Predict Suicides to save lives

Capstone Project Report

Ish Gupta

4/19/21

ML Engineer Nanodegree

Definition

Project Overview

National Institutes of Health, and Center for Disease Control and Prevention and the U.S. Census Bureau conducted a National Longitudinal Mortality Study (NLMS) for the purpose of studying effects of differentials in demographic and socio-economic characteristics on mortality.

Mortality information was obtained from death certificates available for deceased persons through the National Center for Health Statistics. Standard demographic and socio-economic variables such as education, income, and employment, as well as information collected from death certificates including cause of death are available for analyses.

Problem Statement

The survey consists of the reasons of death for the people who lost their life, and were included in the Survey. The reasons are related to Health, accidents, and intended loss of life, i.e., Suicide.

The data can be analysed and studied to see if there can be any measures that the government can take to reduce Suicidal attempts, by classifying the deaths caused by intended action, vs accidents.

Giving up on Life, is the most brutal act we can do to ourselves and our loved ones. Let's try, if we can employ data technologies to save one from messing up with not just this Life, but the afterlives as well.

Metrics

The metrics I have used for evaluation are:

- ➢ Accuracy – performance of a model against the test data set
- ➢ Confusion matrix- to see how well did the model do in identifying actual cases of Suicide, and false negatives

Analysis

Data Exploration

The dataset is 1048576 rows x 43 columns, with attributes in respect to citizenship, demographics, family, education, profession, income, health, habits, mortality.

Out of the major contributing attributes, the numeric attributes are:

1. Age - age of the person at the time of the survey
2. Follow- The length of follow-up period in days

The categorical features include:

3. Sex- Gender
4. POB - Region of Birth
5. Race - An expanded version of race which separates the American Indian, etc. and Asian, etc. out of the Other category in early CPS files
6. Hisp - Hispanic origin classifies all persons by Mexican, Hispanic (not Mexican) or Non-Hispanic (Not Mexican or Hispanic) origin
7. Ms- marital status

8. RELTRF - Relationship to reference person within the household
9. HHNUM - number of persons residing in the household at the time of the interview
10. SSNYN - Indicator of the presence or absence of Social Security Number on the CPS record
11. STATER - state of residence at the date of interview
12. URBAN - Urban or rural status
13. SMSAST - Is a household located in an SMSA or not? (SMSA - standard metropolitan statistical area)
14. TENURE - the type of ownership of the residence
15. EDUC- highest grade completed
16. VT - Indicates whether person was U.S. veteran
17. CITIZEN - person's citizenship at the time of survey
18. health - the status of person health at the time of survey
19. Working - if the person was employed
20. ESR - Recoded employment status
21. IND - Job specific industrial classification codes
22. MAJIND - Major industry classification recode of 2007 specific industry
23. RCOW - Recoded Class of Worker
24. ADJINC - Inflation Adjusted Income
25. POVPCT - Income as Percent of Poverty Level
26. HISTATUS - Indicates whether person had health insurance coverage any time in the calendar year prior to the interview
27. HITYPE - the type of health insurance

output variable:

28. Cause113- cause of death – Suicide (1), or accident (2), or other Health related issues (0)

Data cleaning

Removing Less significant features:

I started off looking at the data variables which would not make a signification contribution towards the outcome, which can be:

1. Variables which are just for record keeping and have distinct value per row
    a. *RECORD* - sequential number for each record on the file
2. Variables with a single unique value throughout the data
    a. Variables related to smoking and tobacco usage, which were mostly blanks in the data we had
        i. found: *SMOK100, AGESMK, SMOKSTAT, SMOKHOME, CURRUSE, EVERUSE*
3. attributes with inclination towards an outcome event
    a. check for any variables inclined towards any particular outcome event
        i. found: none
4. Variables missing more than 40% of the values
    a. *OCC:* 4 Digit Occupation Code
    b. *IND*: 270 unique Job specific industrial classification codes
    c. *RCOW*: 5 class of worker codes to explain if the individual works/worked in private, government, self-employed, work-without-pay, or has never worked
    d. *CITIZEN*: person's citizenship at the time of survey
    e. *HEALTH*: status of person's health

    f. *INDDEA*: indicator, if the person is alive or not
    g. *INDALG*: mortality indicator, basis the algorithmic calculation

