|  |  |
| --- | --- |
| A picture of a winding road and trees  Predicting Car Crash Severity for South australia  Capstone Project | Abstract  Prediction of Crash Severity is an old topic. This project focus on find a better way to apply for machine learning, including data understanding, cleaning, Data Balancing, one hot encoding, Normalization, PCA, Predictive Modelling, Predicting, Evaluation and what I found.  YINGMIN ZHAO  Data Science |

Contents

[1. Introductory 2](#_Toc52135261)

[1.1. Background 2](#_Toc52135262)

[1.2. Purposes and Requirements 2](#_Toc52135263)

[2. Methodology 3](#_Toc52135264)

[2.1. Data sources 3](#_Toc52135265)

[2.1.1. Data Cleaning 4](#_Toc52135266)

[2.1.1.1. feature selection 4](#_Toc52135267)

[2.1.1.2. Data Formatting 5](#_Toc52135268)

[2.1.2. Data Understanding and Pre-processing 6](#_Toc52135269)

[2.1.2.1. Distribution of sub-categories 6](#_Toc52135270)

[2.1.2.2. Relationship to Severity 7](#_Toc52135271)

[2.1.2.3. Pearson Correlation 11](#_Toc52135272)

[2.1.2.4. Unbalanced Data 12](#_Toc52135273)

[2.1.2.5. One-Hot Encoding 13](#_Toc52135274)

[2.1.2.6. Principal component analysis (PCA) and Dimensionality Reduction 13](#_Toc52135275)

[2.2. Predictive Model 14](#_Toc52135276)

[2.2.1. Building Baseline 14](#_Toc52135277)

[2.2.2. Cross Validation for Model Parameter Optimization 16](#_Toc52135278)

[3. Results 17](#_Toc52135279)

[3.1. Result with Original Data(unbalanced) 18](#_Toc52135280)

[3.2. Result with Down-sampling Data 20](#_Toc52135281)

[3.3. Result with Up-sampling Data 22](#_Toc52135282)

[4. Conclusion & Discussion 24](#_Toc52135283)

[5. References 24](#_Toc52135284)

[6. Appendix 24](#_Toc52135285)

# Introductory

## Background

The number of traffic accidents has been a rising trend globally due to increases in population and motorization. Different factors are involved in traffic crashes. If the possibility of drivers/cars getting into an accident can be predicted, the drivers would drive more carefully, change the route or travel plan. Also, the critical issue identified could be solved in advance. So how can we know the possibility of getting into a car accident? What kind factors/conditions are affecting the possibility and accident severity? This project could provide the answers/hints.

## Purposes and Requirements

The purpose of Capstone project is building a prediction system for Car Accident Severity using machine learning technologies. It involves statistical modelling considering crash severity as a dependent variable which is the target of prediction. As an option, the shared dataset of Seattle city of USA can be used for it, or you can find own dataset from open data source.

I would like to research on car crash statistics of Australia instead of Seattle City. Aims to understanding followings:

1. Understanding what the factors impact on car accidents
2. The possibility of car accidents happening with the certain found factors
3. Technically, what the machine learning algorithm/model are more suitable for the project.

As expecting, the data set should include labelled severity for each accident, the supervised machine learning algorithms will be applied for it.

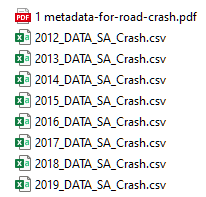
In fact, this is not a new topic, I will focus on finding the suitable features/variables and better algorithm.

# Methodology

## Data sources

I am interested in crash statistic of Australia where I am living, even though a shared dataset sample is provided by this project. I would like to run it on my own dataset. Unfortunately, I only found a good one for South Australia, not NSW’s. It fully covers 2012 – 2019 and would be pretty good to support this task. (All relevant materials are shared in Appendix.)

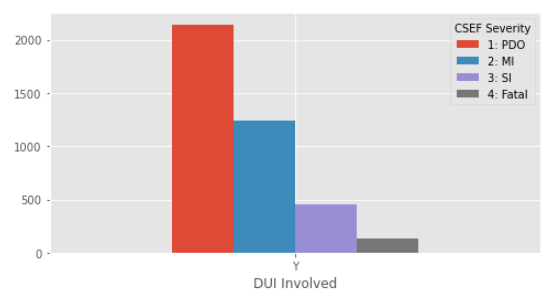
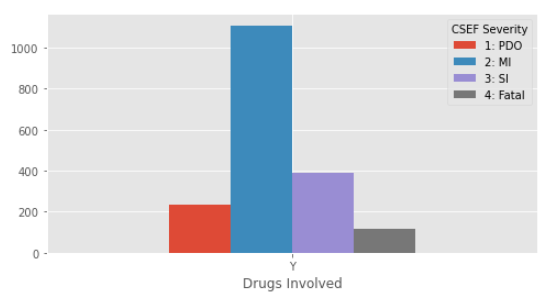
The data set have seperate csv files for each year with metadata. There are total 127672 records and 33 attributes.



## Data Cleaning

## feature selection

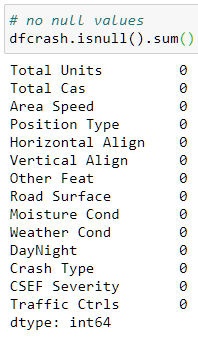
Removed “records index, Date/time and location information” which are not related to the accidents; The number of injuries is also removed, which just a result and won’t work as an independent variable; Drunk and drug involvement indicators only have small samples(<3% of 127672 rows), and the portion(vs. Severity) doesn’t show the correlation to severity, both are dropped as well. Now there are all Non-human factors as independent features.



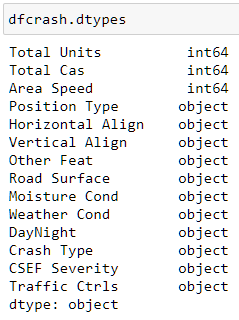
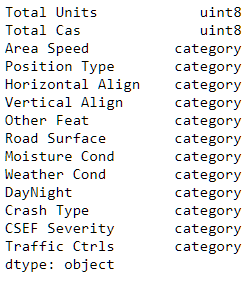
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Features | Description | Removed Columns | |
| 1 | Total Units | The total number of units involved in a road crash | **REPORT\_ID** | **DUI Involved** |
| 2 | Total Cas | Total number of casualties (fatalities + treated injuries) as a result of a road crash | **Stats Area** | **Drugs Involved** |
| 3 | Area Speed | The speed limit at the time and location of the crash | **Suburb** | **ACCLOC\_X** |
| 4 | Position Type | Identifying if a crash location was at an intersection or midblock | **Postcode** | **ACCLOC\_Y** |
| 5 | Horizontal Align | Defines the horizontal alignment of the road at the sight of the crash | **LGA Name** | **UNIQUE\_LOC** |
| 6 | Vertical Align | Defines the vertical alignment of the road at the sight of the crash | **Total Fats** | **Entity Code** |
| 7 | Other Feat | Defines other relevant features of the crash site locations | **Total SI** | **Unit Resp** |
| 8 | Road Surface | Defines the road surface type at the crash location | **Total MI** | **Day** |
| 9 | Moisture Cond | Defines the pavement surface moisture condition at the crash location | **Year** | **Time** |
| 10 | Weather Cond | Defines the weather condition at the time and location of the crash | **Month** |  |
| 11 | DayNight | The lighting condition at the time and location of the crash |  |  |
| 12 | Crash Type | Defines the road crash type |  |  |
| 13 | Traffic Ctrls | Defines the traffic control at the time and location of the road crash |  |  |
|  | CSEF Severity | 4: Fatal  3: SI = Serious Injury,  2: MI = Minor Injury,  1: PDO = Property Damage Only |  |  |

## Data Formatting

* It is perfect data without any null values.



