1) |  |

**Q9:**

| **Rate their attributes on a scale of 1-10: (1=awful, 10=great) after the first date with him,/her.** | |
| --- | --- |
| **Attributes** | **Scale(1-10)** |
| Attractive |  |
| Sincere |  |
| Intelligent |  |
| Fun |  |
| Ambitious |  |
| Shared Interest/ Hobbies |  |
| Overall, how much do you like this person?  (1= I don’t like at all, 10= I like a lot) |  |
| How probable do you think the person will say yes for you? (1= not probable, 10= extreme probable) |  |

**career\_cd:**

1 = Law

2 = Academic/Research

3 = Computer / Informatics

4 = Doctor/Medicine

5 =Engineer

6 = Creative Arts/Entertainment/novelist

7 = Student

8 = Banking/Consulting/Finance/Marketing/Business/CEO/Entrepreneur/Admin

9 = Education

10 = Undecided /jobless

11 = Social Work

12 = Speech Pathology

13 = Politics

14 = Pro sports/Athletics

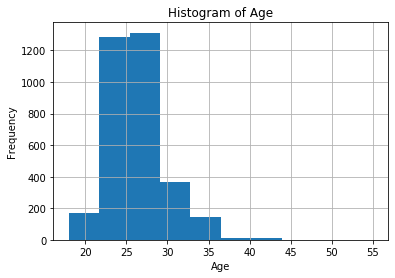
15 = Other

17 = Architecture

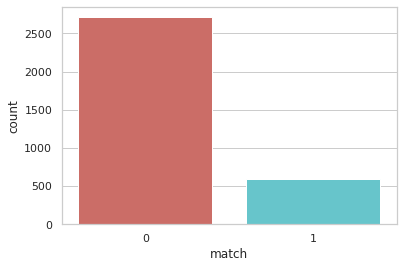
18 = Real Estate

19= International/Humanitarian Affairs

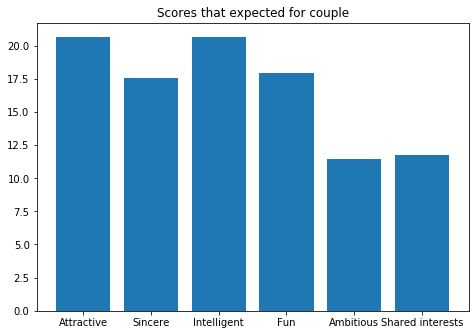
After carrying out data filtering, the dataset contains 3302 records and we start to analyze the data. The graph below shows the average age of participants in the dataset. Most participants aged between 22 to 28 were included in the questionnaire.

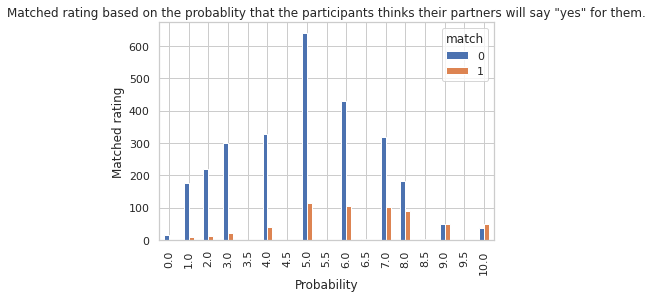
****

The graph below shows the property of the datasets with a total 3303 participants data used for our model. In the result, the number of unmatched results exceeds 2500 and the data of the match result is around 600.

****

Next, the graph below shows the average marks given to the opposite sex after the first dating. It shows that “attractiveness” and “intelligence” get the highest score among all attributes as these two attributes can be easily shown in the dating. In contrast, “ambitious” and “shared interests” are given the lowest marks as these attributes will only be shown after having deep and long conversation between couples and hardly shown in first and speedy dating.

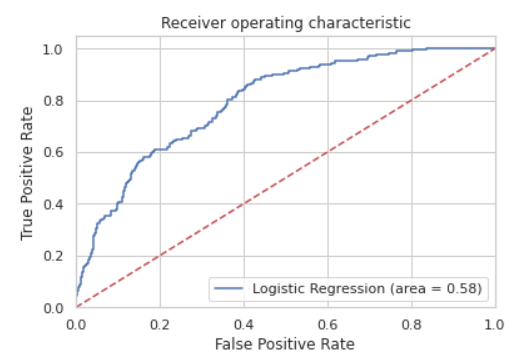
****



The diagram above shows the matched rating based on the probability that the person thinks their partner will say “yes” for them where 1 means not probable and 10 means extreme probable. We can find that the ratio of unmatched couples to matched couples increases when probability increases from 0 to 3, and achieves the highest ratio, which is approximately 20:1 when the participants rate “3 marks” that their partner will say yes to them. After that, the ratio of unmatched couples to matched couples decreases when probability increases from 4 to 10, and achieves the lowest ratio, which is 0.79:1 when probability equals to 10. This graph proved that, when the participants have higher confidence towards their partner after the first speed dating, the higher the possibility that they will fall in love with each other after the speed dating.