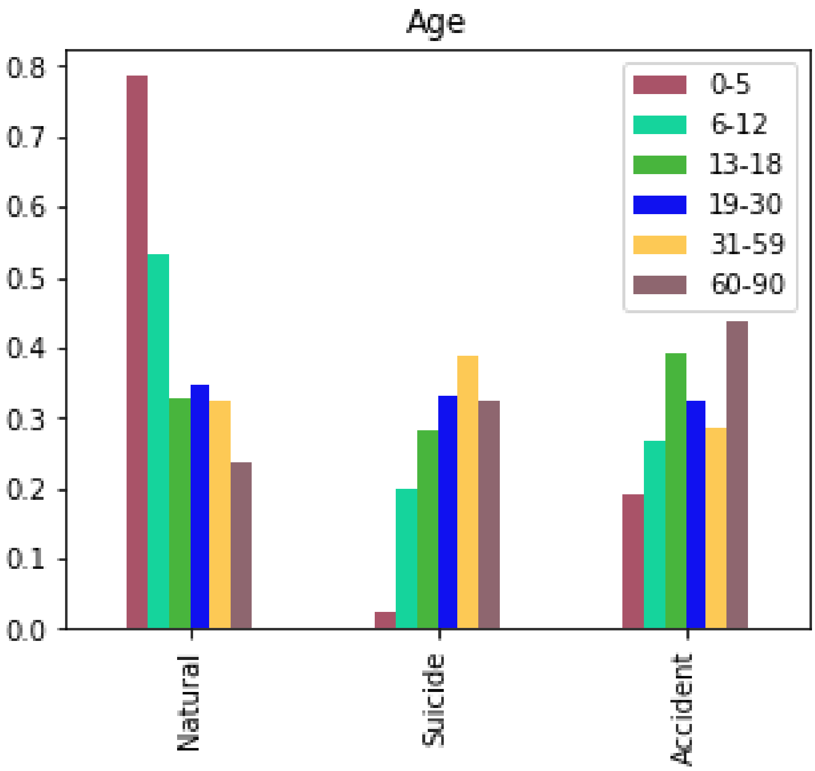
5. Variables describing the outcome (when it happened, or after it has happened)
    a. *DAYOD:* day of the week on which the decedent died
    b. *HOSP*: place of death
    c. *HOSPD*: location of death relative to a hospital
    d. *CAUSE113:* alternated with 'Suicide' – for 3 distinct categories:
        i. *0* – Natural death
        ii. *1* – Suicide
        iii. *2* - Accident
6. variables completely missing in one or the other case of outcome event
    a. found: none

Exploratory Visualization

The variables which have <= 6 distinct categories in the data were studied for their effect on deaths.

Age

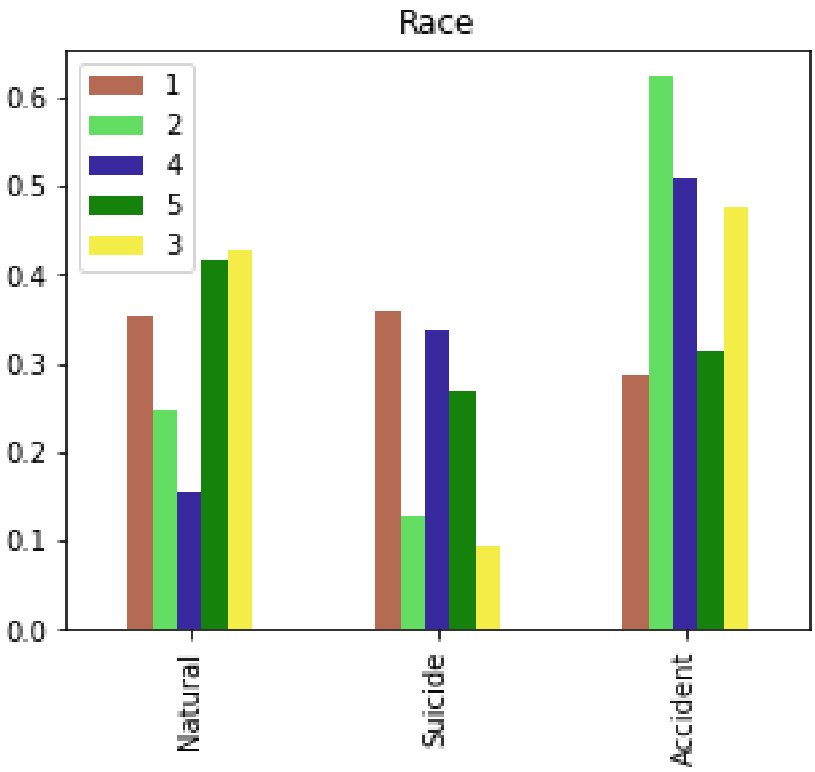
1. Infants have died most natural deaths
2. Senior citizens have had most accidents
3. People within the range of 31-59 are the ones with the greatest number of Suicides, followed by youngsters within age 19-30. This is known that this is the age of passion, when all of the people are busy in making their career, family. When failures come, they may lead to frustration, depression like issues.



