BITS Pilani
Pilani | Dubai | Goa | Hyderabad
Work Integrated Learning Programmes

Fetal Health

Introduction to Data Science

M.Tech Data Science and Engineering

1: Overview

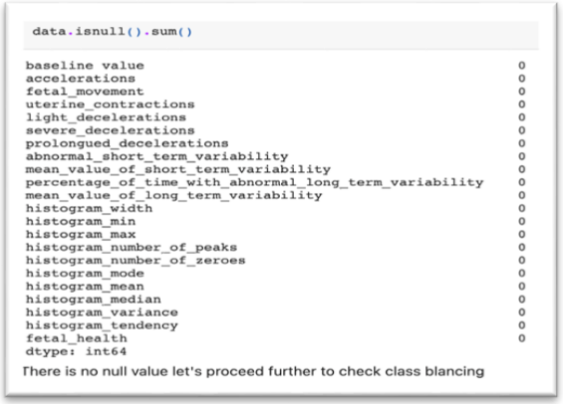
- **Objective**: We are provided with the data set, this dataset contains 2126 records of features extracted from Cardiotocogram exams, which were then classified by expert obstetrician into 3 classes: "Normal", "Suspect" & "Pathological". Our objective is to perform the following operations on the dataset:
  Predictive Analytics
  - Apply Decision Tree Classifier on the dataset. Consider the X variables as baseline value, fetal_movement, uterine_contractions, light_decelerations, severe_decelerations, prolonged_decelerations and abnormal_short_term_variability; and Y variable as fetal_health. Keep the train-test split at 70-30 ratio.
  - Then finding the overall accuracy of the decision tree model, the confusion matrix, the implications of Type-I error and Type-II error in this example. What are the Precision, Recall, F1 score for the three classes, based on the confusion matrix, Plot the ROC curves for the three classes

2: Data Acquisition

- **How many features:** 21 (21 independent variables and one dependent target variable)
- **Size of the dataset:** The dataset size: (2126, 22)
- **Multiple files:** No single input file (fetal_health.csv downloaded from https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification)
- **What kind of data – numerical or character**: Numerical Data
- **Balanced or imbalanced – what is the distribution:** imbalanced as normal is dominant
- **Distribution of Training set, validation set, testing set:** Training size:70% and test size: 30%

3: Data Wrangling

- There is no null value observed in the given dataset

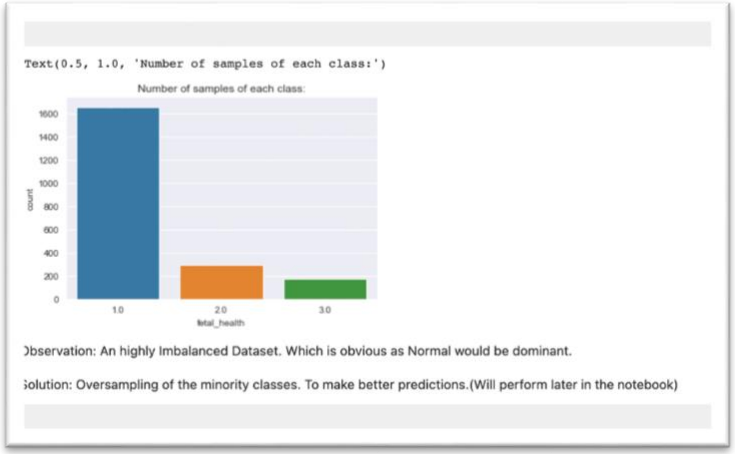


```
data.isnull().sum()
```

```
baseline value                                            0
accelerations                                             0
fetal_movement                                            0
uterine_contractions                                      0
light_decelerations                                       0
severe_decelerations                                      0
prolongued_decelerations                                  0
abnormal_short_term_variability                           0
mean_value_of_short_term_variability                      0
percentage_of_time_with_abnormal_long_term_variability    0
mean_value_of_long_term_variability                       0
histogram_width                                           0
histogram_min                                             0
histogram_max                                             0
histogram_number_of_peaks                                 0
histogram_number_of_zeroes                                0
histogram_mode                                            0
histogram_mean                                            0
histogram_median                                          0
histogram_variance                                        0
histogram_tendency                                        0
fetal_health                                              0
dtype: int64
```