**Results**

**ROC Curve**

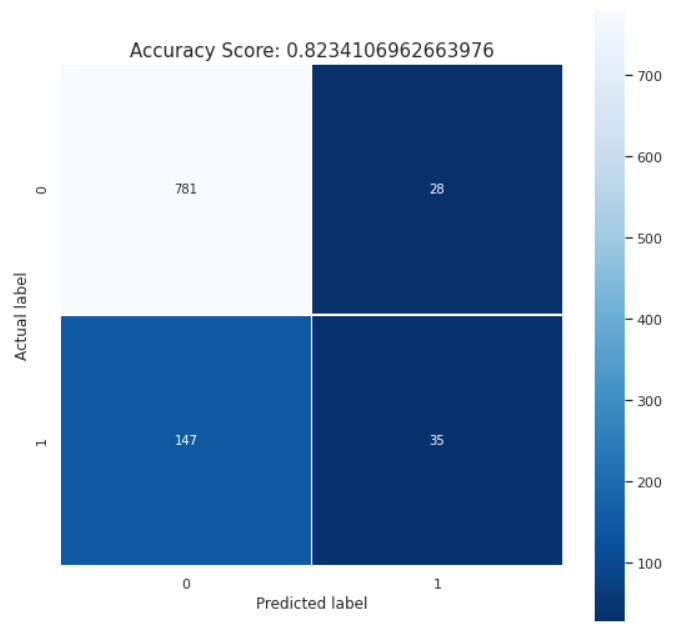
****

The Roc curve is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0. Put another way, it plots the false alarm rate versus the hit rate.

The true positive rate is calculated as the number of true positives divided by the sum of the number of true positives and the number of false negatives. It describes how good the model is at predicting the positive class when the actual outcome is positive.

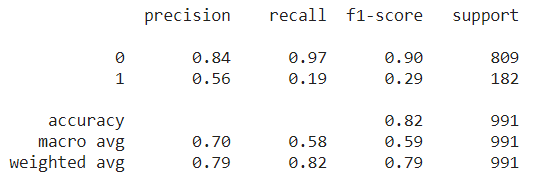
Our model with perfect skill is represented at a point (0,1). A model with perfect skill is represented by a line that travels from the bottom left of the plot to the top left and then across the top to the top right. An operator may plot the ROC curve for the final model and choose a threshold that gives a desirable balance between the false positives and false negatives.

**Confusion Matrix**

****

The diagram above shows the confusion matrix of this model. The accuracy score for this model is 0.83. With regularization, the accuracy will reduce to 0.76. The different layers of blue colour represent the numbers of the label. The y-axis represents the actual label while the x-axis represents the predicted label. Moreover, the diagonal of the confusion matrix represents the correct predictions of the outputs. Thus, this table shows that this model has 816 correct predictions and 175 incorrect predictions.

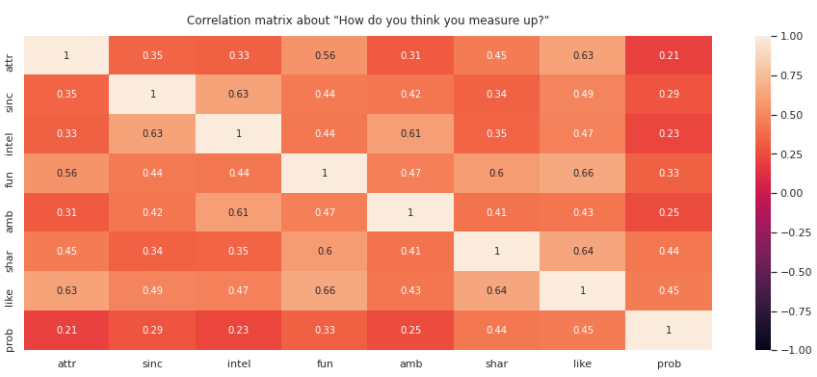
**Compute precision, recall, F-measure and support**

