Machine Learning Engineer Nanodegree - Capstone Project

Li Chen Wu
January, 2019

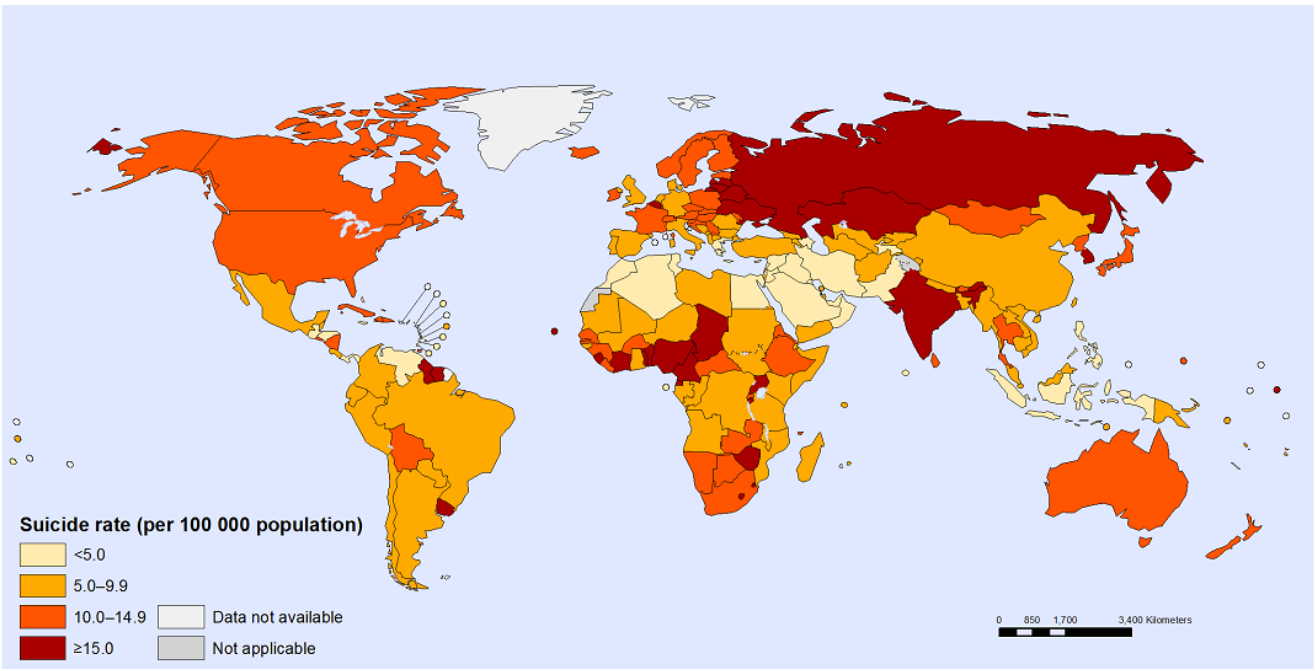
I. Definition

Project Overview

Nowadays, a lot of people from different countries die in suicide every year due to different reasons. According the statistics in Worth Health Orgnization (WHO) (https://www.who.int/mental_health/prevention/suicide/suicideprevent/en/), there are 8000 people die due to suicide each year, that is one person suicide in 40 seconds.

See the two figures below, it seems that the suicide rate is highly related with the areas, ages, and sex. The figures also show that high suicide rate usually occurs in some specific areas and ranges of age.

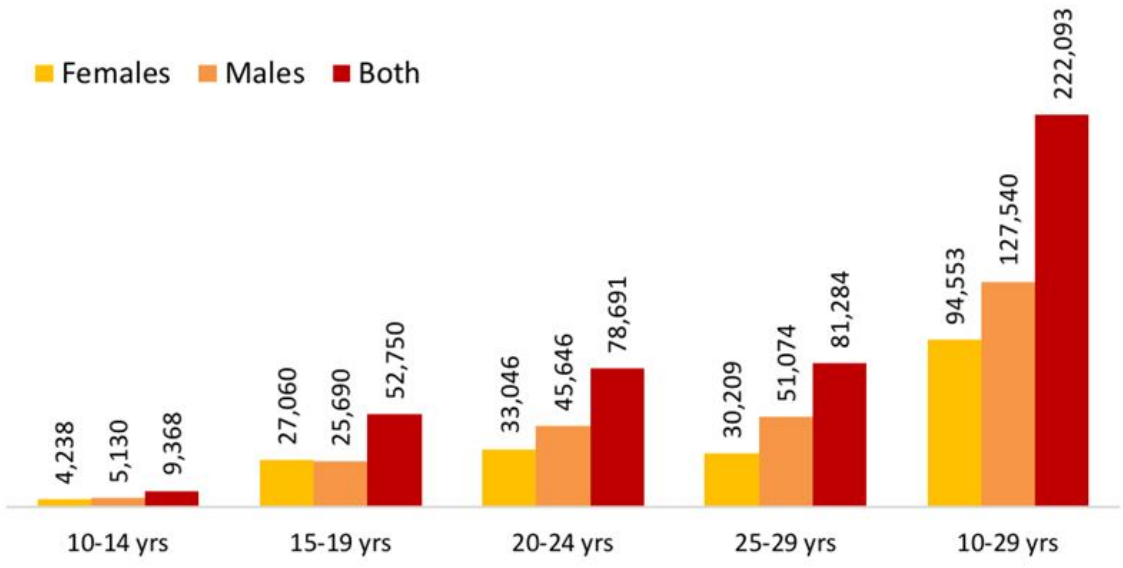
Age-standardized suicide rates (per 100 000 population), both sexes, 2016



