T & T Lab - Ajay Anand

**Predicting The Severity of Airplane Accidents**

**Members:**

**Om Bhuyan (1805137)**

**Prafull Nayan (1805139)**

**Prayukta Sarkar (1805141)**

**Prince Chandra (1805142)**

**Rishabh Pandey (1805149)**

**Kalinga Institute of Industrial Technology**

**Table of Contents**

**Table of Contents :**

**1. Introduction**

1. **Project Scope**
   1. Purpose
   2. Benefits
2. **Exploratory Data and Visualization**
   1. Count of severity
   2. Heat-map
   3. Co-relation of various attributes w.r.t severity
   4. Normal distribution curve of safety score
   5. Safety score VS severity
   6. Total count VS number of violations of rules
   7. Cabin temp VS severity
   8. Turbulence VS control
   9. Max elevation VS severity
   10. Control Matric VS severity

3.11 Turbulence VS severity(on basis of violations)

1. **Feature Selection**
2. **Model building**
3. **Model Evaluation Metrics**
   1. Classification Metrics
   2. Log Loss
   3. Area Under ROC Curve
   4. Confusion Matrix
   5. Classification Report

**7. Model Comparison**

**8. Model Conclusion**

1. **INTRODUCTION :**

Flying has been the go-to mode of travel for years now; it is time-saving, affordable, and extremely convenient. According to the FAA, 2,781,971 passengers fly every day in the US, as in June 2019. Passengers reckon that flying is very safe, considering strict inspections are conducted and security measures are taken to avoid and/or mitigate any misshapen. However, there remain a few chances of unfortunate incidents.The aim is to build Machine Learning models to anticipate and classify the severity of any airplane accident based on past incidents. With this, all airlines, even the entire aviation industry, can predict the severity of airplane accidents caused due to various factors and, correspondingly, have a plan of action to minimize the risk associated with them

1. **PROJECT SCOPE:**

**2.1 Purpose:**

# .In this project, we will be comparing Bagging Methods with Boosting Methods to see which one gives a better accuracy. We will be using KNN Classifier, Random Forest, Support Vector Machines,Logistic Regression and **Hyper parameter tuning with GridSearchCV and randomizedSearchCV**.The data set consists of certain parameters recorded during the incident⁠ such as cabin temperature, turbulence experienced, number of safety complaints prior to the accident, number of days since the last inspection was conducted before the incident, an estimation of the pilot’s control given the various factors at play, and the likes.

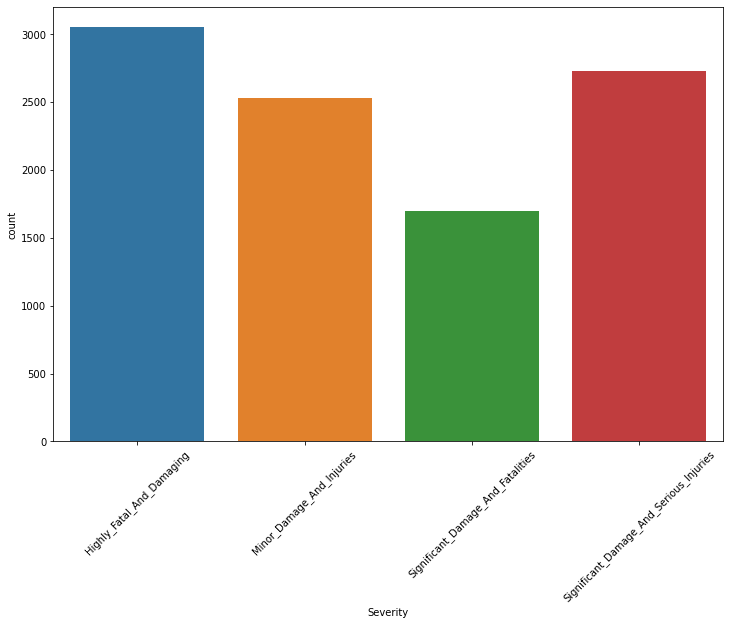
**2.2 Benefits :**

The benefits of practicing this problem by using Machine Learning techniques are as follows:

This challenge encouraged us to apply our Machine Learning skills to build models that can anticipate the severity of any airplane accident.This challenge helped us enhance our knowledge of classification actively. Classification is one of the basic building blocks of Machine Learning