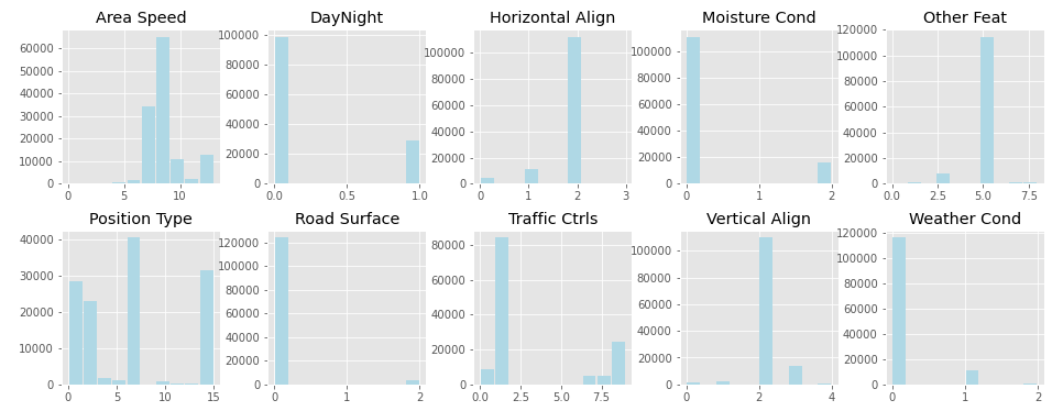
* Cast all Object type to DataFrame category type, int64 to int8 type. Area Speed is enum type so set as category.

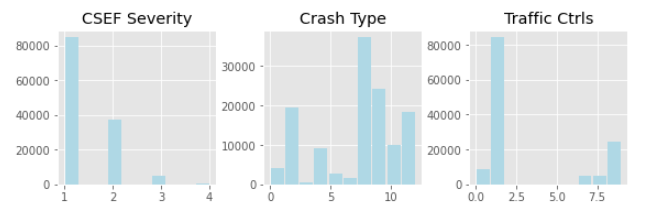
 

## Data Understanding and Pre-processing

## Distribution of sub-categories

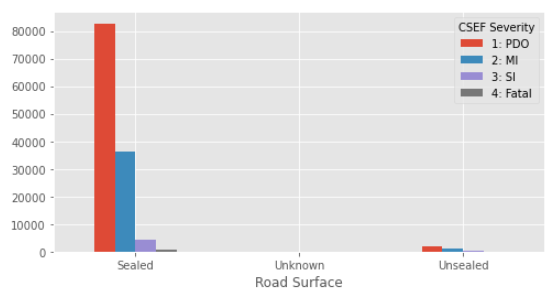
most features have imbalance sub-categories. Need furthur checking of the relationship with severity.



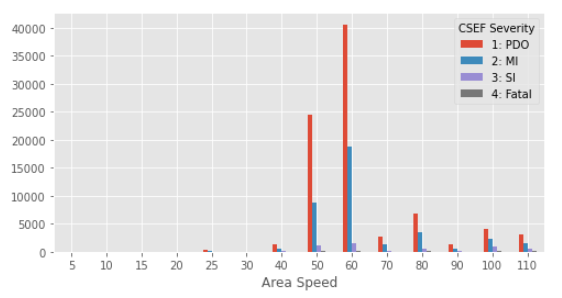


## Relationship to Severity

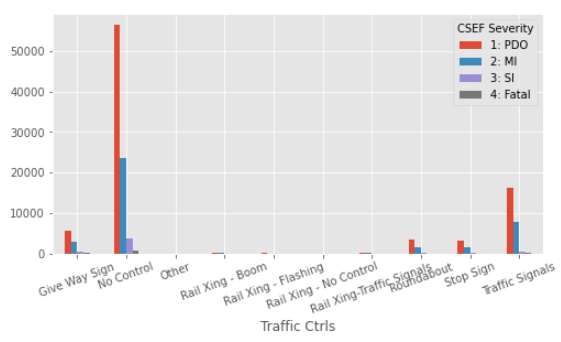
* Each categories of Road surface shows obvious tendency. It would be one of right features potentially.



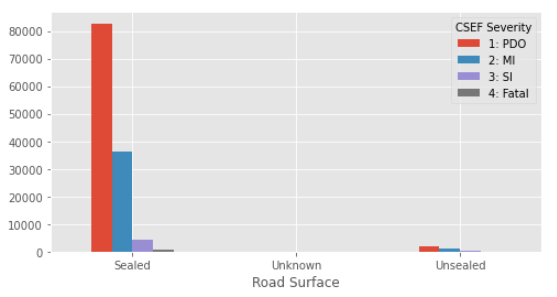
* 50&60 KM of Area Speed connected to 75% accidents, but each category has similar portion of severity.



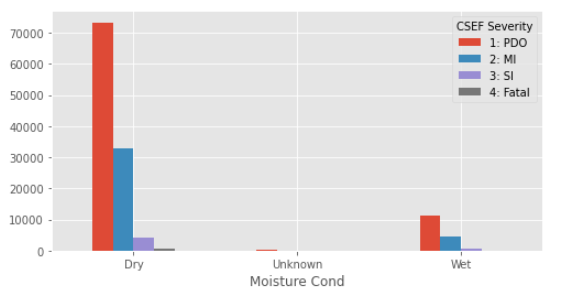
* 60% accidents happened in the area without traffic control. Main categories have similar portion of severity.



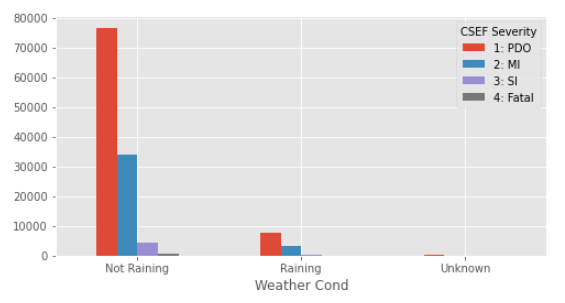
* Most accidents happened on sealed road and 60% Severity 1: POD of obvious tendency.



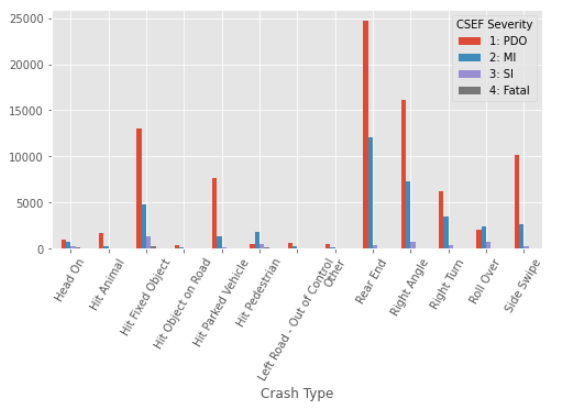
* Most accidents happened on dry road and 55% Severity 1: POD of obvious tendency.



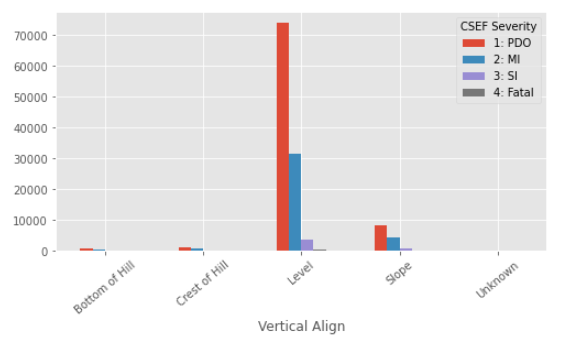
* Most accidents happened on no-raining day and 60% Severity 1: POD of obvious tendency.