Data Source: World Health Organization
Map Production: Information Evidence and Research (IER)
World Health Organization

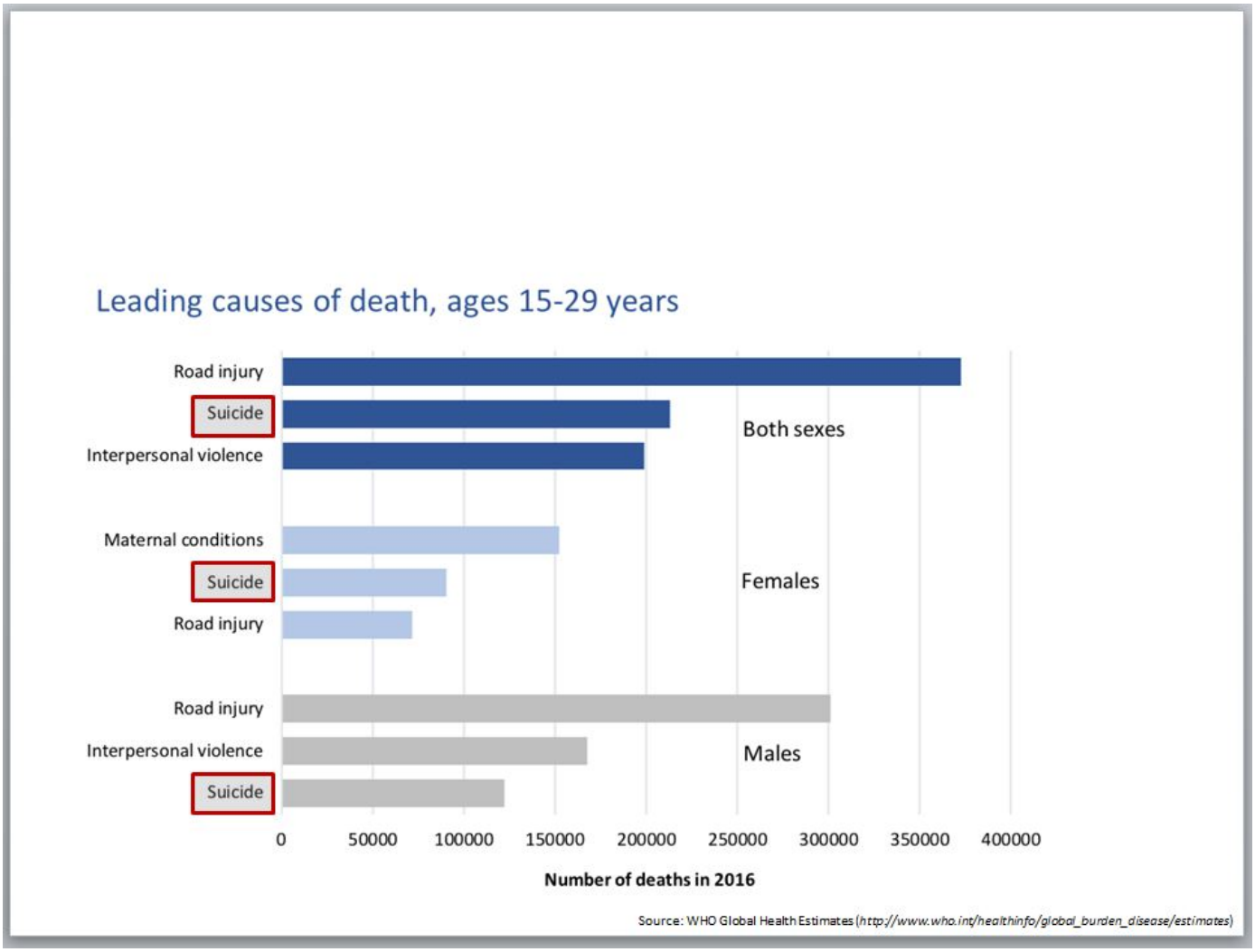
World Health Organization

Number of suicides globally in young people, 2016



Source: WHO Global Health Estimates (*http://www.who.int/healthinfo/global_burden_disease/estimates*)

I am also astonished of the huge number of suicide people and the suicide is also one of the leading cause of death in teenager.



There are many researchers do resarch on suicide analysis. Pete Burnap et al. [2015] used machine learning to classify the text on Twitter to find which were relating to suicide. Jihoon Oh et al. [2017] also used artificial neural network classifier to detect the patients who had actual suicide attempts by the training data of patients' self-report psychiatric scales and questionnaires about their history of suicide attempts.

In this project, machine learning technique is adopted and the suicide related dataset from Kaggale is used to build the training model to predict whether a person tends to suicide or not. It is an important that if we know that beforehand, we can give them more help and avoid the suicide happening.

Problem Statement

The goal of this project is to build a binary classification model which can detect whether a person tends to suicide or not by ouput Y or N.

To solve this problem, first, I obtain a suicide related dataset, The Demographic /r/ForeverAlone Dataset (https://www.kaggle.com/kingburrito666/the-demographic-rforeveralone-dataset/home), on Kaggle . Then I preprocess the data through log transform, numeric scaling and one-hot encoding so that it can be as the input of the training model. Finally, I choose Adaboost to train the model and fine tune its parameters.