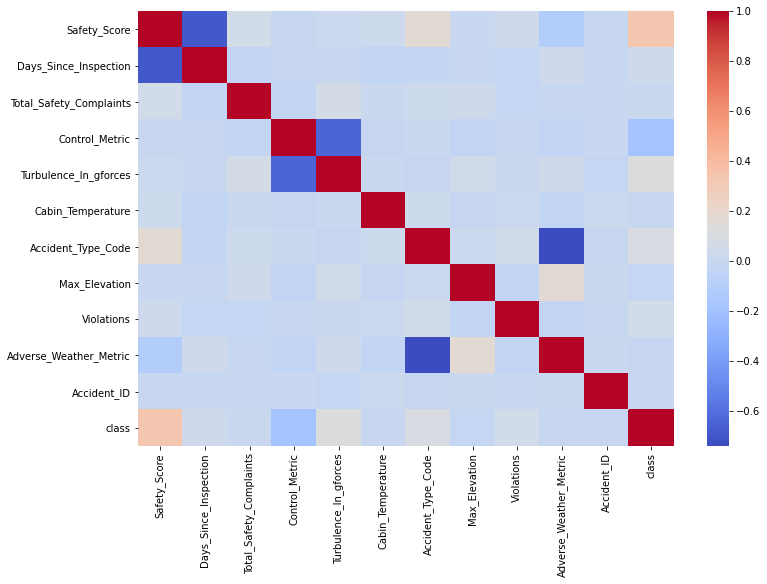
**3.Exploratory Data Analysis and Visualization**

**3.1 Count of severity.**

****

This graph shows the count of each class of airplane crash’s severity.

**3.2 Heat-map.**

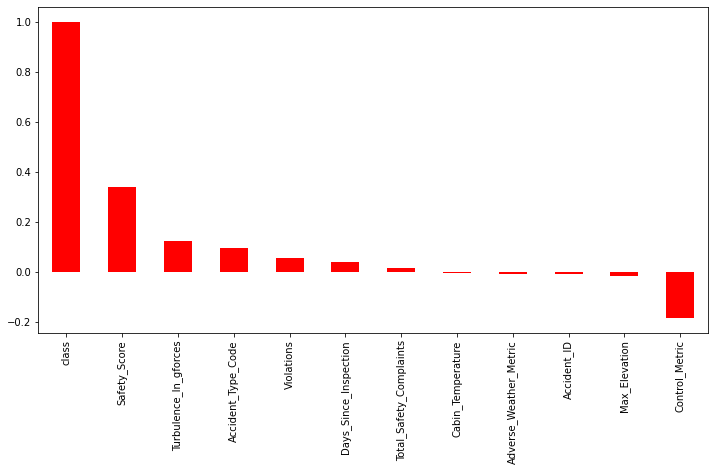


This heat map basically shows the co-relation of attributes among each other.

Positive co-relation:attributes are directly proportional.

Negative co-relation:attributes are inversely proportional.

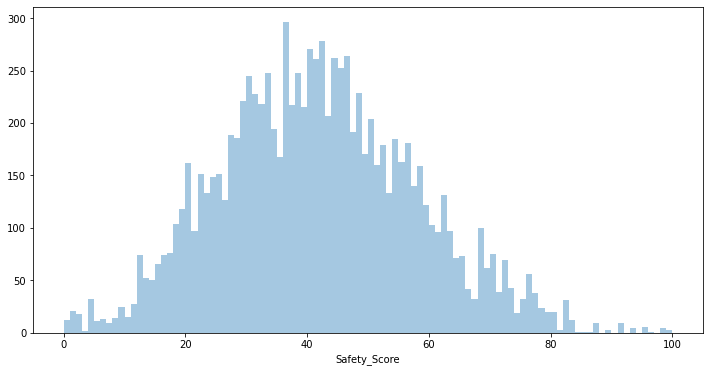
**3.3 Co-relation of various attributes w.r.t severity.**



In Y-axis we have co-relation, and on X-axis we have the attributes.