There is no null value let's proceed further to check class blancing

Highly imbalanced dataset

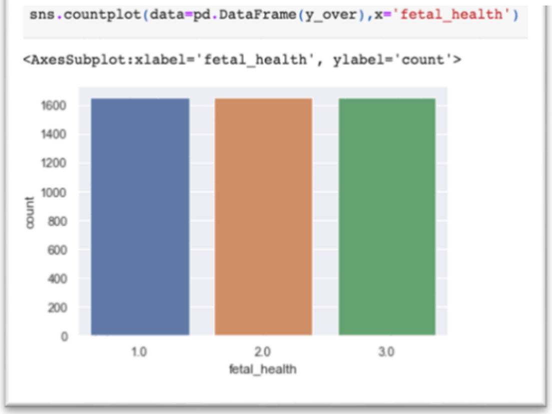


```
Text(0.5, 1.0, 'Number of samples of each class:')
```

Observation: An highly Imbalanced Dataset. Which is obvious as Normal would be dominant.

Solution: Oversampling of the minority classes. To make better predictions.(Will perform later in the notebook)

Now all the classes have same no of samples after oversampling of minority classes



```
sns.countplot(data=pd.DataFrame(y_over),x='fetal_health')
```

```
<AxesSubplot:xlabel='fetal_health', ylabel='count'>
```

Observation: Attributes such as severe_decelerations, prolongues_decelerations are more or less constant with few variations. We removed them using PCA by getting 3 components with highest variance and 95% total variance in Dimensionality Reduction.

Fetal Health

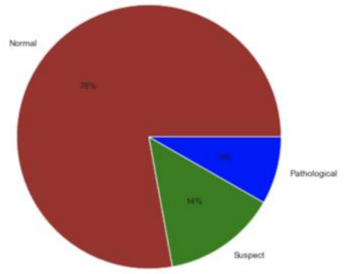
Introduction to Data Science

M.Tech Data Science and Engineering

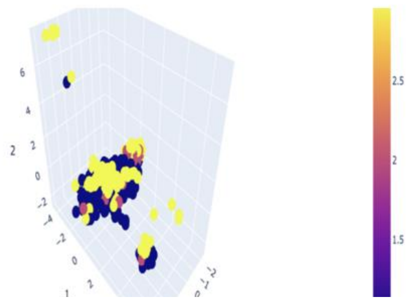
4: Feature Engineering Techniques and Results

Feature engineering plays an important role as everything from the data to the output of the same is dependent on the feature engineering which is being performed. Firstly a pie chart shown is being visualised which shows about the different health types of the fetal and get to know if it is a significant feature or not. While we are getting a correlation matrix of all the features to their significance. The correlation matrix is used in getting the relation of each feature or attribute with itself and also with the other features present out in the dataset.