To predict whether a person tends to suicide or not, there are some features of the person listed below should be inputted and the detail will be described in section [**II. Analysis > Data Exploration**]

- **gender**: What is your Gender?
- **sexuallity**: What is your sexual orientation?
- **age**: How old are you?
- **income**: What is your level of income?
- **race**: What is your race?
- **body weight**: How would you describe your body/weight?
- **friends**: How many friends do you have?
- **social_fear**: Do you have social anxiety/phobia?
- **depressed**: Are you depressed?
- **employment**: Employment Status: Are you currently...?
- **edu_level**:What is your level of education?

Metrics

In the dataset, there are 384 data labeled 'N' and only 85 data labeled 'Y'. Due to the unbalanced of the dataset classes, it is not suitable to use accuaracy as the evaluated metric. To evaluate the model properly, I choose Area Under the Curve (AUC), which is good at evaluating unbalanced classes, to measuer my model.

AUC computes the area under the Receiver Operating Characteristic curve (ROC) which "is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied" – from wiki (https://en.wikipedia.org/wiki/Receiver_operating_characteristic). ROC defines the x-axis and y-axis on two dimensional coordinate system base on the True Positive Rate (TPR), known as sensitivity or recall in machine learning, and False Positive Rate (FPR)

```
X: TPR = TP/(TP+FN)
Y: FPR = FP/(FP+TN)
```

We can evaluate our model by compute AUC score. If a model can split a dataset well, the AUC score would be greater than 0.8 and close to 1.0, otherwise the AUC score would be 0.5 or less.

II. Analysis

Data Exploration

The Form of the Dataset

The dataset is download from Kaggle, and it is a .csv file.

- File name: foreveralone.csv
- File size: 107 KB

The Basic Statistics of the Datasuet

- Row number: 469
- The data number with attempt_suicide value ‘Y’: 85, that is 18% of all data.
- The data number with attempt_suicide value ‘N’: 384, that is 82% of all data.

The Data Type of the Dataset

- **time**: (DateTime) Timestamp
- **gender**: (String) What is your Gender?
  - Male
  - Femaale
  - Transgender female
  - Transgender female
- **sexuallity**: (String) What is your sexual orientation?
  - Straight
  - Bisexual
  - Gay/Lesbian
- **age**: (String) How old are you?
- **income**: (String) What is your level of income?
  - $0
  - $1 to $10,000
  - $10,000 to $19,999
  - $20,000 to $29,999
  - $30,000 to $39,999
  - $40,000 to $49,999
  - $50,000 to $74,999
  - $75,000 to $99,999
  - $100,000 to $124,999
  - $125,000 to $149,999
  - $150,000 to $174,999
  - $174,999 to $199,999
  - $200,000 or more
- **race**: (String) What is your race?
  - White non-Hispanic
  - Asian

    - Black
    - Middle Eastern
    - ... etc
- **body weight**: (String) How would you describe your body/weight?
    - Normal weight
    - Underweight
    - Overweight
    - Obese
- **virgin**: (String) Are you a virgin?
    - Yes
    - No
- **prostitution legal**: (String) Is prostitution legal where you live?
    - Yes
    - No
- **pay_for_sex**: (String) Would you pay for sex?
    - No
    - Yes and I have
    - Yes but I haven't
- **friends**:(Number) How many friends do you have?
- **social_fear**: (String) Do you have social anxiety/phobia?
    - Yes
    - No
- **depressed**: (String) Are you depressed?
    - Yes
    - No
- **what_help_from_others**: (String) What kind of help do you want from others?
    - wingman/wingwoman
    - Set me up with a date
    - date coaching
    - ... etc
- **employment**: (String) Employment Status: Are you currently...?
    - Employed for wages
    - Out of work and looking for work
    - Out of work but not currently looking for work
    - A student
    - Unable to work
    - Retired
    - Military
    - Self-employed

    - A homemaker
- **job_title**: (String) What is your job title?
    - Student
    - Mechanical drafter
    - Game programmer
    - Software Engineer
    - ... etc
- **edu_level**: (String) What is your level of education?
    - Associate degree
    - Some college, no degree
    - High school graduate, diploma or the equivalent (for example: GED)
    - Bachelors degree
    - Trade/technical/vocational training
    - Masters degree
    - Some high school, no diploma
    - Doctorate degree
    - Professional degree
- **improve_yourself_how**: (String) What have you done to try and improve yourself?
    - None
    - Joined a gym/go to the gym
    - Joined clubs/socual clubs/ meet upds
    - Therapy
    - Other exercise
    - ... etc
- **attempt_suicide**: (String) Have you attempted suicide?
    - Yes
    - No