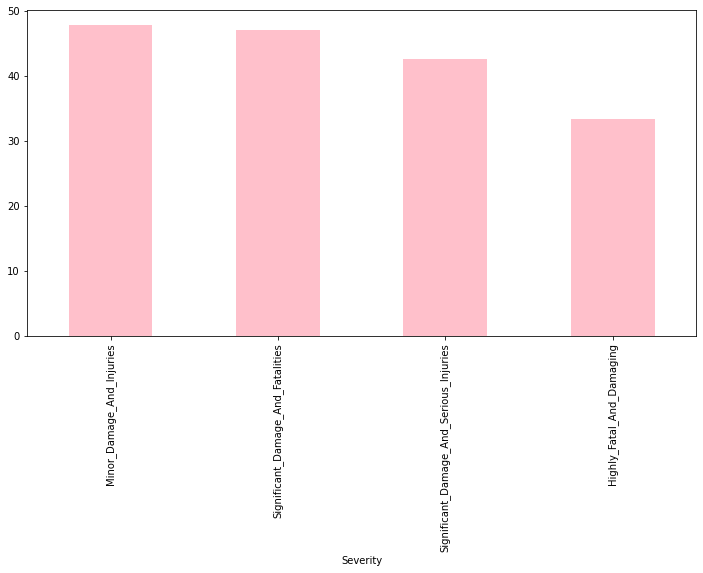
Like we can observe safety\_score is nearly 0.4 or 40% directly proportional to severity(class in our case) Control\_metric is nearly -0.2 or 20% inversely proportional to severity.

**3.4 Normal distribution curve of safety\_score.**



This curve basically shows the distribution of data points.As we can see most of the data lies between 20 to 60 Safety\_score.

**3.5 Safety\_score VS severity.**



Severity

Minor\_Damage\_And\_Injuries 47.800305

Significant\_Damage\_And\_Fatalities 47.102113

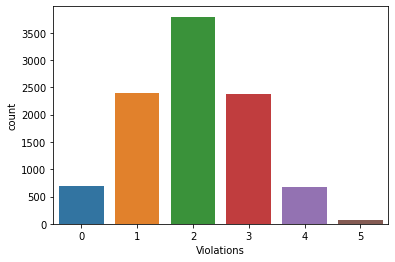
Significant\_Damage\_And\_Serious\_Injuries 42.667443

Highly\_Fatal\_And\_Damaging 33.353611

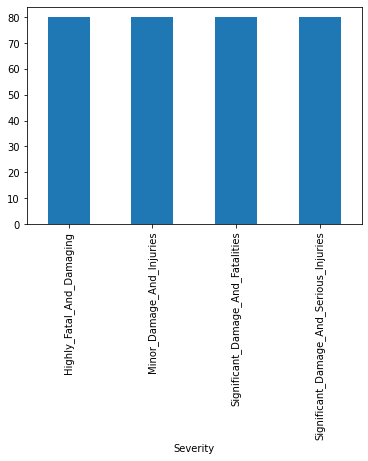
As we can observe from the graph as well as from the table for the records whose safety\_score is high damage is low.

So we can conclude that safety\_score is inversely co-related with the damage of aircraft.

**3.6 Total\_count VS number of violations of rules.**



As we can observe from the above graph most of the damages are caused due to violations of 2 rules.

**3.7 Cabin\_temp VS severity**

Severity

Highly\_Fatal\_And\_Damaging 80.023526

Minor\_Damage\_And\_Injuries 79.903961

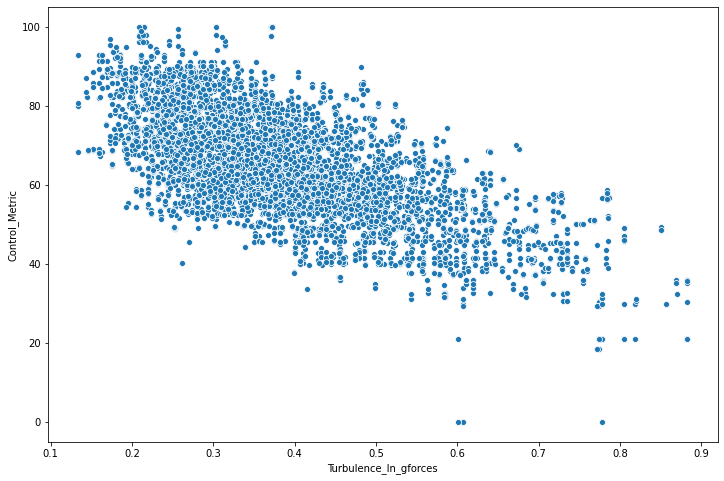
Significant\_Damage\_And\_Fatalities 80.041516

Significant\_Damage\_And\_Serious\_Injuries 79.924478

As we can observe for all kind of severity the cabin\_temp is constant.