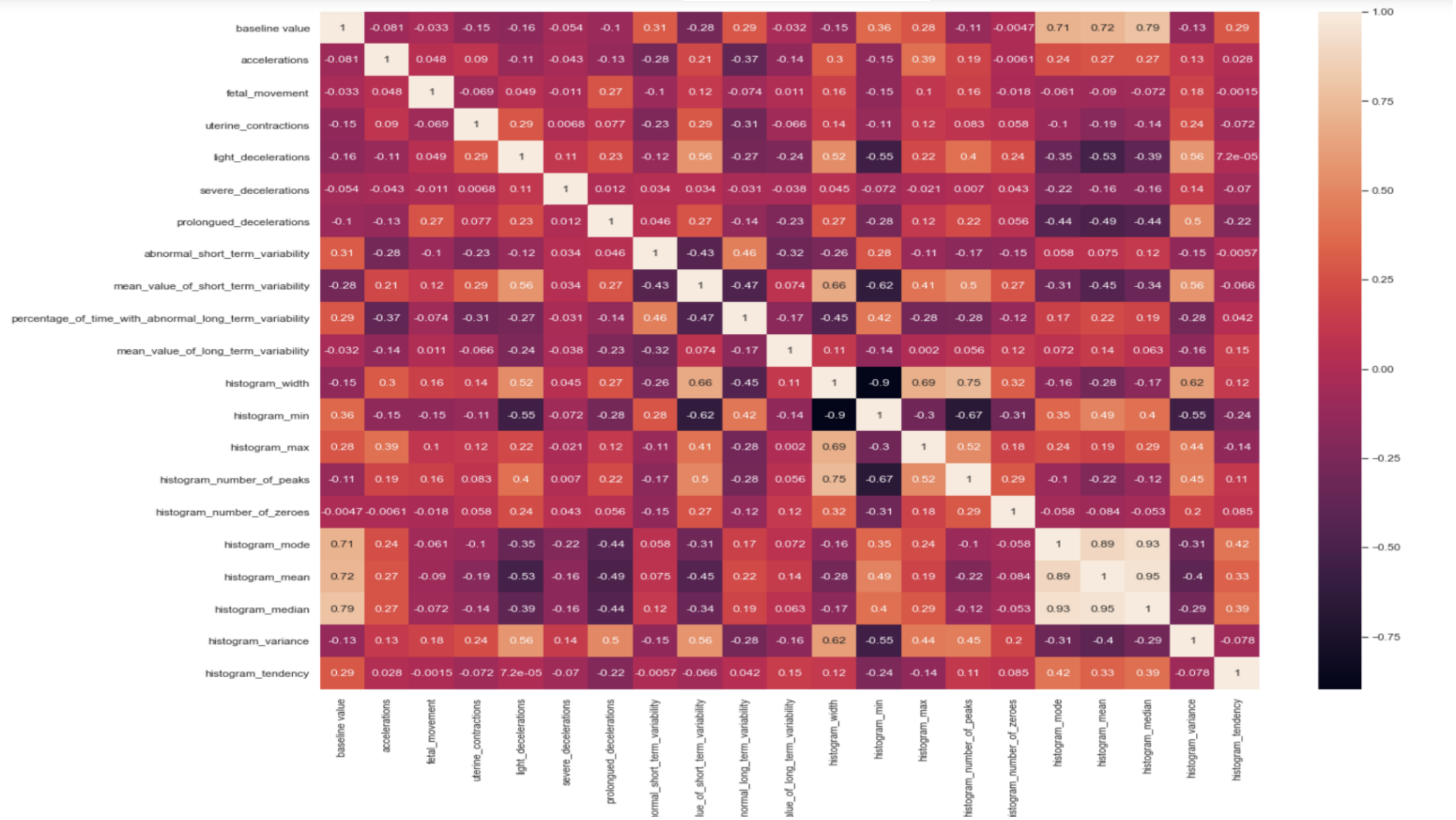
Data Distribution



Dimensionality Reduction and Visualization



Correlation Matrix

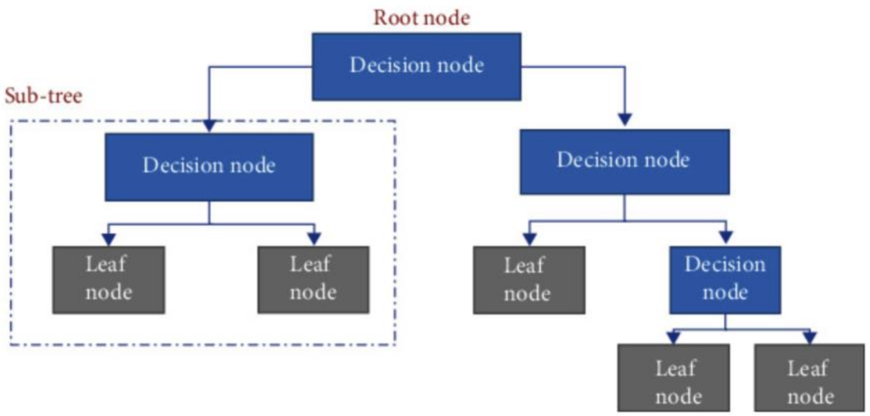


	baseline value	accelerations	fetal_movement	uterine_contractions	light_decelerations	severe_decelerations	prolongued_decelerations	ıormal_short_term_variability	lue_of_short_term_variability	normal_long_term_variability	alue_of_long_term_variability	histogram_width	histogram_min	histogram_max	histogram_number_of_peaks	istogram_number_of_zeroes	histogram_mode	histogram_mean	histogram_median	histogram_variance	histogram_tendency
baseline value	1	-0.081	-0.033	-0.15	-0.16	-0.054	-0.1	0.31	-0.28	0.29	-0.032	-0.15	0.36	0.28	-0.11	-0.0047	0.71	0.72	0.79	-0.13	0.29
accelerations	-0.081	1	0.048	0.09	-0.11	-0.043	-0.13	-0.28	0.21	-0.37	-0.14	0.3	-0.15	0.39	0.19	-0.0061	0.24	0.27	0.27	0.13	0.028
fetal_movement	-0.033	0.048	1	-0.069	0.049	-0.011	0.27	-0.1	0.12	-0.074	0.011	0.16	-0.15	0.1	0.16	-0.018	-0.061	-0.09	-0.072	0.18	-0.0015
uterine_contractions	-0.15	0.09	-0.069	1	0.29	0.0068	0.077	-0.23	0.29	-0.31	-0.066	0.14	-0.11	0.12	0.083	0.058	-0.1	-0.19	-0.14	0.24	-0.072
light_decelerations	-0.16	-0.11	0.049	0.29	1	0.11	0.23	-0.12	0.56	-0.27	-0.24	0.52	-0.55	0.22	0.4	0.24	-0.35	-0.53	-0.39	0.56	7.2e-05
severe_decelerations	-0.054	-0.043	-0.011	0.0068	0.11	1	0.012	0.034	0.034	-0.031	-0.038	0.045	-0.072	-0.021	0.007	0.043	-0.22	-0.16	-0.16	0.14	-0.07
prolongued_decelerations	-0.1	-0.13	0.27	0.077	0.23	0.012	1	0.046	0.27	-0.14	-0.23	0.27	-0.28	0.12	0.22	0.056	-0.44	-0.49	-0.44	0.5	-0.22
abnormal_short_term_variability	0.31	-0.28	-0.1	-0.23	-0.12	0.034	0.046	1	-0.43	0.46	-0.32	-0.26	0.28	-0.11	-0.17	-0.15	0.058	0.075	0.12	-0.15	-0.0057