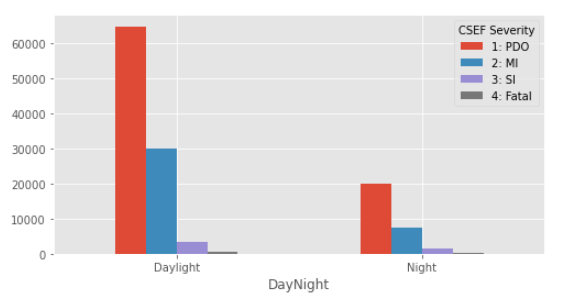
* There are many Crash type with Relative average distribution and similar portion of severity.



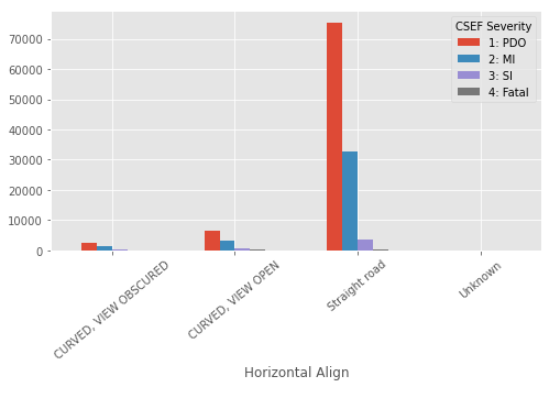
* 90% happened on level ground.



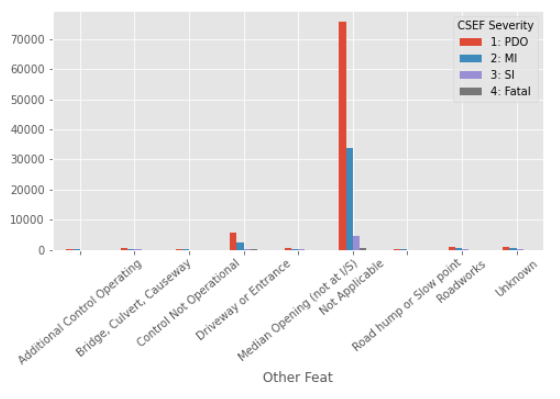
* Most happened on daytime.



* Most happened on straight road.

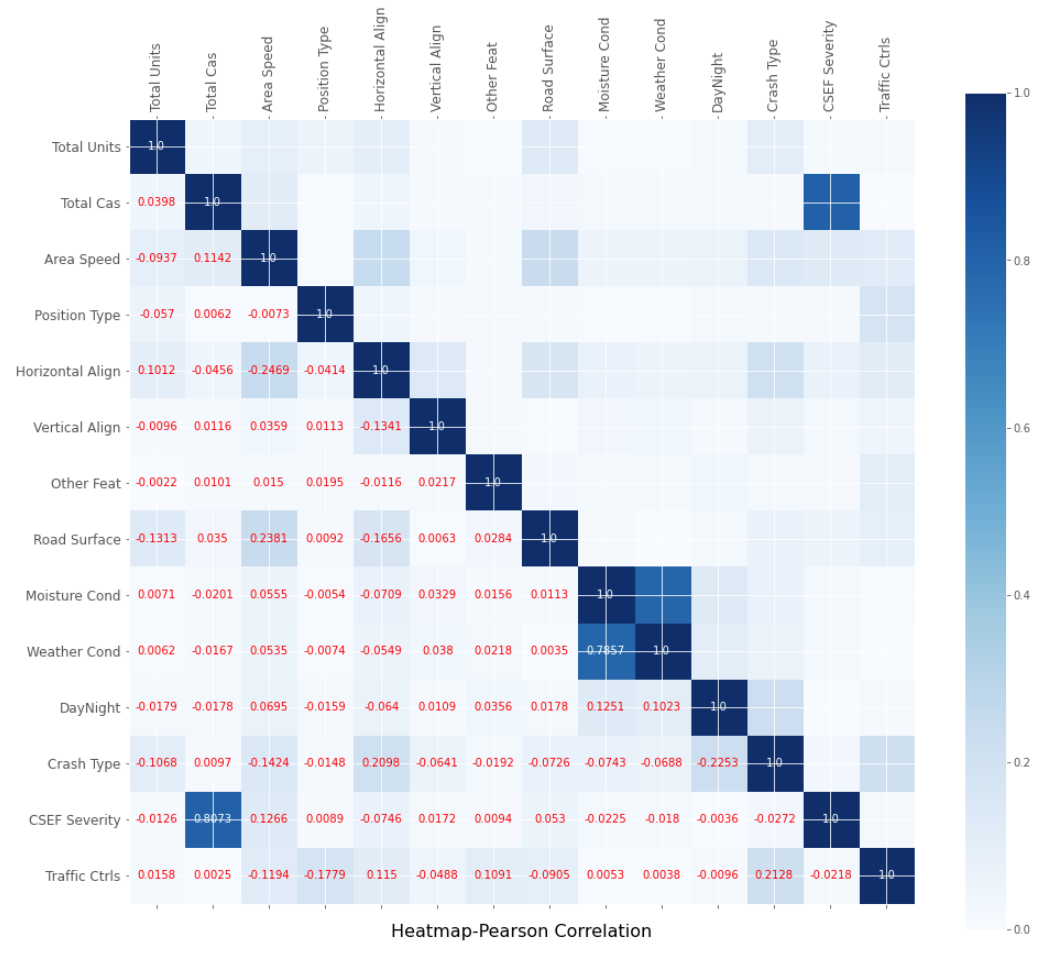


* Most happened on opening area.

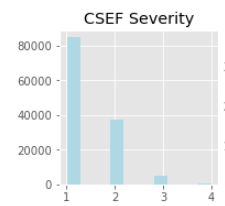


## Pearson Correlation

Overall, all single feature has the very weak positive or negative correlation. However, it does not mean we will have inaccurate prediction using these dataset as Pearson correlation only shows single correlation each other. Further correlation analysis is required.



## Unbalanced Data



Balanced portion of severity is not expected because most are minor traffic accident. That is a good thing for our life, but it is an unfavourable factor for the research. Severity 2: Minor Injury is Mild level of imbalance – 20-40% of the data set; Severity 3: Serious Injury is Moderate level – 1-20%; Severity 4: Fatal is Extreme level – < 1%.

Following data will be used and compared with the performance.

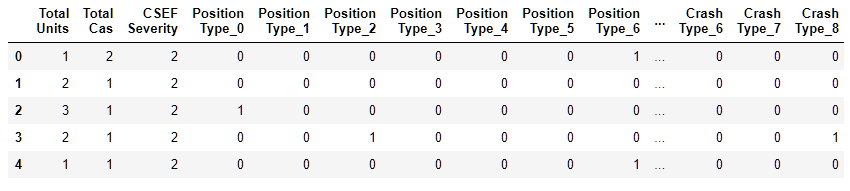
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Severity | Original Data Quantity | Up-sampling Quantity | Down-sampling Quantity | Test Data Quantity |
| 1: POD | 84775 | 84775 | 37300 | 2000 |
| 2: MI | 37300 | 84775 | 37300 | 2000 |
| 3: SI | 4875 | 84775 | 37300 | 2000 |
| 4: Fatal | 722 | 84775 | 37300 | 2000 |

## One-Hot Encoding

Most data I got are categorical. Only Decision Tree algorithms can work with it directly, with no data transform required. Many machine learning algorithms cannot operate on label data directly. They require all input variables and output variables to be numeric.