Race

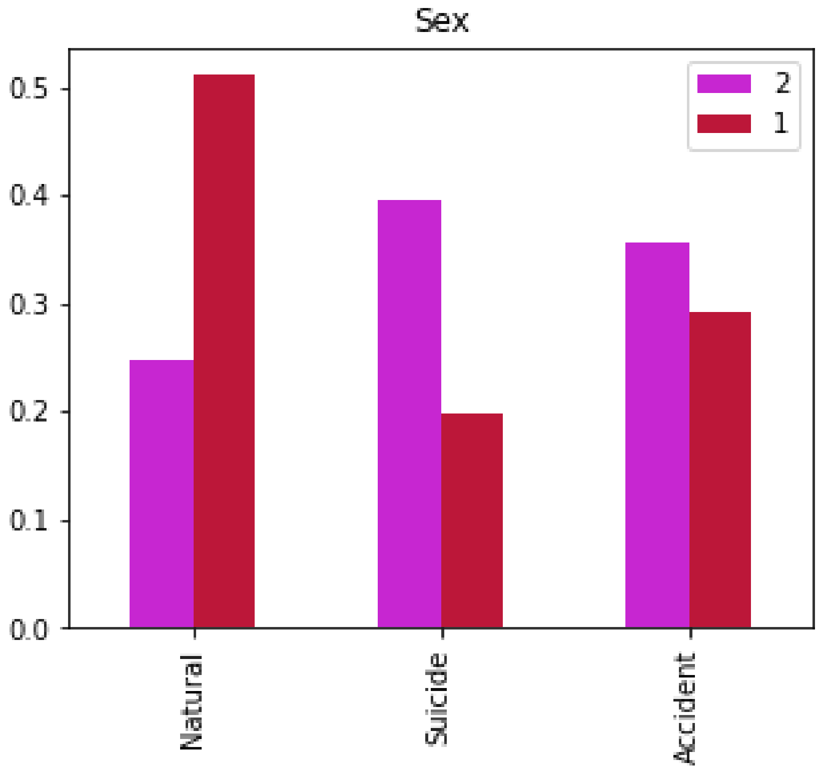
While checking Race, we could see that:

1. Race = 2 (Black) has had most accidents in comparison to other 4 categories of Race
2. Race = 1 (White) has had most suicides, and
3. Race = 5 (Other non-white) have had most deaths due to health issues



Gender:

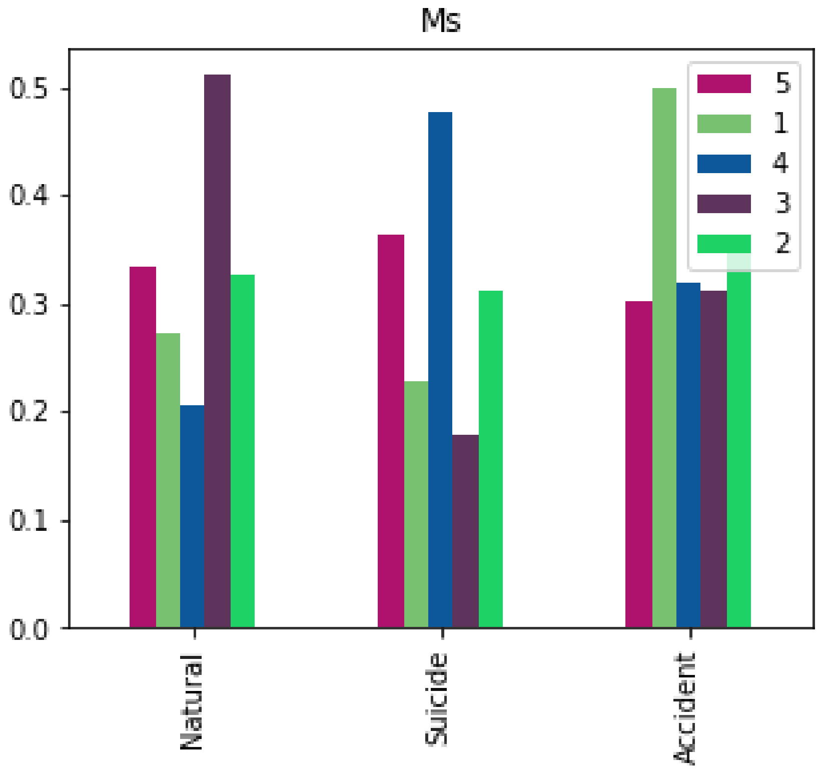
Men (category 1) have had the greatest number of natural deaths, but Females exceed over in Suicides and Accidents. This could mean, that females have better health than Men, but are more prone towards emotional hurt, and accidents.