mean_value_of_short_term_variability	-0.28	0.21	0.12	0.29	0.56	0.034	0.27	-0.43	1	-0.47	0.074	0.66	-0.62	0.41	0.5	0.27	-0.31	-0.45	-0.34	0.56	-0.066
percentage_of_time_with_abnormal_long_term_variability	0.29	-0.37	-0.074	-0.31	-0.27	-0.031	-0.14	0.46	-0.47	1	-0.17	-0.45	0.42	-0.28	-0.28	-0.12	0.17	0.22	0.19	-0.28	0.042
mean_value_of_long_term_variability	-0.032	-0.14	0.011	-0.066	-0.24	-0.038	-0.23	-0.32	0.074	-0.17	1	0.11	-0.14	0.002	0.056	0.12	0.072	0.14	0.063	-0.16	0.15
histogram_width	-0.15	0.3	0.16	0.14	0.52	0.045	0.27	-0.26	0.66	-0.45	0.11	1	-0.9	0.69	0.75	0.32	-0.16	-0.28	-0.17	0.62	0.12
histogram_min	0.36	-0.15	-0.15	-0.11	-0.55	-0.072	-0.28	0.28	-0.62	0.42	-0.14	-0.9	1	-0.3	-0.67	-0.31	0.35	0.49	0.4	-0.55	-0.24
histogram_max	0.28	0.39	0.1	0.12	0.22	-0.021	0.12	-0.11	0.41	-0.28	0.002	0.69	-0.3	1	0.52	0.18	0.24	0.19	0.29	0.44	-0.14
histogram_number_of_peaks	-0.11	0.19	0.16	0.083	0.4	0.007	0.22	-0.17	0.5	-0.28	0.056	0.75	-0.67	0.52	1	0.29	-0.1	-0.22	-0.12	0.45	0.11
histogram_number_of_zeroes	-0.0047	-0.0061	-0.018	0.058	0.24	0.043	0.056	-0.15	0.27	-0.12	0.12	0.32	-0.31	0.18	0.29	1	-0.058	-0.084	-0.053	0.2	0.085
histogram_mode	0.71	0.24	-0.061	-0.1	-0.35	-0.22	-0.44	0.058	-0.31	0.17	0.072	-0.16	0.35	0.24	-0.1	-0.058	1	0.89	0.93	-0.31	0.42
histogram_mean	0.72	0.27	-0.09	-0.19	-0.53	-0.16	-0.49	0.075	-0.45	0.22	0.14	-0.28	0.49	0.19	-0.22	-0.084	0.89	1	0.95	-0.4	0.33
histogram_median	0.79	0.27	-0.072	-0.14	-0.39	-0.16	-0.44	0.12	-0.34	0.19	0.063	-0.17	0.4	0.29	-0.12	-0.053	0.93	0.95	1	-0.29	0.39
histogram_variance	-0.13	0.13	0.18	0.24	0.56	0.14	0.5	-0.15	0.56	-0.28	-0.16	0.62	-0.55	0.44	0.45	0.2	-0.31	-0.4	-0.29	1	-0.078
histogram_tendency	0.29	0.028	-0.0015	-0.072	7.2e-05	-0.07	-0.22	-0.0057	-0.066	0.042	0.15	0.12	-0.24	-0.14	0.11	0.085	0.42	0.33	0.39	-0.078	1