Integer Encoding and One-Hot Encoding can be used to convert categorical data to numerical data. For categorical variables where no such ordinal relationship exists, the integer encoding is not enough. It would lead the model to assume a natural ordering between categories may result in poor performance or unexpected results.

So, One-Hot encoding is applied.

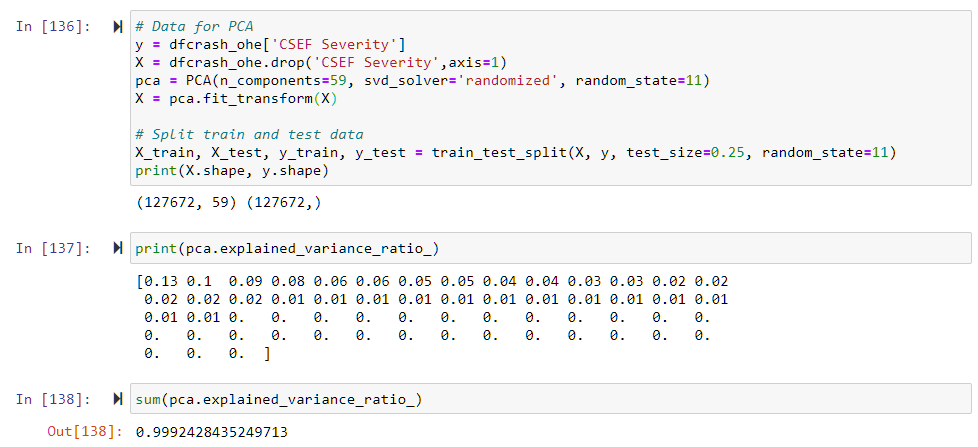


## Principal component analysis (PCA) and Dimensionality Reduction

One-hot encoding added many new binary variables/columns, total 80 variables. Also, in previous section, Pearson correlation could not find any strong relationship.

Features selection need be implemented for two purposes: 1. Measuring correlation for multiple dimensional variables; 2. Dimensionality Reduction.

I reduced Dimensionality to 59 components which cover 99.9% variance ratio.



Simultaneously, the data set is normalized by PCA; Next, Split the dataset to 75% for training and 25% for testing; the random state is set to 11 for models; Then the dataset is ready to be fed to model.

## Predictive Model

This is a multiple classification problem. I would like to use several algorithms with supervised method to find the best model - K-Nearest Neighbours model, Decision Tree model, Logistic Regression model and Support Vector Machine.

## Building Baseline

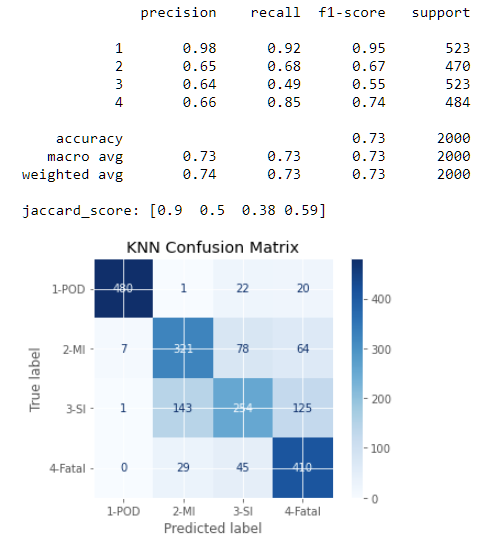
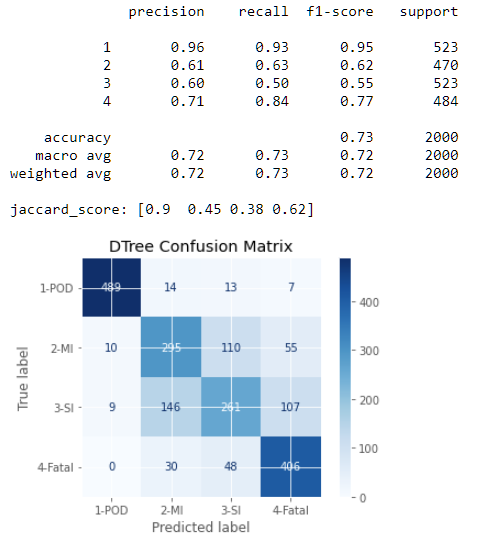
To reduced running time, I created a small dataset and built the baseline models.

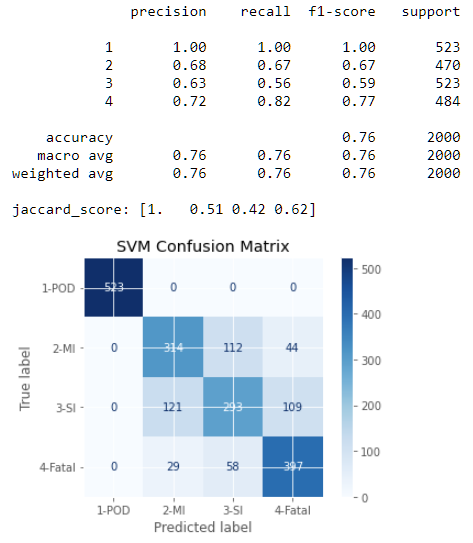
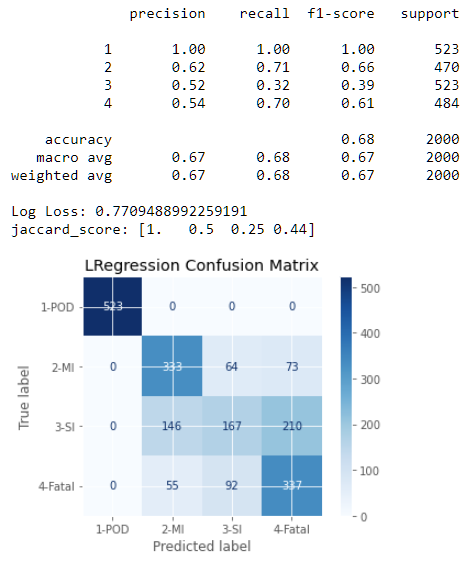
Severity 1: POD - reached 95% ~ 100% accuracy.

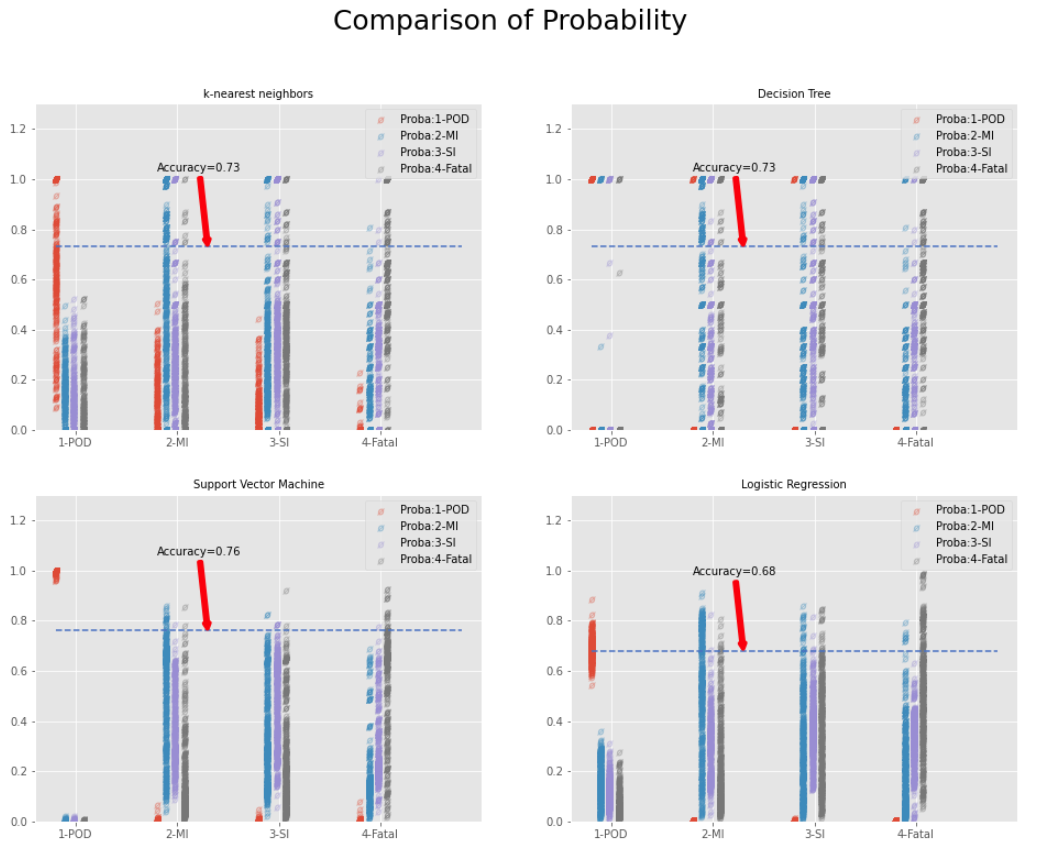
Severity 2: MI - has 65% accuracy.