Marital Status

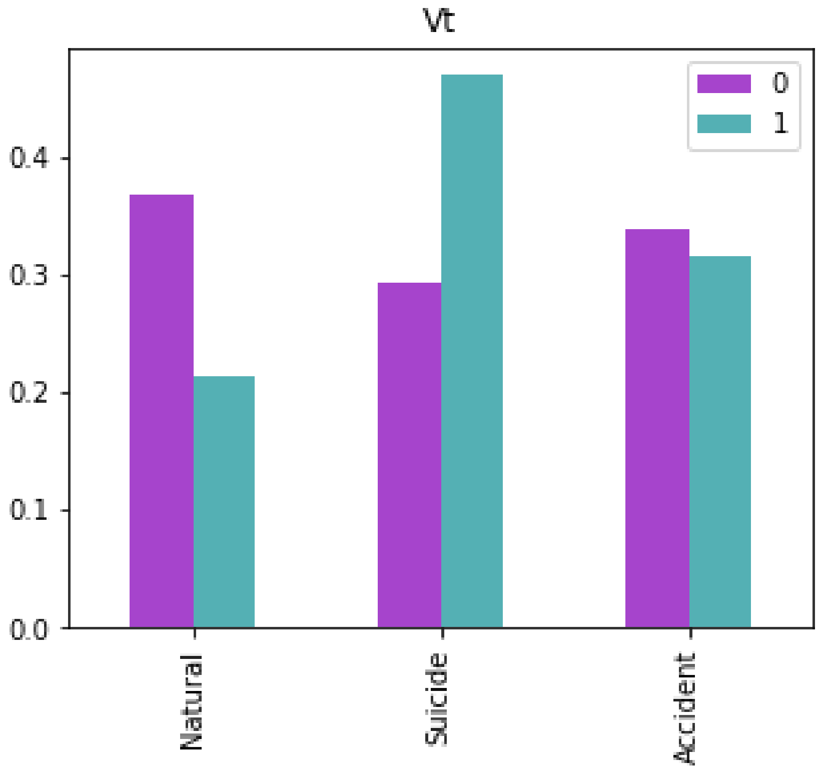
1. Married (category 1) people top the accidents
2. Separated (category 4) top the Suicides
3. Divorced (category 3) top natural deaths

This could mean, married people are comparatively careless on roads, separated accumulate emotional stress overtime, and end up taking their lives- which can be taken as an attribute to counsel the individuals (by the Administration) in order to help them out.