So we can conclude that Cabin\_temp doesn’t have much effect on the plane crash.

**3.8 Turbulence VS control.**



Deriving insights from the above graph we can say that control\_metric and turbulence\_in\_gforces forms a negatively sloped graph which indicates that as the turbulence increases the control of airplane decreases and causes accident.So turbulence is one of the major factor for airplane accidents often times.

**3.9 Max\_Elevation VS severity.**



Severity

Highly\_Fatal\_And\_Damaging 31940.145431

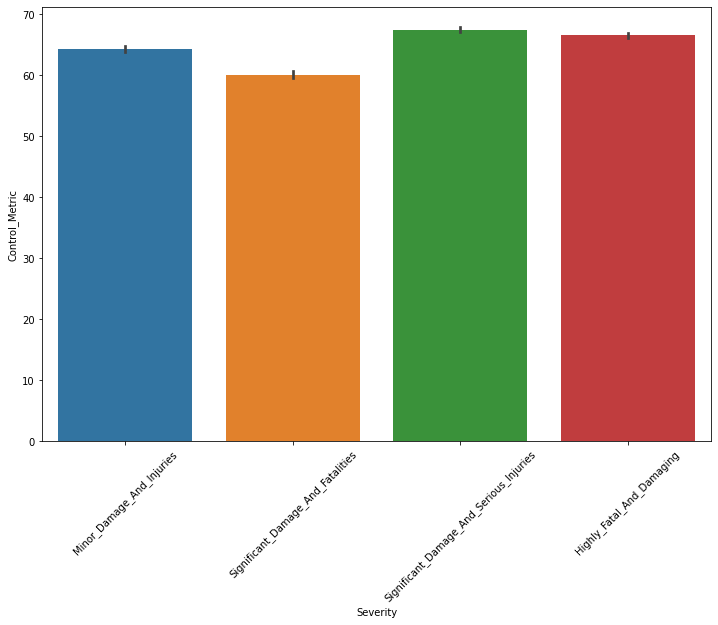
Minor\_Damage\_And\_Injuries 32225.568307

Significant\_Damage\_And\_Fatalities 31300.789137

Significant\_Damage\_And\_Serious\_Injuries 32298.893626

As we can see max\_elevation of all the four classes is roughly same. So we can predict that elevation of plane hardly affects the accident.

**3.10 Control\_matric VS severity.**



Severity

Highly\_Fatal\_And\_Damaging 66.541401

Minor\_Damage\_And\_Injuries 64.350232

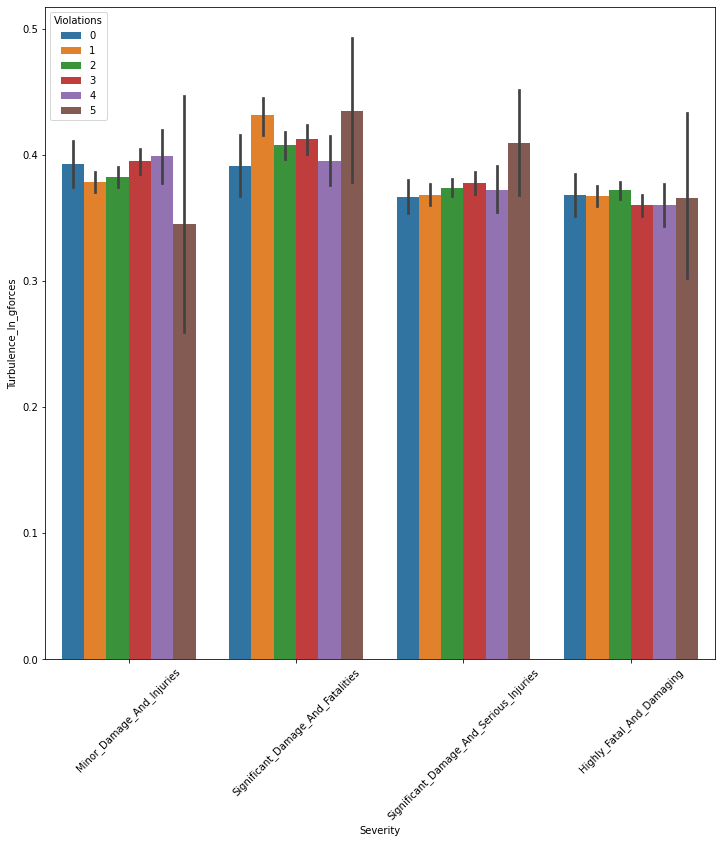
Significant\_Damage\_And\_Fatalities 60.059293