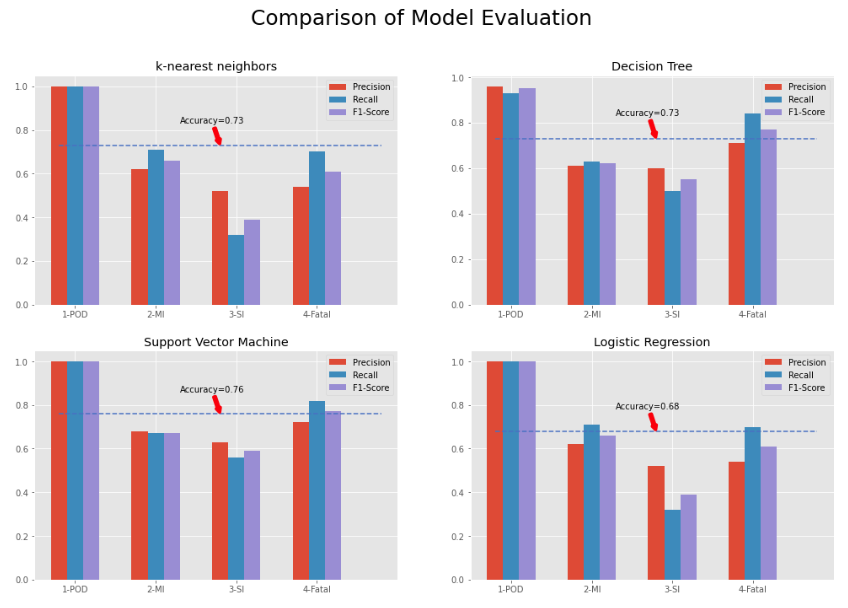
Severity 1: SI - only 40% ~ 55% accuracy.

Severity 1: Fatal - has 60% ~ 77% accuracy.





## Cross Validation for Model Parameter Optimization

Compare randomized search and grid search for optimizing hyperparameters of SVM. All parameters that influence the learning are searched simultaneously (except for the number of estimators, which poses a time / quality trade-off).

The randomized search and the grid search explore exactly same space of parameters. The result in parameter settings is quite similar, while the run time for randomized search is drastically lower.

The performance is may slightly worse for the randomized search and is likely due to a noise effect and would not carry over to a held-out test set.

I selected randomized search cross validation as it takes less time.

|  |  |  |  |
| --- | --- | --- | --- |
| K-nearest neighbours | Decision Tree | Support Vector Machine | Logistic Regression |
| weights='distance' | Splitter=’best' | Kernel=’rbf’ | Solver=’liblinear’ |
| p = 1 | max\_features'=’sqrt' | Gamma=’scale’ | Penalty=’l2’ |
| n\_neighbors = 60 | max\_depth=90 | Degree=5 | C=5 |
| leaf\_size = 10 | Criterion='entropy’ | C=10 |  |
| Algorithm = ‘kd\_tree’ |  |  |  |

# Results

Based on the results, all models are ranked as following. Refer to next 3 section as details.

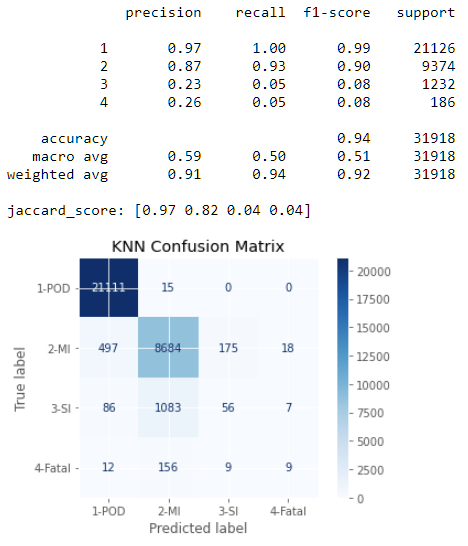
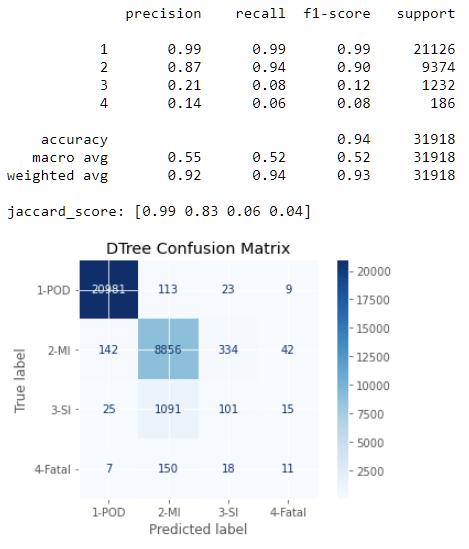
|  |  |  |  |
| --- | --- | --- | --- |
| Rank1 | Rank2 | Rank3 | Rank4 |
| Up-sampling Data | Down-sampling Data | Baseline | Original Data |
| Decision Tree | Support Vector Machine | K-nearest neighbours | Logistic Regression |

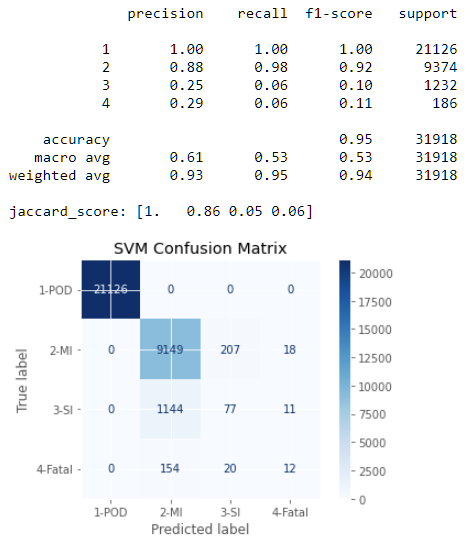
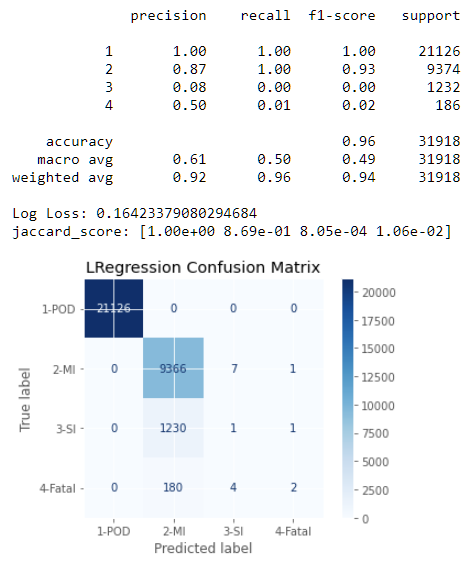
Average Accuracy