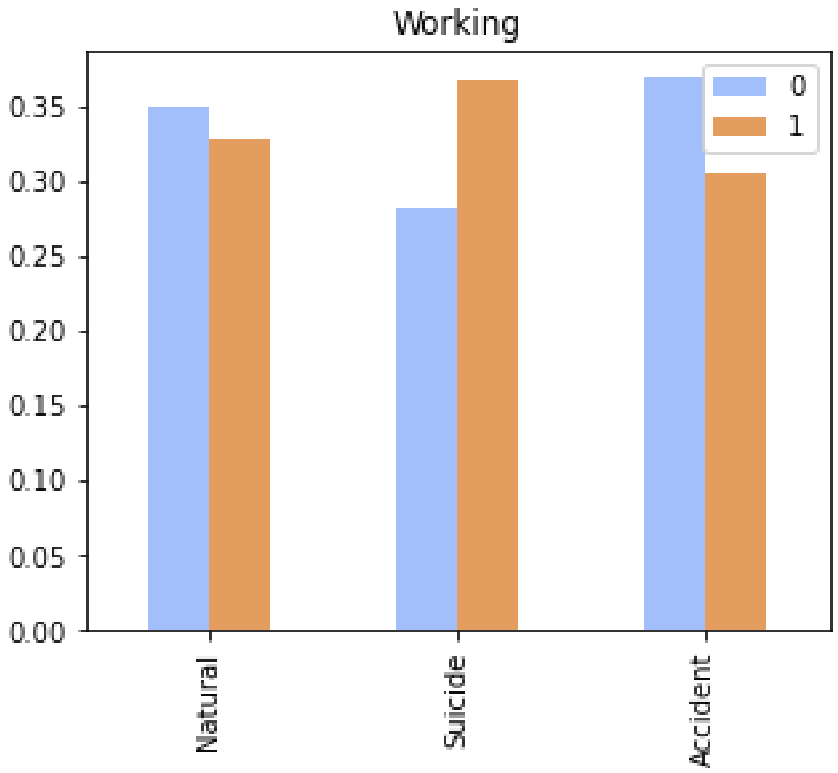
Veteran

This shows that Veterans (category 1) top in committing Suicides, which could possibly be because of PTSD issues. But, can be an important lead for administration to look after and help them out by counselling.



Working

People who are working happen to commit more suicides in comparison, with the people not involved in any employment. This could be related to not having good Work-Life balance, and the resulting stress.



Algorithms and Techniques

Benchmark

Methodology

Data Pre-processing

In the final version of EDA, I could finalize below steps to clean the data, and bring it into a shape to be ready for building Classifiers:

1. Categorize eligible variables into classes:
   a. distribute age into classes: ["0-5", "6-12", "13-18", "19-30", "31-59", "60-90”]
   b. categorize POB (place of birth) field into categorized regions
   c. categorize STATER (state of residence) codes into political divisions
2. convert follow up days into years
3. create new variable:
   a. “Suicide” (using *CAUSE113*) to denote the reason of death was natural, accidental, or intentional:
      i. 1- Suicide
      ii. 2 – Accident
      iii. 0 - Natural
   b. “Working” (using *OCC*) to know if an individual is/was working? (using Occ)

4. Impute *MAJIND, MAJOCC* with ‘-1’ for missing values, rather than removing them to study mortality rate is affected by occupation, industry an individual works
5. type caste all variables into category type
6. The data was widely imbalanced for deaths due to health issues, for the proportion was more than 95%, so the outcome event has to be randomly trimmed in proportion with the Suicides, and Accidents so as to balance the data, which gave ~ 6300 rows vs 1048576.
7. impute missing values using MissForest
8. save the dataset for use by other scripts which will build a Classifier using XGBoost, and PyTorch
9. create an initial Random Forest Classifier to see first level observations.
10. Drop *FOLLOW,* which denoted the days an individual was followed up for survey. Since, it cannot be an effective contributor towards a person harming his/her life.
11. Re- process a RandomForestClassifier, and study if there are any features which can be considered to drop out of the data.
12. Study permutation importance of variables to check feature importance, use RFE (sklearn) to validate the observations.
13. Filter the unimportant variables out, and re-process a classifier to see improvements.
14. Now, use find most optimal parameters for a Classifier using cross validation (GridSearchCV), and check if it helps improving the accuracy of observations, and if it reports any changes in the feature importance.
15. Use the data persisted in step 7 for further scripts

Implementation

The project contains 3 scripts/notebooks used in processing the data, and building a classifier.

0. *ml_toolkit.py* contains utility methods, and is being referenced in all the scripts
1. *eda.ipynb* – This script:
    a. explores the data from the beginning
    b. does necessary data cleaning, as listed in above points
    c. transforms the data by:
        i. removing unwanted variables
        ii. creating and replacing already available attributes with new concise ones
        iii. imputing missing values
        iv. one-hot encoding the data, to have an individual attribute per categorical value of all the variables
        v. build an initial classifier (RandomForestClassifier) to perform Feature selection
    d. builds RandomForestClassifier using:
        i. the data with categorical variables
        ii. with the same data, but after removing some unimportant variables as identified in the above iteration
        iii. the one-hot encoded data
        iv. and finally with the encoded data, but with optimal parameters obtained using cross validation
    e. records accuracy of the RandomForestClassifeir from each step to compare in the final step.