Significant\_Damage\_And\_Serious\_Injuries 67.480750

Control\_matric of significant\_damage\_and\_fatalities is the least

And control\_matrics of highly\_fatal\_and\_damaging is the most

**3.11 Turbulence VS severity(on basis of violations)**



The above graph shows the relationship between the Turbulence and severity.

1. g: For highly\_fatal\_and\_damaging class the turbulence is almost same regardless the number of violations of rules.

Similarly we can derive insights about other classes as well.

**4.Model Building**

# ****We have build models by four algorithms and then compare which algoritms has best accuracy.Algoriths are:-****

1. K Nearest Neighbour(KNN)
2. Support Vector Machine(SVM)
3. Random Forest Classifier
4. Logistic Regression

1**.K Nearest Neighbour**

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other:

The KNN algorithm hinges on this assumption being true enough for the algorithm to be useful. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics we might have learned in our childhood— calculating the distance between points on a graph.

***Note:****An understanding of how we calculate the distance between points on a graph is necessary before moving on. If you are unfamiliar with or need a refresher on how this calculation is done, thoroughly read “[Distance Between 2 Points](https://www.mathsisfun.com/algebra/distance-2-points.html)” in its entirety, and come right back.*

There are other ways of calculating distance, and one way might be preferable depending on the problem we are solving. However, the straight-line distance (also called the Euclidean distance) is a popular and familiar choice.

## **Choosing the right value for K**

To select the K that’s right for your data, we run the KNN algorithm several times with different values of K and choose the K that reduces the number of errors we encounter while maintaining the algorithm’s ability to accurately make predictions when it’s given data it hasn’t seen before.

Here are some things to keep in mind:

1. As we decrease the value of K to 1, our predictions become less stable. Just think for a minute, imagine K=1 and we have a query point surrounded by several reds and one green (I’m thinking about the top left corner of the colored plot above), but the green is the single nearest neighbor. Reasonably, we would think the query point is most likely red, but because K=1, KNN incorrectly predicts that the query point is green.
2. Inversely, as we increase the value of K, our predictions become more stable due to majority voting / averaging, and thus, more likely to make more accurate predictions (up to a certain point). Eventually, we begin to witness an increasing number of errors. It is at this point we know we have pushed the value of K too far.
3. In cases where we are taking a majority vote (e.g. picking the mode in a classification problem) among labels, we usually make K an odd number to have a tiebreaker.