The Display and Finding of the Dataset

	time	gender	sexuality	age	income	race	bodyweight	virgin	prostitution_legal	pay_for_sex	friends	social_fear	depressed	what_help_from_others	employment	job_title	edu_level
0	5/17/2016 20:04:18	Male	Straight	35	30,000to 39,999	White non-Hispanic	Normal weight	Yes	No	No	0.0	Yes	Yes	wingman/wingwoman, Set me up with a date	Employed for wages	mechanical drafter	Associate degree
1	5/17/2016 20:04:30	Male	Bisexual	21	1to 10,000	White non-Hispanic	Underweight	Yes	No	No	0.0	Yes	Yes	wingman/wingwoman, Set me up with a date, date...	Out of work and looking for work	-	Some college, no degree
2	5/17/2016 20:04:58	Male	Straight	22	$0	White non-Hispanic	Overweight	Yes	No	No	10.0	Yes	Yes	I don't want help	Out of work but not currently looking for work	unemployed	Some college, no degree
3	5/17/2016 20:08:01	Male	Straight	19	1to 10,000	White non-Hispanic	Overweight	Yes	Yes	No	8.0	Yes	Yes	date coaching	A student	student	Some college, no degree
4	5/17/2016 20:08:04	Male	Straight	23	30,000to 39,999	White non-Hispanic	Overweight	No	No	Yes and I have	10.0	No	Yes	I don't want help	Employed for wages	Factory worker	High school graduate, diploma or the equivalen...
5	5/17/2016 20:09:09	Male	Straight	24	50,000to 74,999	White non-Hispanic	Normal weight	Yes	No	Yes but I haven't	2.0	Yes	Yes	date coaching	Employed for wages	game programmer	Bachelors degree
6	5/17/2016 20:10:56	Male	Straight	22	1to 10,000	White non-Hispanic	Underweight	Yes	No	No	2.0	Yes	Yes	Set me up with a date, date coaching	Employed for wages	Janitor	High school graduate, diploma or the equivalen...
7	5/17/2016 20:11:13	Female	Gay/Lesbian	24	20,000to 29,999	White non-Hispanic	Normal weight	Yes	No	No	10.0	Yes	Yes	wingman/wingwoman, date coaching	Employed for wages	Fabricator	Trade/technical/vocational training
8	5/17/2016 20:11:52	Male	Straight	20	10,000to 19,999	White non-Hispanic	Overweight	Yes	No	Yes but I haven't	0.0	Yes	Yes	Set me up with a date	Employed for wages	cashier	Some college, no degree
9	5/17/2016 20:13:37	Male	Straight	33	50,000to 74,999	White non-Hispanic	Overweight	No	No	Yes but I haven't	6.0	Yes	Yes	Set me up with a date	Employed for wages	Software Engineer	Masters degree

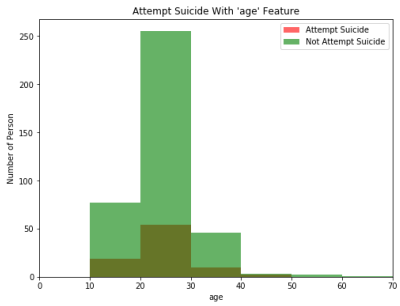
Based on the description of each features above, the '**time**' is only a timestamp. I think it is not important for this problem, so I would remove it from my feature list. Besides, the features '**job_title**', '**what_help_from_others**' and '**improve_yourself_how**' are too messed to find a rule and categorize them; therefore I would also remove them.

The remain features I would use to training are **gender**, **sexuallity**, **age**, **income**, **race**, **body weight**, **virgin**, **prostitution legal**, **pay_for_sex**, **friends**, **social_fear**, **depressed**, **employment** and **edu_level**. And the target labeled to predict is **attempt_suicide**.

Exploratory Visualization

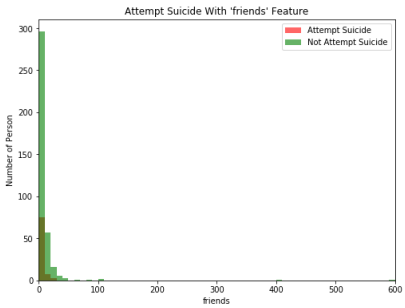
To explore the relation between each feature and suicide_attempt, I calculate the histogram for each features and then illustrate them through bar graph.

Age



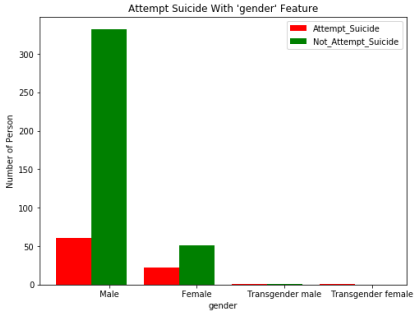
(https://i.imgur.com/0v81mIl.png)

Friends



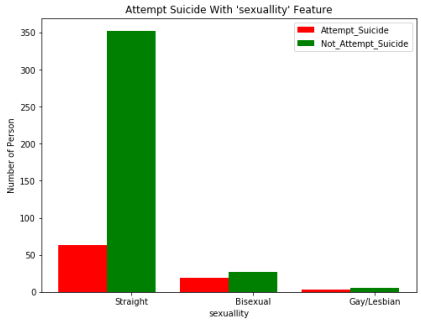
(https://i.imgur.com/bFht29c.png)

Gender



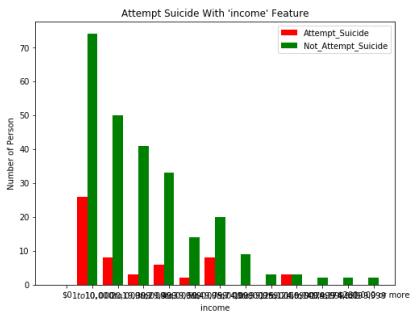
(https://i.imgur.com/5gDYMnI.png)