6: Modelling

It depicts the whole decision tree design flowchart. This assignment makes use of a decision tree classifier. This classifier seems to recursively divide the example space. It is a predictive paradigm that acts as a mapping between the characteristics of an item and their values. It regularly splits each potential data result into parts. Each non leaf node corresponds to a feature experiment, each branch to the outcome of the experiment, and each leaf node to a judgment or classification. The root node of the tree, which is at the very top, reflects the most often used prediction model. The decision node and the leaf node are the two nodes in a decision tree. The choice nodes are used to make those selections and have numerous branches, whereas the leaf nodes are the result of those choices and contain no additional branches. The outcomes of the tests or judgments are contingent on the dataset's properties.

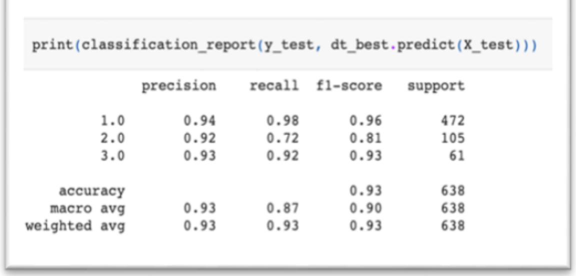
Flowchart of decision tree classifier



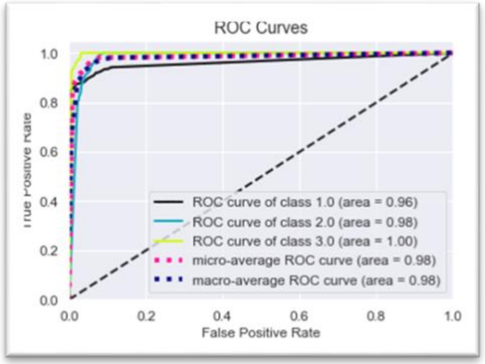
7: Modeling Results and Conclusion

It shows the DT model's classification report. Here, the overall achieved F1-score is 93%. The individual F1-score is 96% for normal, 81% for suspected, and 93% for pathological. It displays the prediction of the DT model. The projected result is displayed in the confusion matrix, as well as the model's computed performance. The total number of correct predictions is 595, with 43 incorrect forecasts.

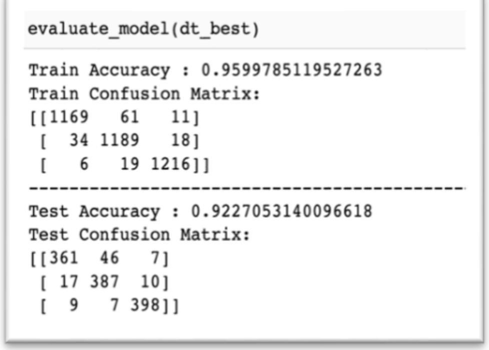
Classification Report



```
print(classification_report(y_test, dt_best.predict(X_test)))

              precision    recall  f1-score   support

         1.0       0.94      0.98      0.96       472
         2.0       0.92      0.72      0.81       105
         3.0       0.93      0.92      0.93        61

    accuracy                           0.93       638
   macro avg       0.93      0.87      0.90       638
weighted avg       0.93      0.93      0.93       638
```



Confusion Matrix



```
evaluate_model(dt_best)

Train Accuracy : 0.9599785119527263
Train Confusion Matrix:
[[1169   61   11]
 [  34 1189   18]
 [   6   19 1216]]
--------------------------------------------
Test Accuracy : 0.9227053140096618
Test Confusion Matrix:
[[361  46   7]
 [ 17 387  10]
 [  9   7 398]]
```

Type I Error: Null hypothesis mistakenly rejected or False Positive Error = 0.054
Type II Error: Null hypothesis mistakenly accepted or False Negative Error = 0.082