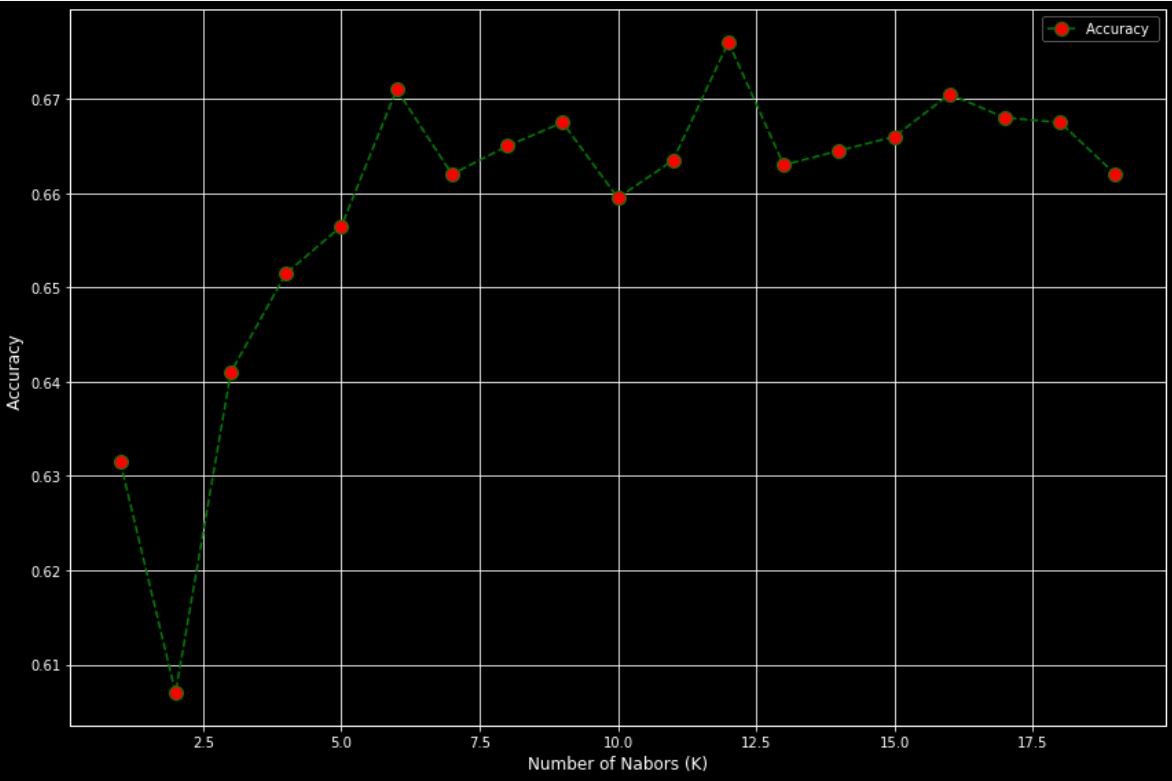


Fig:showing k value near 12 is giving best accuracy.

## **Support Vector Machines**

Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks. But, it is widely used in classification objectives.

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points.



Fig:Showing hyperplanes and support vectors.

## **Hyperplanes and Support Vectors**

Hyperplanes are decision boundaries that help classify the data points. Data

points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds 3.

Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

**3.Logistic Regression**

Logistic Regression is a Machine Learning algorithm which is used for the classification problems, it is a predictive analysis algorithm and based on the concept of probability.

We can call a Logistic Regression a Linear Regression model but the Logistic Regression uses a more complex cost function, this cost function can be defined as the ‘****Sigmoid function****’ or also known as the ‘logistic function’ instead of a linear function.

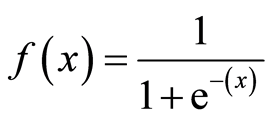
The hypothesis of logistic regression tends it to limit the cost function between 0 and 1. Therefore linear functions fail to represent it as it can have a value greater than 1 or less than 0 which is not possible as per the hypothesis of logistic regression.

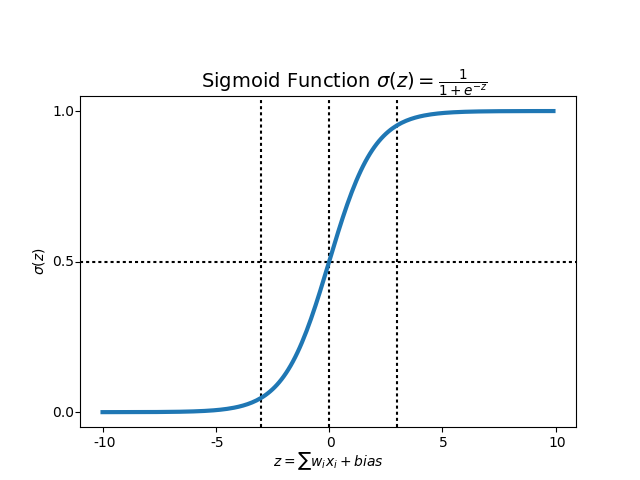
IMG_256

Fig: Logistic regression hypothesis

**Sigmoid Function**

In order to map predicted values to probabilities, we use the Sigmoid function. The function maps any real value into another value between 0 and 1. In machine learning, we use sigmoid to map predictions to probabilities.





1. **Random Forest Classifier**

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. See in figure below:

The reason that the random forest model works so well is:

*****A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.*****



**How does the algorithm work?**

**It works in four steps:**

1. Select random samples from a given dataset.
2. Construct a decision tree for each sample and get a prediction result from each decision tree.
3. Perform a vote for each predicted result.