Sexuallity



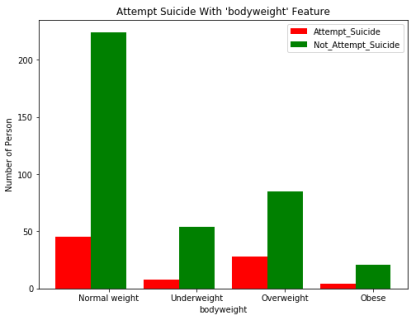
(https://i.imgur.com/skuNJWX.png)

Income



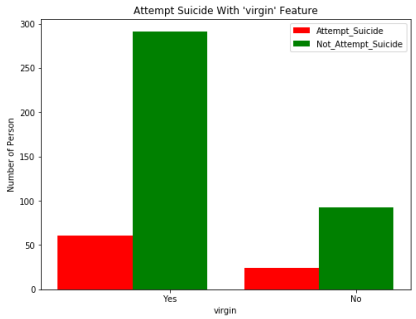
(https://i.imgur.com/I77eG3m.png)

Body/Weight



(https://i.imgur.com/bAw9XsJ.png)

Virgin

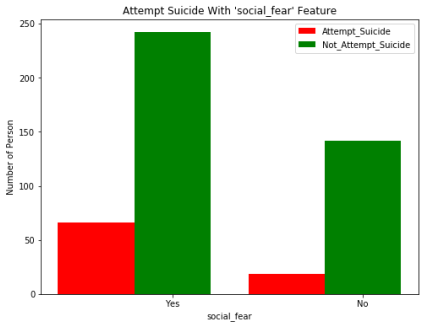


(https://i.imgur.com/3Xh4Be5.png)

Prostitution Legal

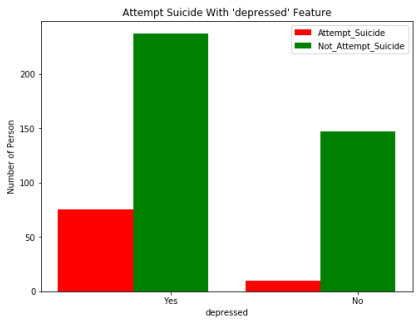
(https://i.imgur.com/laWKcLs.png)

Social Fear



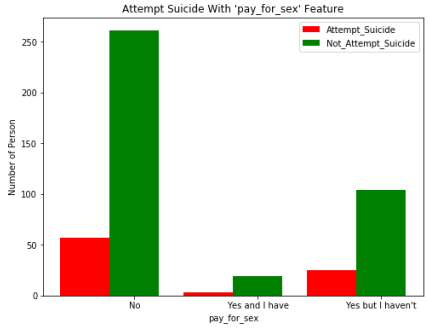
(https://i.imgur.com/YGnZFGU.png)

Depressed



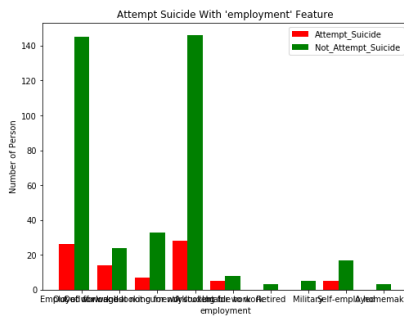
(https://i.imgur.com/z2gMquu.png)

Pay for Sex

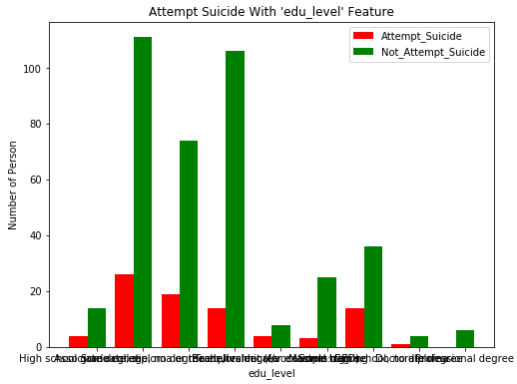


(https://i.imgur.com/18AMFCW.png)

Employment



(https://i.imgur.com/Wk2zT5r.png

Education Level		
 (https://i.imgur.com/vc9PCL9.png)		

- **Age**
  Found that the age in range [40, 50] and [60, 70] has high suicide rate, but it is not significant.
- **Friends**
  Found that the friends in range [0, 10] has high suicide rate.
- **Gender**
  Found that the transgender male and tansgender female has highest suicide rate, and then the female. The last is male.
- **Sexuallity**
  Found that Bisexual and Gay/Lesbian has significant high suicide rate.
- **Income**
  The income level '$1 to $10,000', '$50,000 to $74,999' and '$125,000 to $149,999' has high suicide rate.
- **Body/Weight**
  The overwight has high suicide rate.
- **Virgin**
  It is not so significant for 'Yes' and 'No' in feature virgin.
- **Prostitution Legal**
  It is not so significant for 'Yes' and 'No' in feature prostitution_legal.
- **Social Fear**
  The person who has social fear has high suicide rate.
- **Depressed**
  The person who is depressed has high suicide rate.
- **Pay for Sex**
  It is not so significant for any values in feature pay_for_sex.

- **Employment**
  The value ‘Out of work and looking for work’ and ‘Unable to work’ has high suicide rate.
- **Education Level**
  The value ‘Some high school, no diploma’ has high suicide rate.

Algorithms and Techniques

For the data preprocessing, the logarithmic transformation is applied to features ‘age’ and ‘friends’ to deal with the skewed data. The features ‘age’, ‘income’ and ‘friends’ are also normalized between 0 and 1.0.