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Dataset | Quantity | K-nearest neighbours | Decision  Tree | Support Vector Machine | Logistic  Regression | Remark |
| Baseline | 8000 | 0.73 | 0.73 | 0.76 | 0.68 |  |
| Original data | 127672 | 0.94 | 0.94 | 0.95 | 0.96 | Poor accuracy for Severity 3&4 |
| Down-sampling data | 149200 | 0.82 | 0.84 | 0.84 | 0.68 |  |
| Up-sampling data | 339100 | 0.85 | 0.86 | 0.85 | 0.67 | Best |

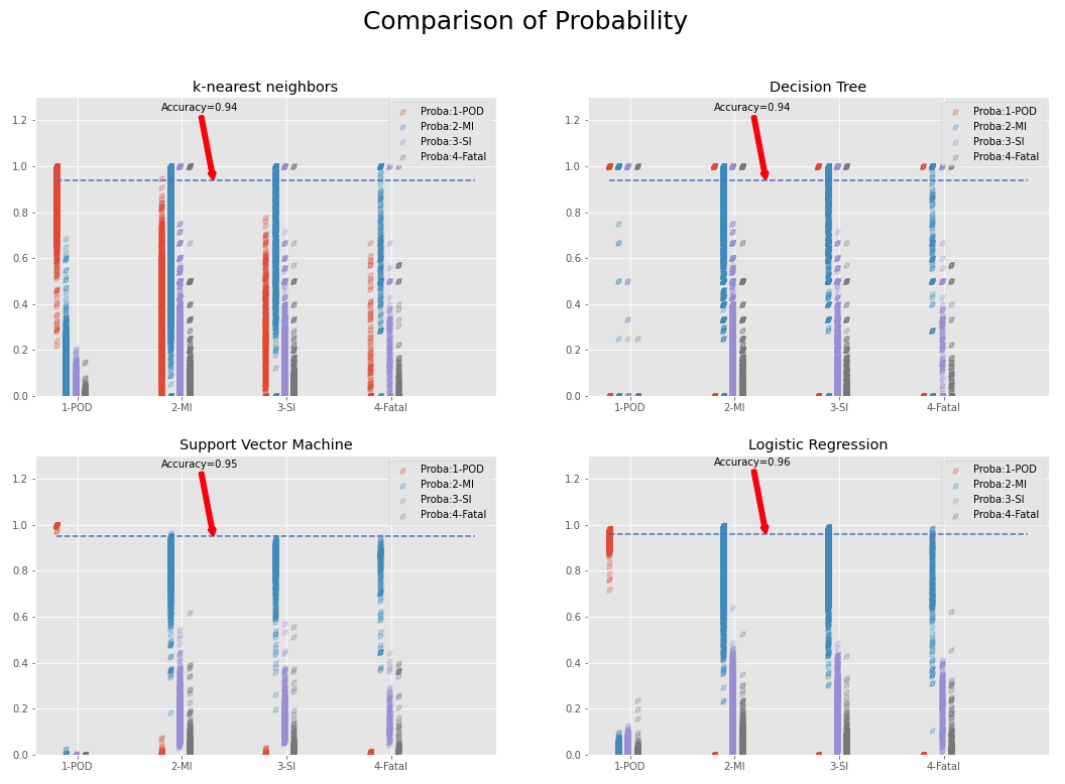
# Result with Original Data(unbalanced)

Original Data only has limited samples for Severity 3&4, Model had not enough training on it. Even though got high accuracy on Severity 1&2, Its results for reference only. If only predict Severity 1&2, could be a pretty result.

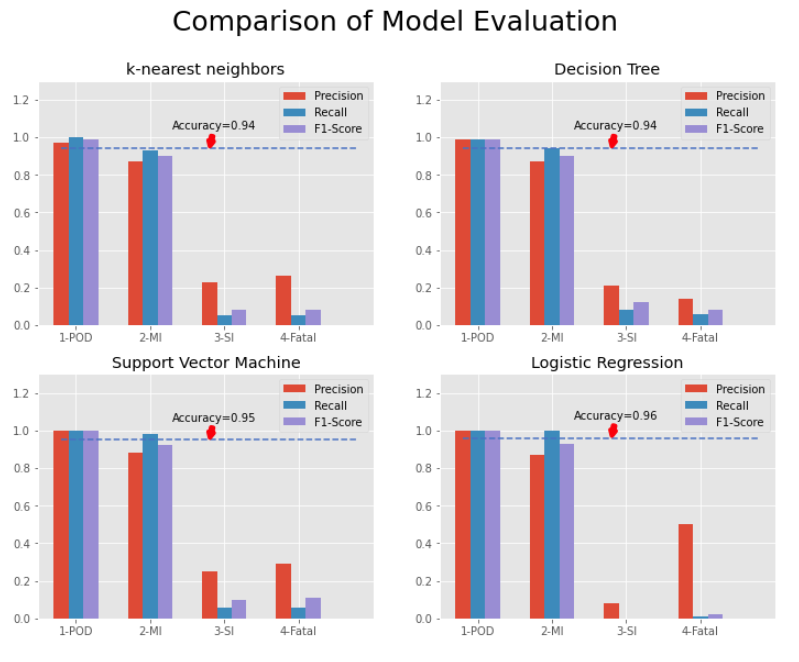
 

Probability Scatter shows Severity 3&4 were predicted as Severity 2.



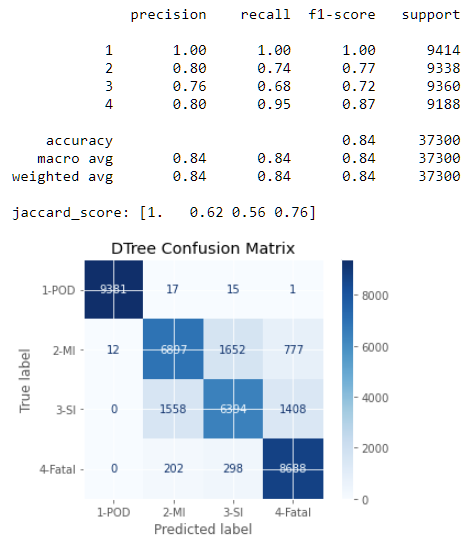
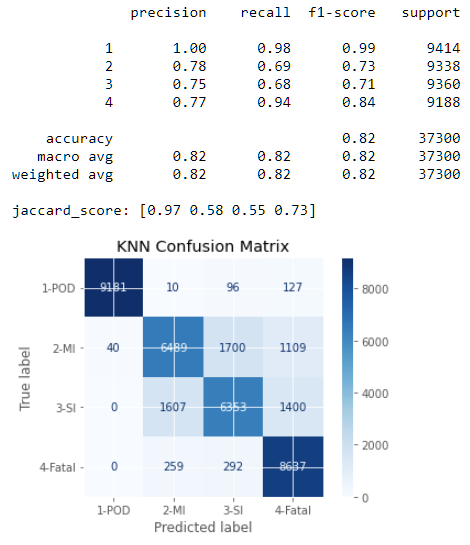
Note: The point in Probability scatter was shifted 0.09 per category on X Axes, to avoid overlap.