****

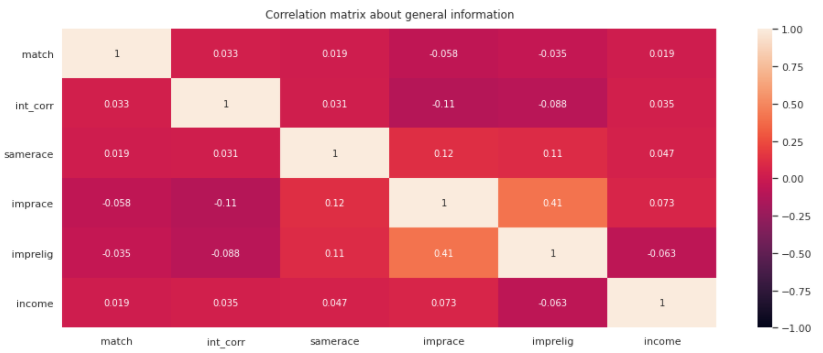
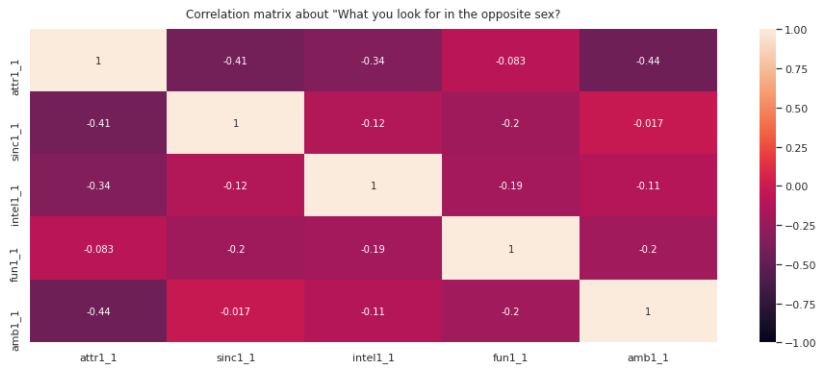
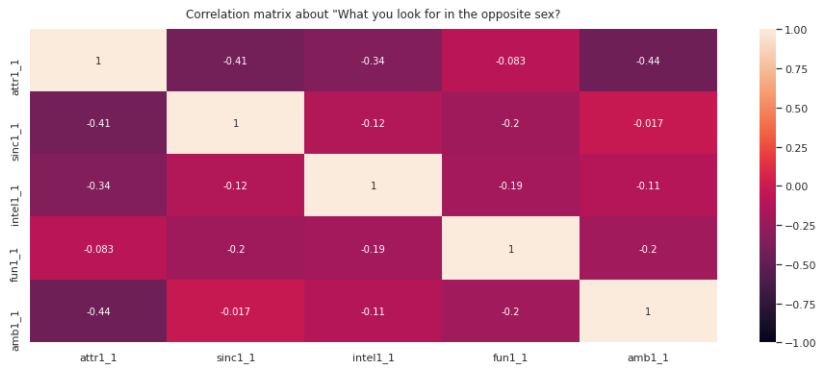
From the previous matrix confusion, the sum of the values in the m column of the n row is the sum of the true positive and false positive for the row. Thus, all these values are used in computing the precision, recall, F-measure and support. 0 represents the failure to be a couple between two persons while 1 represents the couple formed successfully. In this table, the precision and recall for 0 class is quite satisfied which are 0.78 and 0.73 respectively which shows that it has a high accuracy. However, the 1 class has a precision of 0.56 and a recall of 0.19 which has a lower accuracy. By referring to the table, the accuracy of F1-score is 0.82 and to improve this model, we might consider some external factors such as diseases, other factors from family background like their mind set, and also destiny which is unpredictable.

**Correlation matrix**

Correlation matrix is used to determine which attributes are correlated and unrelated with each other. The attributes with a higher correlation will give more impacts with each other.



The first correlation matrix is about “How do you think you measure up?”. In this table, all the attributes mostly have some correlation with each other in a range of 0.21 to 0.66. The most related attributes are fun and like they are enjoyed with their partners which they may like.

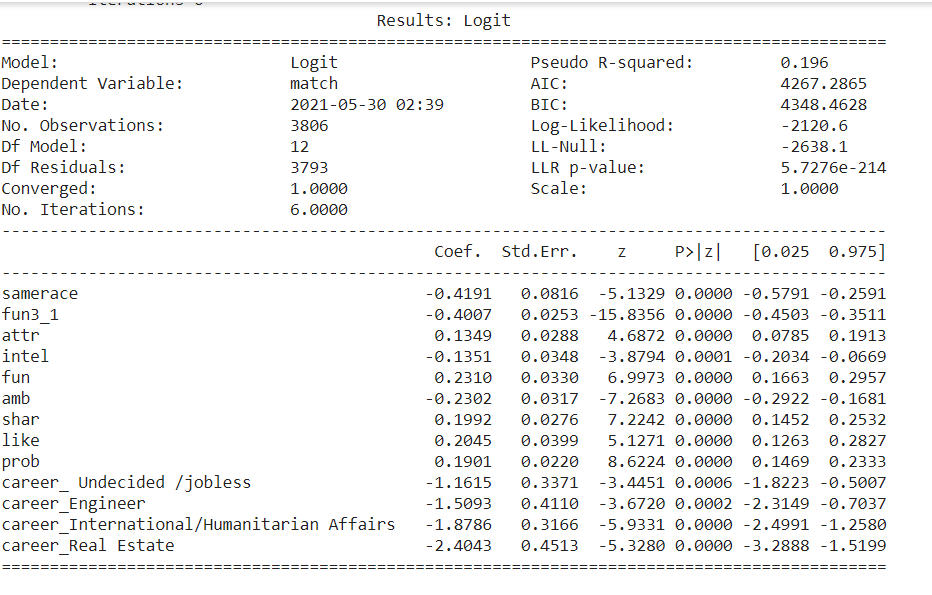
The second correlation matrix is about “What you look for in the opposite sex”. As we can see that all the values are negative, all these attributes are unrelated with each other as they are measured in different specifications. 