For training the model, there are four algorithems which contains Decision Trees, AdaBoost, SVM and Random Forest used in the project, and then the best one ‘AdaBoost’ is chose as the final model which performs good AUC score.

The Adaboost in sklearn libaray, the default parameters are listed below, and I would fine tuene these parameters to reach highest AUC score.

- n_estimators = 50
- learning_rate = 1.0
- algorithm = ’SAMME.R’

Benchmark

The benchmark model I defined is Logistic Regression. For the Logistic Regression, I set the parameters as below

```
LogisticRegression(random_state=42, solver='lbfgs',max_iter=100)
```

The result of the Logistic Regression is

```
Accuracy score on testing data: 0.8191
AUC score on the testing data: 0.5229
```

Althogh the accuracy reach 81%, it is not a good model due to its unbalanced classes and the AUC score has only 0.52.

III. Methodology

Data Preprocessing

There are six steps for the data preprocissing.

1. **Remove features:**
   The features listed below are removed for the training model.

   - time
   - what_help_from_others
   - improve_yourself_how
   - job_title
   - virgin
   - prostitution_legal
   - pay_for_sex

   Because the feature 'time' is only a timestamp and features 'job_title', 'what_help_from_others', 'improve_yourself_how' are too messed to categorize for them. And according to **[II. Analysis > Exploratory Visualization]**, it seems that features 'virgin', 'prostitution_legal', 'pay_for_sex' have no significant affect on the suicide results. Therefore, these features are removed for the training model.

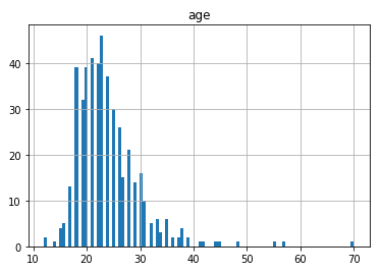
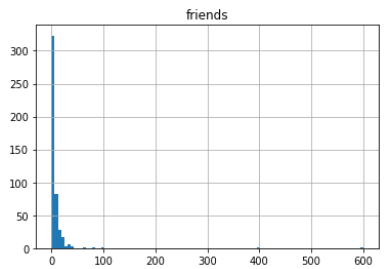
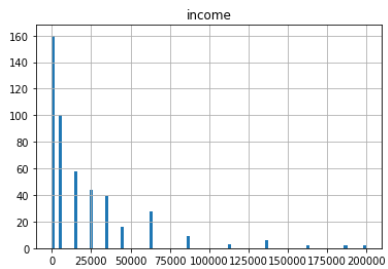
2. **Handle with column 'income'**
   Due to the data type of the feature 'income' is a string, for example '$1 to $10,000'. The 'income' would be reassigned as its average value `($1+$1000)/2` with data type 'float'.

3. **Set 'Yes' to 1, 'No' to 0 and convert to upper case for the string**
   Convert the values 'Yes' and 'No' to binary 1 and 0 for the feature 'social_fear', 'depressed' and the label 'attempt_suicide'. To avoid that they are same string but in different cases, I also set the string data type features to its upper case.

4. **Transform skewed continuous features**
   A dataset may contain features of which values tend to lie near a single number, which is very large or small. The training algorithem can be sensitive to this kind of distribution data and underperform if the range is not properly normalized.

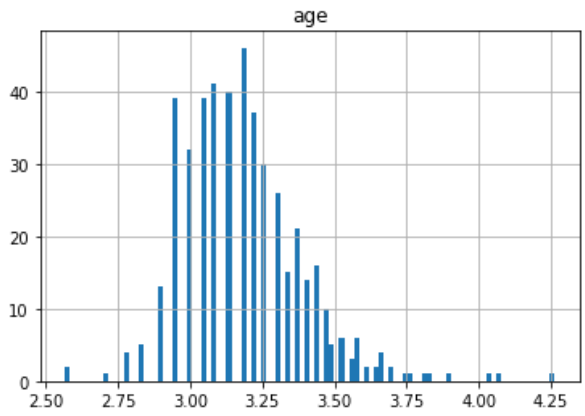
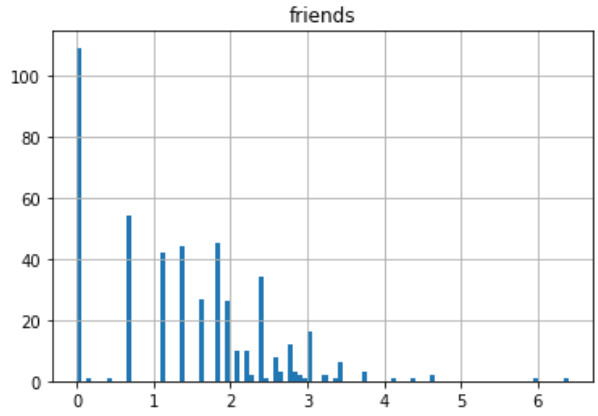
   To find what values of the continuous features are skewed, the histograms of them are calculated and display below.

- Raw data histogram of 'age', 'friends' and 'income'

age	friends	income
		

After visualizing the continuous features, I find that the values of ‘age’ and ‘friends’ are skew. Hence the **logarithmic transformation** is applied on them and the result is the bar graph below.

- The histogram after performing logarithmic transformation

age	friends
	

5. **Normalize numerical features**
   To treat each numerical features equally, the scaling had applied on the features ‘age’, ‘income’ and ‘friends’.