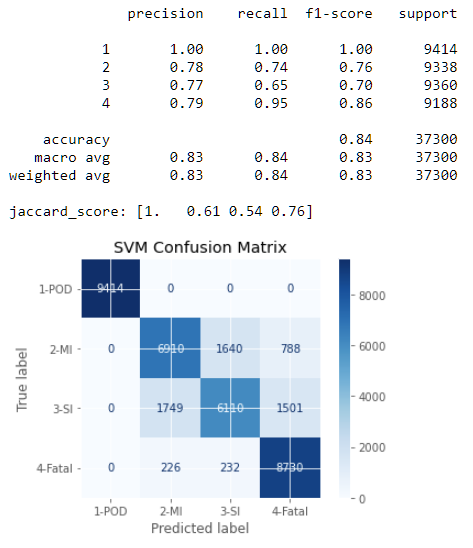
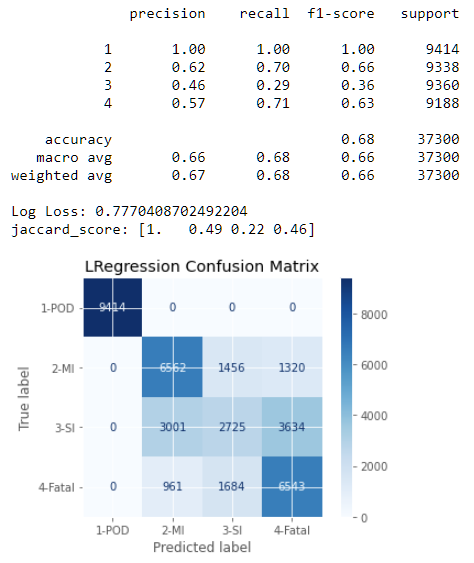


# Result with Down-sampling Data

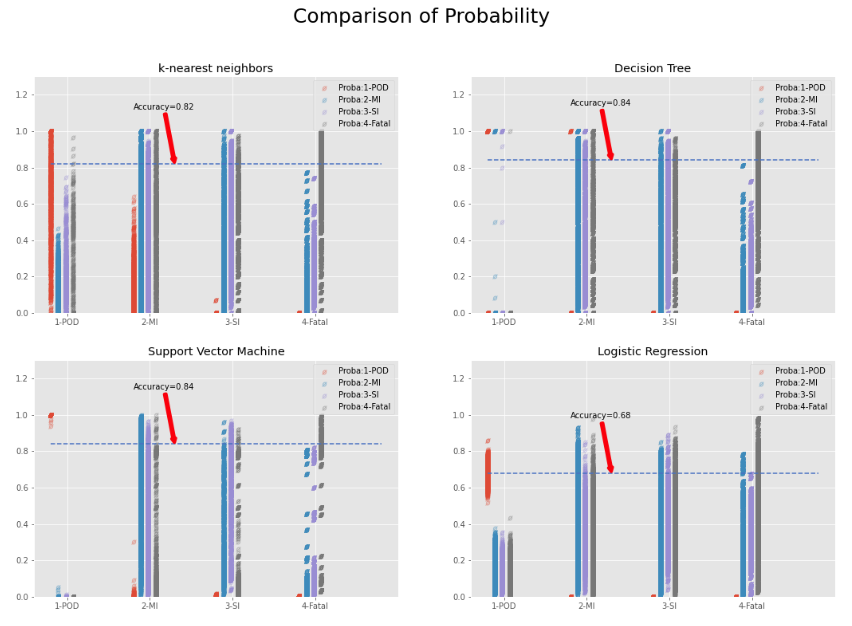
Regarding Down-sampling data, actually, Severity 1’s is down-sampling, Severity 3&4’s is up-sampling.

Using Down-sampling data, overall result is comparable and balanced. Decision Tree is the best model. Support Vector Machine has same accuracy – 84%.

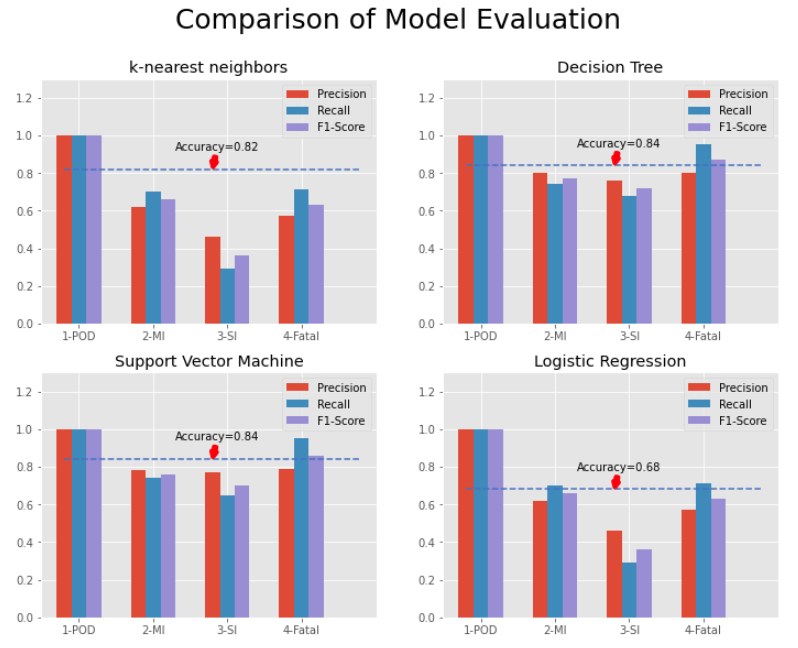


The distribution of probability for Severrity 2&3 is similar – Probability of Severity 2 is higher. It cause low predicting accuracy on Severity 3.

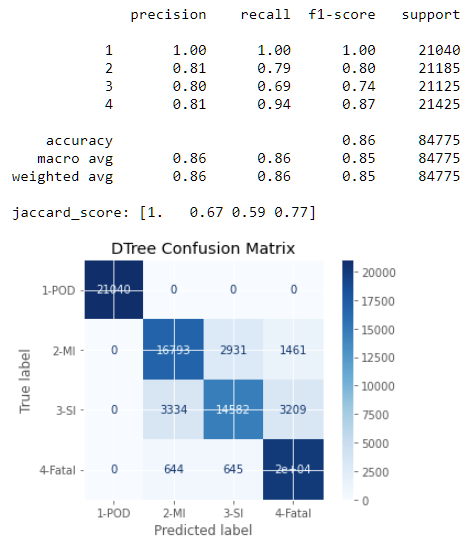
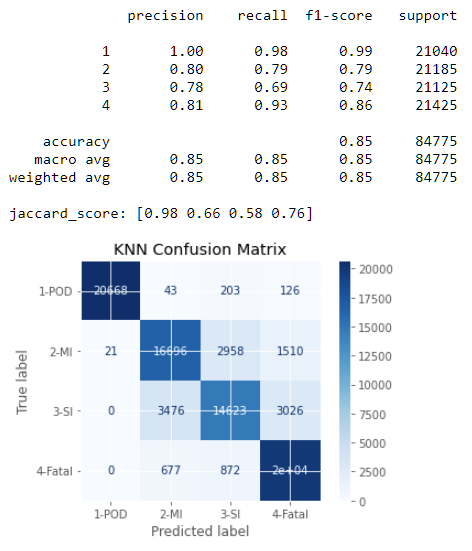