2. *process_xg_classifier.ipynb – This script:*
    1. uses the data prepared in *eda.ipynb*
    2. to use cross validation in finding optimal parameters for an XGBoost Classifier
    3. performs another round of cross validation to confirm the optimal parameters
    4. records accuracy of the above classifiers for comparison in the final step

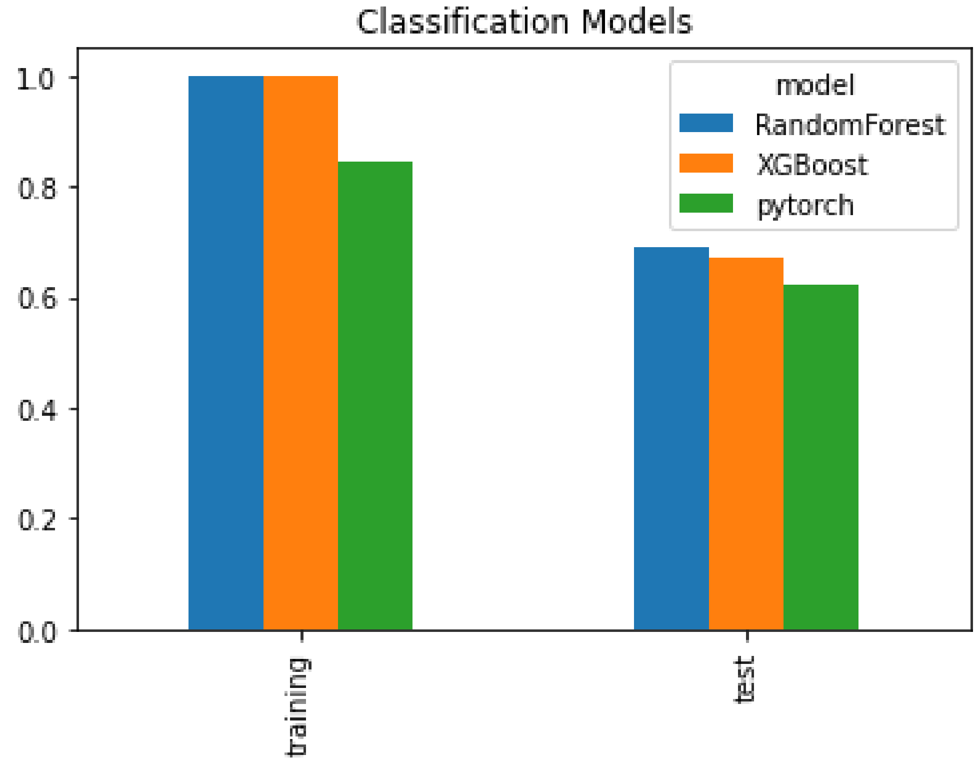
3. *process_nn_classifier.ipynb* - This script:
    1. uses the data prepared in eda.ipynb
    2. builds a 3 layer classifier using 3 different approaches for comparison
        1. Approach 1:
        2. Approach 2:
        3. Approach 3:

Refinement

- I started with XGBoost classifier to segregate Suicides and Accidents
- Then, I wanted to compare the performance of XGBoost with Pytorch
- While looking at the capstone acceptance metrics on Udacity, as I started working on the EDA script, I felt I could also incorporate a Random Forest Classifier, for it will be interesting to compare the performance.
- I had multiple hit and trials with different choices of model parameters in XGBoost, and pytorch, but I started raw in RandomForest, and ended up doing a model using CV in RF too.
- After all this, I realized, maybe the problem statement that I am trying to resolve by comparing Suicides with Accidents might be not that fruitful (performance-wise), but the other way round, where we can see mortality events in cases of Health-related issues, and Suicides, and finally happened to update all the execution for this.
- The code for pytorch was repetitive for the execution, and then I thought of optimizing it by putting into individual methods, and called them in an iteration over 3 model choices.

Results

Model Evaluation and Validation



Justification

- XGBoost performed the best in comparison, but Pytorch can do a much better job if we had a larger dataset for all the outcome events
- False Negatives can be a problem in such a use case, where we wrongly classify a Suicide, to be not a Suicide. But that was improved when we tweaked the XGBoost model
- Out of the three models in pytorch:
  - second had the least False Negatives, but it was still more than XGBoost classifier.
  - But after applying softmax activation function to the first choice, it exceeded the second model in having lowest False negatives