Before Normalize

age	income	friends
35	34999.5	0.0
21	5000.5	0.0
22	0.0	10.0
19	5000.5	8.0
23	34999.5	10.0

After Normalize

age	income	friends
0.599960	0.174998	0.000000
0.309880	0.025003	0.000000
0.336063	0.000000	0.374753
0.253740	0.025003	0.343392
0.361132	0.174998	0.374753
0.385177	0.312498	0.171696
0.336063	0.025003	0.171696
0.385177	0.124998	0.374753
0.282479	0.074998	0.000000
0.566292	0.312498	0.304115

6. **Perform one-hot encoding on the non-numeric features**
   Because the learning algorithems expect the input to be numeric, the none numeric features is converted to numeric via one-hot encoding. The features which are applied one-hot encoding are listed below.
   - gender
   - sexuallity
   - race
   - bodyweight
   - employment
   - edu_level

Implementation

Spliting Data

Due to the unbalanced classes for the dataset, I split the dataset into training data and testing data by three steps in order to make sure that training data and testing data have same ratio with label 1 and label 0.

First, I gather all the rows with attempt_suicide = 0 and then splited them with 80% as training data `(X_train_0, y_train_0)` and 20% as testing data `(X_test_0, y_train_0)`.

Then, I also gather all the rows with attempt_suicide = 1 and splited them with 80% as trining data `(X_train_1, y_train_1)` and 20% as testing data `(X_test_1, y_train_1)`.

Finally, the data with label 0 and that with label 1 are merged in to one training data and testing data `(X_train, y_train)`.

Training Model

To choose a best model for this problem, I implement four algorithms:

- decision trees
- adaboost
- svm
- random forest

The grid below illustrates the results of the four algorithems with default parameters set by sklearn library and manually tuned parameters .

	Decision Trees	AdaBoost	SVM	Random Fore
Parameters	default	default	default	default
Accuracy	0.7766	0.8298	0.8191	0.8511
AUC	0.5657	0.5982	0.5000	0.5229
----------	----------	----------	----------	----------
Parameters	max_depth=10, min_samples_split=5, max_features=0.9	n_estimators=80	kernel='poly', gamma=4.5, coef0=2.0, max_iter=750	n_estimators= max_features=
Accuracy	0.8404	0.8298	0.7660	0.8511
AUC	0.5817	0.6211	0.5821	0.5882

After roughly tuning the parameter, the algorithm **Adaboost** outperforms the others. Although the accuracy of the Adaboost is not the highest, the AUC score, which is good evaluated metric for unbalanced classes, is higer than others. Therefore, I chose Adaboost as the my training algorithm.

Refinement

The **GridSearch** is adopted to fine tune the parameters of the adaboost, including 'n_estimators', 'learning_rate', 'algorithm'.

```
parameters = {
    'n_estimators':[10, 50, 80, 100, 150, 200, 500],
    'learning_rate':[0.1, 0.5, 1.0, 2.0, 5.0],
    'algorithm':['SAMME','SAMME.R']}
```

The best best estimator output from grid search is

```
AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None,
          learning_rate=1.0, n_estimators=100, random_state=42)
```

The output scores are

```
Accuracy score on testing data: 0.8191
AUC score on the testing data: 0.5917
```

It seems that the grid search dosen't peform very well. After adjusting the parameters by it and using the best estimator to training the model. The accuracy and AUC score only has 0.8191 and 0.5917 which are less than that of the parameters I tuned manually (0.8298 / 0.6211). Therefore, the final model I would choose the Adaboost with the parameters I tuned maually.

IV. Results

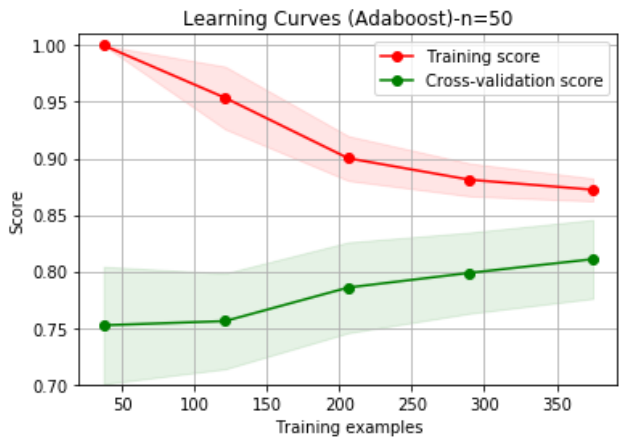
Model Evaluation and Validation

See in the analysis in section **[III. Methodology > Implementation]**, the AUC score of the **Adaboost** is better than that of others for the dataset, so I choose Adaboost as my final model. The parameters of the final model is

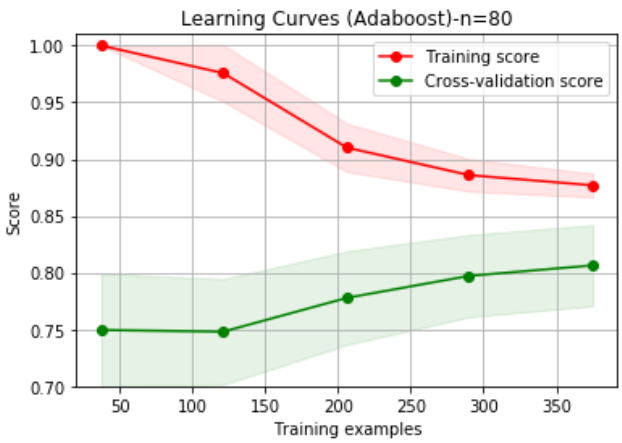
- n_estimators = 80
- learning_rate = 1.0
- algorithm = 'SAMME.R'

In order to find the best value of n_estimators, I display the learning curve for n_estimators = 50, 80, 150, 250, 300. It can be seen that there is not significant between n_estimators = 50 and 80, but when n_estimators is greater thatn 150, the cross-validation score decreases which means that the model is starting to overfitting. Finally, I decide to use n_estimators=80 which has higher AUC scores than n_estimators=50.