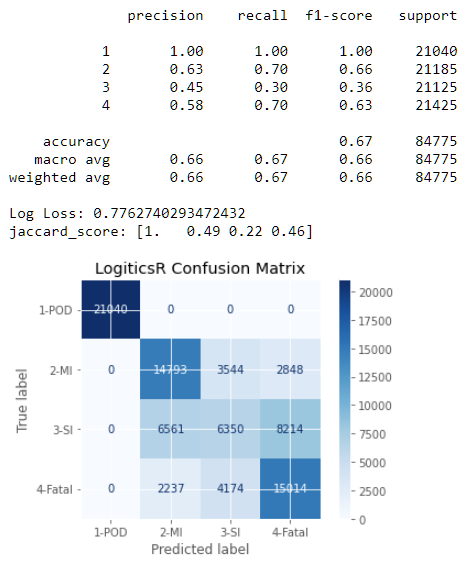
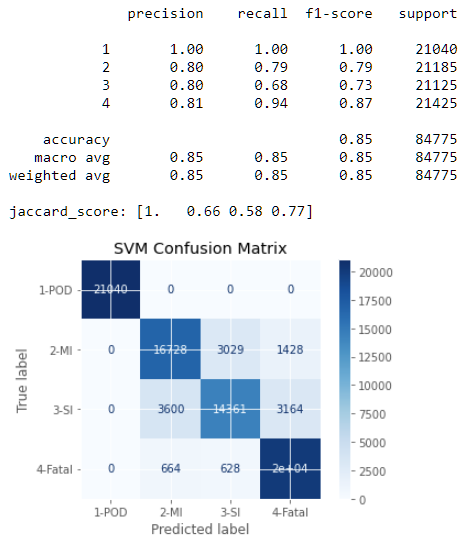
Note: The point in Probability scatter was shifted 0.09 per category on X Axes, to avoid overlap.



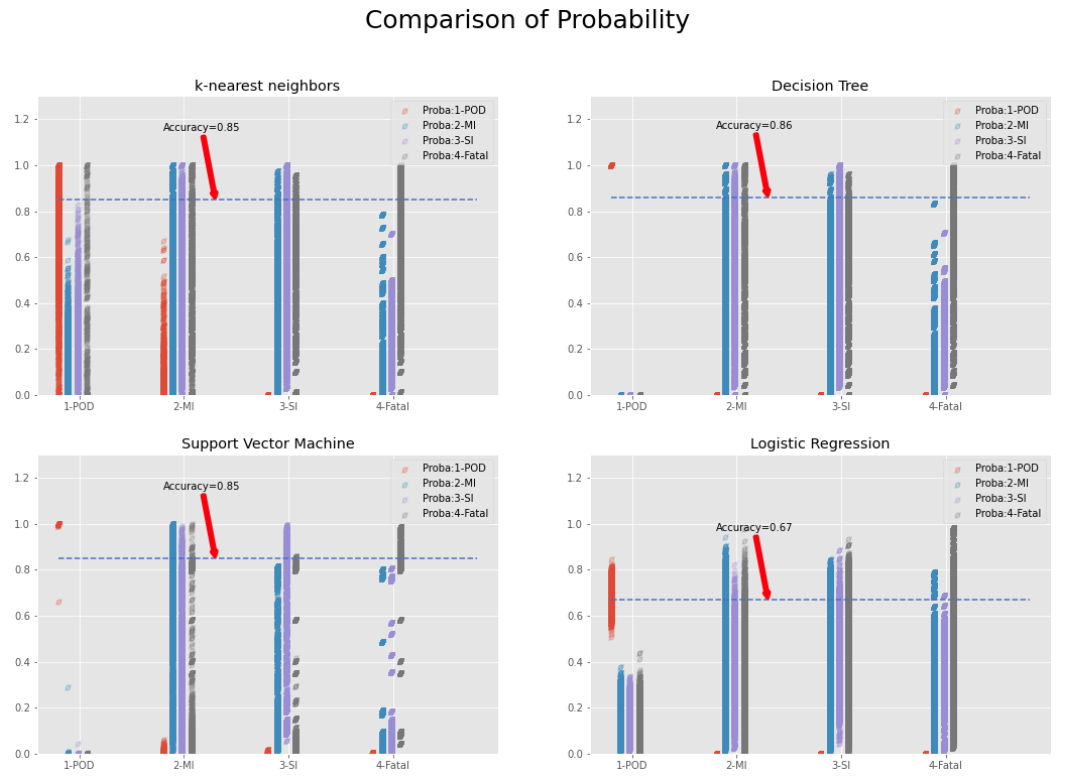
# Result with Up-sampling Data

Using Up-sampling data, overall result is most comparable and balanced. Same as Down-sampling data, Decision Tree also is the best model – 86% Accuracy. Support Vector Machine reached 85% accuracy.

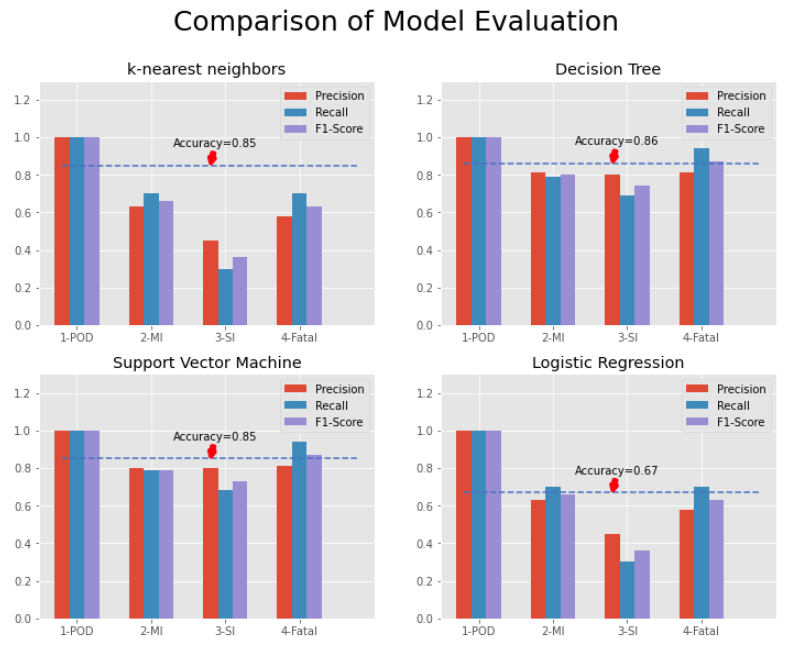




Same as Down-sampling data, the distribution of probability for Severrity 2&3 is similar – Probability of Severity 2 is higher. It cause low predicting accuracy on Severity 3.



Note: The point in Probability scatter was shifted 0.09 per category on X Axes, to avoid overlap.



# Conclusion & Discussion

Decision Tree model have the best performance for multiple classification with Up-sampling data; Its 86% average accuracy is a quality result for this project; The result of Logistic Regression and K-Nearest Neighbours model are acceptable; Logistic Regression is the worst one, which solve the problem based on binary classification algorithm.

And all models have the similar trend compared with Baseline, Down-sampling data and Up-sampling data. More sampling for training, more accurate.

Every single feature has the weak correlation, even though checked the multi-dimension correlation using PCA, but it was invisible for features.

Down-sampling improved 11% from baseline, using Up-sampling got 2% more improvement. Following items can be applied for further optimization.

* Measure and visualize Multi-dimension correlation
* Seeking other algorithm for multiple classification.
* Deep optimization for model parameters.
* Reduce the running time for Cross Validation and predicting.

# References

* [IBM Data Science Professional Certificate](https://www.coursera.org/specializations/ibm-data-science) Course
* Python Lib
  + SciKit-Learn Lib
  + Pandas Lib
  + Matplotlib

# Appendix

Data Source and Metadata

<https://data.sa.gov.au/data/dataset/road-crash-data>