4.Select the prediction result with the most votes as the final prediction.

**5.HyperParameter Tuning**

 M***odel Parameters*** are the properties of training data that will learn on its own during training by the classifier or other ML model. For example, weights and biases, or split points in Decision Tree.

***Model Hyperparameters*** are instead properties that *govern the entire training process*. They include variables which determines the network structure (for example, Number of Hidden Units) and the variables which determine how the network is trained (for example, Learning Rate). Model hyperparameters are set *before* training (before optimizing the weights and bias).

Choosing good hyperparameters provides two main benefits:

* Efficient search across the space of possible hyperparameters; and
* Easier management of a large set of experiments for hyperparameter tuning.

****Hyperparameters Optimisation Techniques****

The process of finding most optimal hyperparameters in machine learning is called ****hyperparameter optimisation.****

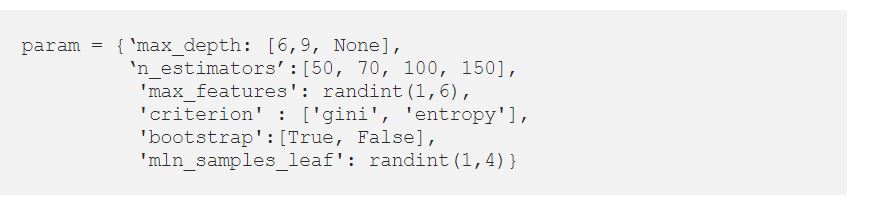
Common algorithms include:

* Grid Search
* Random Search
* Bayesian Optimisation

**Here we will see only Random Search**.

RandomizedSearchCV is very useful when we have many parameters to try and the training time is very long. For this example, I use a random-forest classifier, so I suppose you already know how this kind of algorithm works.

The first step is to write the parameters that we want to consider and from these parameters select the best ones.



Now we can create our ****RandomizedSearchCV**** object and fit the data. Finally, we can find the best parameters and the best scores.



## 

## **6.MODEL EVALUATION METRICS:**

## **6.1 Classification Metrics**

Classification problems are perhaps the most common type of machine learning problem and as such there are a myriad of metrics that can be used to evaluate predictions for these problems.

In this section we will review how to use the following metrics:

1. Classification Accuracy.
2. Log Loss.
3. Area Under ROC Curve.
4. Confusion Matrix.
5. Classification Report.
6. F1 Score

### **6.2 Classification Accuracy**

Classification accuracy is the number of correct predictions made as a ratio of all predictions made.This is the most common evaluation metric for classification problems, it is also the most misused. It is really only suitable when there are an equal number of observations in each class (which is rarely the case) and that all predictions and prediction errors are equally important, which is often not the case.

Accuracy = No. of correct predictions / Total number of input samples

It works great if there are an equal number of samples for each class. For example, we have 90% sample of *class A* and 10% sample of *class B* in our training set. Then, our model will predict with the accuracy of 90% by predicting all the training samples belongs to *class A*. If we test the same model with a test set of 60% from class A and 40% from class B. Then the accuracy will fall, and we will get an accuracy of 60%.Classification accuracy is good but it gives False Positive sense of achieving high accuracy. The problem arises due to the possibility of miss-classification of minor class samples are very high.

### **6.3. Log Loss**

[Logistic loss](https://machinelearningmastery.com/logistic-regression-with-maximum-likelihood-estimation/) (or log loss) is a performance metric for evaluating the predictions of probabilities of membership to a given class.The scalar probability between 0 and 1 can be seen as a measure of confidence for a prediction by an algorithm. Predictions that are correct or incorrect are rewarded or punished proportionally to the confidence of the prediction. It usually works well with multi-class classification. Working of Log loss, the classifier should assign a probability for each and every class of all the samples

### **6.4. Area Under ROC Curve:**

Area Under ROC Curve (or ROC AUC for short) is a performance metric for binary classification problems.A U C of a classifier is defined as the probability of a classifier that will rank a randomly chosen positive example higher than a negative example.The AUC represents a model’s ability to discriminate between positive and negative classes. An area of 1.0 represents a model that made all predictions perfectly. An area of 0.5 represents a model as good as random. A ROC Curve is a plot of the true positive rate and the false positive rate for a given set of probability predictions at different thresholds used to map the probabilities to class labels. The area under the curve is then the approximate integral under the ROC Curve.