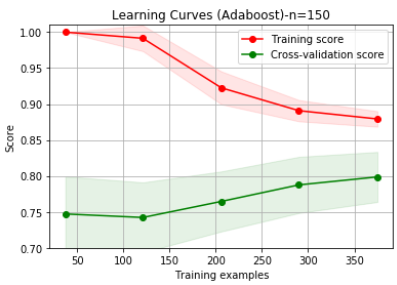
n_estimators=50



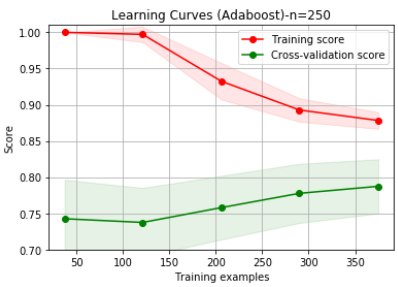
n_estimators=80



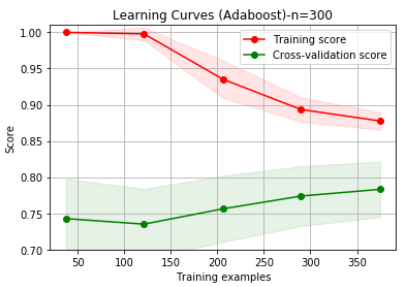
n_estimators=150



n_estimators=250



n_estimators=300



Justification

As shown below, the result of the final model is stronger than that of the benchmark model. Although the accuracy score of the final model is only 0.01 higher than the benchmark, the AUC score is 0.1 higher than the benchmark.

Though the result is better than the benchamrk, I think the final model is not strong enough to predict each person who has suicide attempt. The final model still need to be improved in order to reach higher accuracy and AUC score.

- Benchmark model:Logistic Regression

```
Accuracy score on testing data: 0.8191
AUC score on the testing data: 0.5229
```

- My final model: Adaboost

```
Accuracy score on testing data: 0.8298
AUC score on the testing data: 0.6211
```

V. Conclusion

Reflection

When survey the topic for my MLND project, I found the statistic on the WHO website and I was so stuned by the number of suicide people each year that is one person suicide in 40 seconds. So I decided to do this project: Who Tends to Suicide, and I want to predict the person who is prone to suicide so that we can help them.

I downdloaded the dataset from Kaggle. After obtaining the dataset, I calculated the histogram of each features in order to find the relation between the feature and the label. Then I removed some features which has no significant impact on the suicide result. A series of preprocessing was performed on the data including logarithmic transformation, numeric scaling and one-hot encoding. Next the four alogrithems were implemented including Decision Trees, AdaBoost, SVM, Random Forest, and evaluated the model via AUC score. Finally, I chose the model AdaBoost as the final result which has highest AUC score.

During this project, one thing made me confused. I used GridSearch algorithm to tune the parameters of Adaboost, but the result made me frustrated because the result was worse than the parameters I tuned manually.

Besides, there are two difficulties in this project. The first difficulty is the small size of the dataset, which has only 469 data in total and only has only 85 data with attempt_suicide = ‘Yes’. I beleive that it is the key reason of my low AUC score (0.6211).

The other difficulty is how to deal with the features what_help_from_others, job_title and improve_yourself_how. It seems that the value of these features was the description for the question wrote down by the interviewee, instead of selecting one answer from existing answer list. It is difficult for me to categorize these features without the domin knowlege about them.

Improvement

To improve the result of the model, I will take the features what_help_from_others, job_title and improve_yourself_how in consideration. I think there is important information in the feature what_help_from_others and it is a little bit pity to drop it.

To use the feature what_help_from_others, I can do semantic analysis and give it a score in range [0, 1]. The score closes to 1 indicates that high probablity to suicide, and otherwise, the score closes to 0 indicates that low probablity to suicide.

I believe that the result would be better if the improvement method is implemented on the final model.

VI. References

1. Worth Health Orgnization: https://www.who.int/mental_health/prevention/suicide/suicideprevent/en/ (https://www.who.int/mental_health/prevention/suicide/suicideprevent/en/)
2. Kaggle: https://www.kaggle.com/kingburrito666/the-demographic-rforeveralone-dataset/home (https://www.kaggle.com/kingburrito666/the-demographic-rforeveralone-dataset/home)
3. Machine Classification and Analysis of Suicide-Related Communication on Twitter https://orca.cf.ac.uk/76188/1/p75-burnap.pdf (https://orca.cf.ac.uk/76188/1/p75-burnap.pdf)
4. Classification of Suicide Attempts through a Machine Learning Algorithm Based on Multiple Systemic Psychiatric Scales https://www.frontiersin.org/articles/10.3389/fpsyt.2017.00192/full (https://www.frontiersin.org/articles/10.3389/fpsyt.2017.00192/full)