Feature Engineering

The performance of the model is dependent on data preprocessing and data handling. Suppose if we build a model without handling data, we get an accuracy of around 70%. By applying the Feature engineering on the same model there is a chance to increase the performance from 70% to more.

1. Feature Selection
2. Handling missing values
3. Handling imbalanced data
4. Binning
5. Encoding
6. Handling Outliers
7. Feature Scaling

**Feature Selection**

Feature selection is nothing but a selection of required independent features. Selecting the important independent features which have more relation with the dependent feature will help to build a good model.

**1.1** Correlation Matrix with Heat map

Heat map is a graphical representation of 2D (two-dimensional) data. Each Data value represents a matrix.

**1.2** Univariate Selection

Statistical tests can be used to select the independent features which have the strongest relationship with the dependent feature. Select Best method can be used with a suite of different statistical tests to select a specific number of features

**1.3** Extra Trees Classifier method

Method will help to give the importance of each independent feature with a dependent feature. Feature importance will give you a score for each feature of your data, the higher the score more important or relevant to the feature towards your output variable.

**2 Handling Missing Values**

Filling NA values with mean, median, mode or dropping the NA values

**3 Handling imbalanced data**

Why need to handle imbalanced data?

To reduce overfitting and underfitting problems. Suppose a feature has a factor level 2 (0 and 1). It consists of 1’s is 5% and 0’s is95%. It is called imbalanced data.

**3.1** Under-sampling majority class

Under-sampling the majority class will resample the majority class points in the data to make them equal to the minority class.

**3.2** Oversampling Minority class by duplication

Oversampling minority class will resample the minority class points in the data to make them equal to the majority class.

**3.3** Oversampling minority class using Synthetic Minority Oversampling Technique (SMOTE)

In this method, synthetic samples are generated for the minority class and equal to the majority class.

**4 Binning**

Binning is nothing but any data value within the range is made to fit into the bin. It is important in your data exploration activity. We typically use it to transform continuous variables into discrete ones. Suppose if we have AGE feature in continuous and we need to divide the age into groups as a feature

Then it will be useful.

**5 Encoding**

Why this will apply? Because in datasets we may contain object data types. For Building a model we need to have all features in integer data types. So, Label Encoder and OneHotEncoder are used to convert object data type to integer data type

**6 Handling Outliers**

Outliers are those data points which differs significantly from other observations present in given dataset. It can occur because of variability in measurement and due to misinterpretation in filling data points.

Different outlier detection technique

Outlier can be of two types: Univariate and Multivariate. Above, we have discussed the example of univariate outlier. These outliers can be found when we look at distribution of a single variable. Multi variant outliers are outliers in an n-dimensional space.

* Hypothesis Testing
* Z-score method
* Robust Z-score
* I.Q.R method
* Winsorization method (Percentile Capping)
* DBSCAN Clustering
* Isolation Forest
* Linear Regression Models (PCA, LMS)
* Standard Deviation
* Percentile
* Visualizing the data

**7 Feature scaling**

**7.1 Standardization (Z):** if the data is different scale then we should do standardization it will convert all data in same scale **-3.14-+3.14**

Formula Standardization **(Z) =X-mue/sigma**

**7.2 Normalization:** if the data is different in scale then we apply normalization it will convert all the data in the same way like b/w **0 -1.**

Formula normalization = **x-min/max-min**