The third correlation matrix is about general information. It has the values in a range of -0.11 to 0.41 which shows that the attributes are mostly unrelated to each other. In addition, only the imprace and imprelig is correlated with each other because these two attributes are the scales about the importance of a person he or she dates be of the same racial and religious background respectively. Basically, racial background might affect religious background and vice versa.

**Discussions & Observation**

Before using the dataset for model training, we up-sample the no-subscription using the Smote algorithm(Synthetic Minority Oversampling Technique). At a high level, SMOTE works by creating synthetic samples from the minor class (no-subscription) instead of creating copies. It also randomly chooses one of the k-nearest-neighbors and uses it to create similar, but randomly tweaked, new observations. After the SMOTE operation, our training data had been well balanced. We over-sampled only on the training data, because by oversampling only on the training data, none of the information in the test data is being used to create synthetic observations, therefore, no information will bleed from test data into the model training. After that, we will perform recursive feature elimination(RFE). RFE is based on the idea to repeatedly construct a model and choose either the best or worst performing feature, setting aside and then repeating the process with the remaining features. This process is applied until all features are exhausted. The features with p-value > 0.05 will be filtered out.

Based on the elaboration of data features, our input data included “career” which has 19 options for users to choose. Hence, we try to create dummy data variables for each career in our datasets which have two values, zero and one. After execution of recursive feature elimination(rfe), we deleted the variables which p-value > 0.05. Hence, the remaining variables are shown below.



Only fews features are left and most of them don't meet our requirement in the prediction model. Important features such as rating of attributes during data had been filtered out. Hence, we decided not to perform dummy data variables for the career features, on the other hand represent the field name with integer. Hence, the dataset imported for the model training will be all integer based variables.

Next, we also meet the problems that when we perform regularization on the features, the accuracy of our model decreases to 76% and without the regularization is 82%. Regularization is one of the important prerequisites for improving the reliability, speed, and accuracy of convergence, but it is not a solution to every problem. Irregularity in data is only one of many root causes for slow or otherwise inadequate learning results, and as the results in the question indicates, it can reduce reliability, speed, or accuracy in some cases. Hence, to get better accuracy, we decide to not perform regularization on our training data.

**Suggestion for future works**

1. **Get more data**

The model we obtained has very good training accuracy but relatively low testing accuracy. This means it does not find a good general hypothesis and has a tendency to overfit the training data. One of the common fixes is to generate more data either by getting more response from dating couples or adding more relevant attributes to the existing data to generate new data to learn from.

1. **More diverse model to predict more results**

The model we used in this project was logistic regression which only predicts match or not match between two individuals after the first dating experience. In the future, more results can be predicted as we use more advanced models. For example, Neural network or deep learning to not only give answers yes or no, but more different predictions results can be given.

1. **Increase accuracy more than 90%**

Our learning model gave an accuracy of 83% so far. In the future, we hope it can reach more than 90% to ensure every prediction reaches his optimal results. We will try out more learning models and more data features to get better results. On the other hand, bias and variance problems also can be solved to ensure that there is no overfitting or underfitting problem in the learning model.

**APPENDIX**

Link of code:

<https://colab.research.google.com/drive/1JC_T3G8UhML8SeOP0vnhowUXS2lKjmyW?usp=sharing>

Link of Presentation video:

<https://drive.google.com/file/d/1bWT7meUNpkjWS6nwAIQa6KvHS19E0Qql/view?usp=sharing>

Link of Presentation slide:

<https://www.canva.com/design/DAEf9sjSDTk/MU09K74UCXzaXcjMF0kNNw/view?utm_content=DAEf9sjSDTk&utm_campaign=designshare&utm_medium=link&utm_source=sharebutton>