**True positive rate:** Also called or termed as sensitivity. True Positive Rate is considered as a portion of positive data points which are correctly considered as positive, with respect to all data points those are positives.

True Positive Rate = True Positive / (False Negative +True True Positive)

**True Negative Rate:** Also called or termed as specificity. False Negative Rate is considered as a portion of negative data points which are correctly considered as negative, with respected to all data points those are negatives.

True Negative Rate = True Negative / (True Negative + False Positive)

**False-positive Rate:** False Negative Rate is considered as a portion of negative data points which are mistakenly considered as negative, with respected to all data points those are negatives.

True Negative Rate = False Positive / (True Negative +False Positive)

### **6.5. Confusion Matrix**

The [confusion matrix](https://machinelearningmastery.com/confusion-matrix-machine-learning/) is a handy presentation of the accuracy of a model with two or more classes.The table presents predictions on the x-axis and accuracy outcomes on the y-axis. The cells of the table are the number of predictions made by a machine learning algorithm.It creates a *N X N* matrix, where N is the number of classes or categories that are to be predicted.

There are 4 terms you should keep in mind:

1. True Positives: It is the case where we predicted Yes and the real output was also yes.
2. True Negatives: It is the case where we predicted No and the real output was also No.
3. False Positives: It is the case where we predicted Yes but it was actually No.

4. False Negatives: It is the case where we predicted No but it was actually Yes

Accuracy of the matrix is always calculated by taking average values present in the main diagonal i.e.

Accuracy = (True Positive+ True Negative) / Total Sample Accuracy

### 5. Classification Report

Sci-kit-learn does provide a convenience report when working on classification problems to give you a quick idea of the accuracy of a model using a number of measures.The *classification\_report()* function displays the precision, recall, f1-score and support for each class.

## 6. F1 Score:

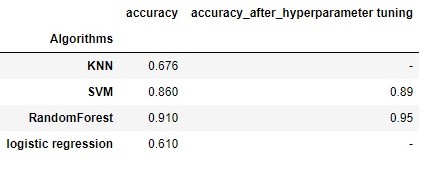
It is a harmonic mean between recall and precision. Its range is [0,1]. This metric usually tells us how precise (It correctly classifies how many instances) and robust (does not miss any significant number of instances) our classifier is.*It is used to measure the test’s accuracy.*

Precision = True Positives / (True Positives + False Positives)

Recall = True Positives / (True Positives + False-negatives)

F1 Score = (2 \* Precision \* Recall) / (Precision + Recall)

**7**.**MODEL COMPARISON:**

****

We can see that Random-Forest Classifier outperforms all other algorithms having an accuracy of 95%.

Deployment Of Project:

Once the model is train and tested we can deploy our model on flask framework for the public use.

**8.MODEL CONCLUSION:**

Accidents are rare, analyses of accident records can provide guidance on broad areas of concern but are inherently incapable of preventing other types of accidents. Incidents are more frequent and are a rich source of safety data, but the quantity of the data is so large that it is difficult to identify meaningful risks and avoid unfruitful diversions..

Although aircraft system malfunctions are involved in a large fraction of aircraft incidents and accidents, improvements in aircraft systems often improve safety by making aircraft more robust—providing flight crews with more accurate information to improve their situational awareness and reducing the likelihood that a human error will result in an incident or accident.

Machine Learning models is to anticipate and classify the severity of any airplane accident based on past incidents. With this, all airlines, even the entire aviation industry, can predict the severity of airplane accidents caused due to various factors and, correspondingly, have a plan of action to minimize the risk associated with them.

**References :-**

1 . <https://www.kaggle.com/kaushal2896/airplane-accidents-severity-dataset>

2 . Boeing. 1997. Statistical Summary of Commercial Jet Airplane Accidents, Worldwide Operations 1959- 1996: Airplane Safety Engineering . Settle, Wash.: Boeing Commercial Airplane Group.

3 . .ICAO (International Civil Aviation Organization). 1994. International Standards and Recommended Practices: Aircraft Accident and Incident Investigation. Annex 13 to the Convention on International Civil Aviation, 8th ed. Montreal, Canada: ICAO.