Use Of Explainable AI

Problem Statement :

Please find the mentioned below dataset to be used for xAI.

1. Biomedical Features of orthopedic patients

Content

Field Descriptions:

Each patient is represented in the data set by six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine (each one is a column):

1. pelvic incidence
2. pelvic tilt
3. lumbar lordosis angle
4. sacral slope
5. pelvic radius
6. degree of spondylolisthesis

Data Preparation Steps for Model Building:

Please find below the data preparation steps in short. These steps are to be performed for model building. The Explainable AI can be used once the mentioned below steps are completed.

1. Data Preparation.
   a) Reading the Datasets
   b) Checking Null Values
   c) Feature Encoding
   d) Univariate and Multivariate Analysis
   e) Outliers Treatment and Anomaly Detection
   f) Scaling and Normalization
   g) Correlations and Causations
   h) Feature Selection and Information Gain

2. Model Building.
   a) Cross Validation and Data Pipelining
   b) Model Creation and Model Accuracy

Use Of Explainable AI :

For any dataset that we considered, the idea of using predictive analysis is to build a model and evaluation of the model with certain metrics. The Model can be classified into 2 basic categories in terms of XAI

a. Glass Box
b. BlackBox

The **Glass Box** constitutes all the model based on Linear and Tree Based Algorithms.

The **Black Box** constitutes the Neural Network – ANN, CNN and RNN.

The traditional AI is non – explanatory in nature as it is supposed to build a model based on algorithm suitable for a problem statement and perform the evaluation after the model is build.

Explainable AI – xAI can perform all the required steps in correct fashion by not only building the model but also explaining the model and it's feature performances even before the result is produced.

For all the independent columns that we are passing in the model, xAI can successfully identify the features influence over the desired result and correcting the parameters to be predicted as inappropriate for the dataset that we are using.

EXPLAINABLE AI

Explainable AI is a set of tools and frameworks to help you understand and interpret predictions made by your machine learning models, natively integrated with a number of Google's products and services. With it, you can debug and improve model performance, and help others understand your models' behavior.

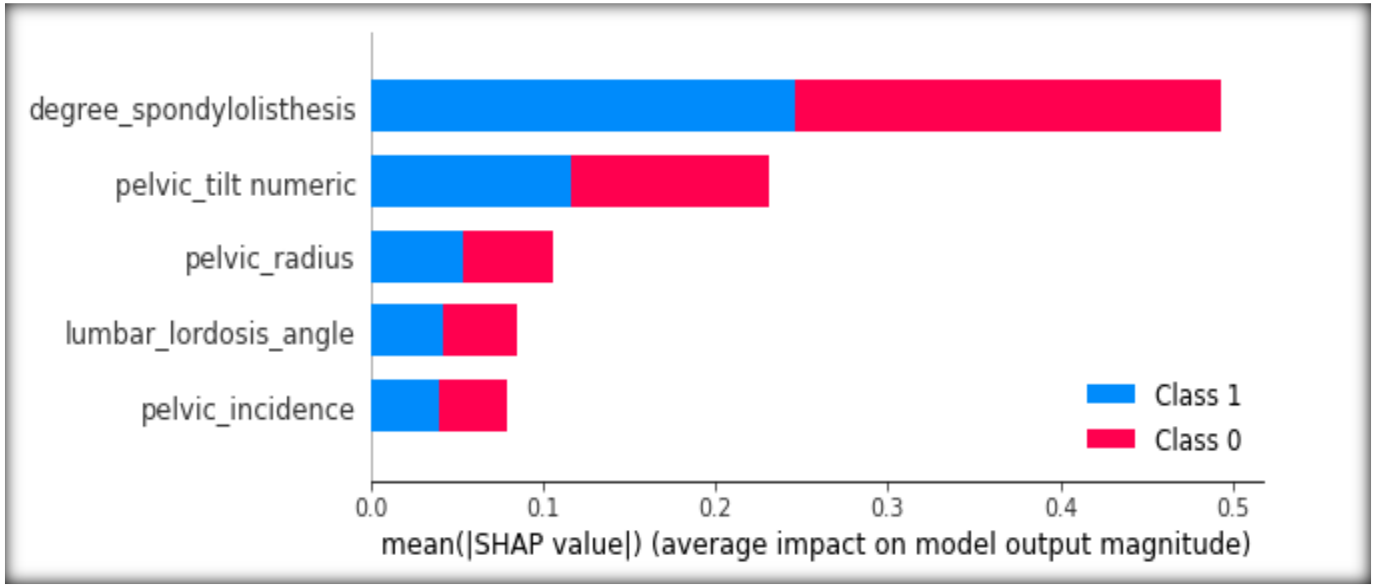
Few of the Libraries that can be used includes,

1. LIME
2. YELLOWBRICKS
3. SHAP
4. Eli5
5. Interpret ML

Use Of The SHAP Libraries :

SHAP Summary Plot

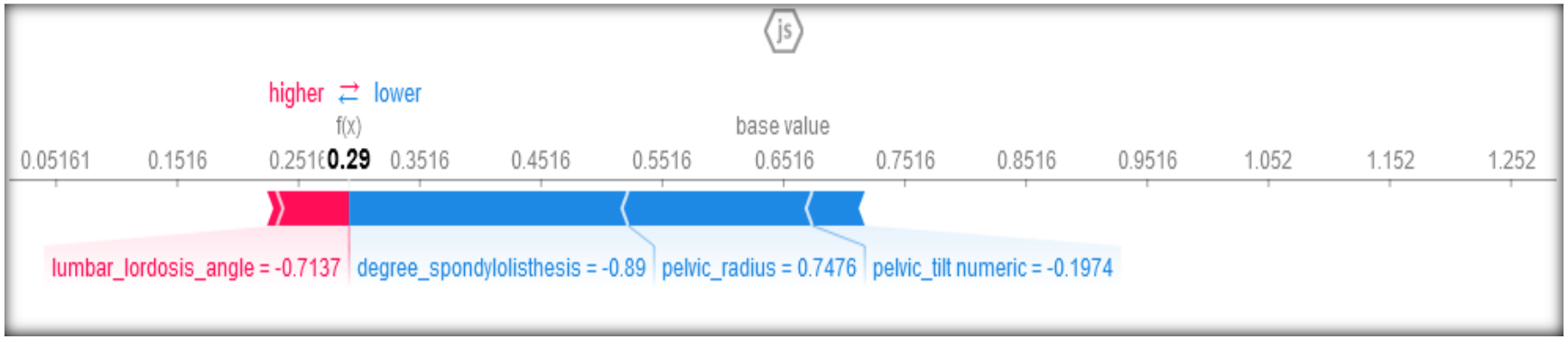
Summary plots are easy-to-read visualizations which bring the whole data to a single plot. All of the features are listed in y-axis in the rank order, the top one being the most contributor to the predictions and the bottom one being the least or zero-contributor. Shap values are provided in the x-axis. As we discussed already, a value of zero represents no contribution whereas contributions increase as the shap value moves away from zero. Each circular dot in the plot represents a single data point. Color of the dot denotes the value of that corresponding feature. It can be observed that the feature 'worst perimeter' contributes greatly to the model's prediction with low values deciding one class and higher values deciding the other.



SHAP Force Plot

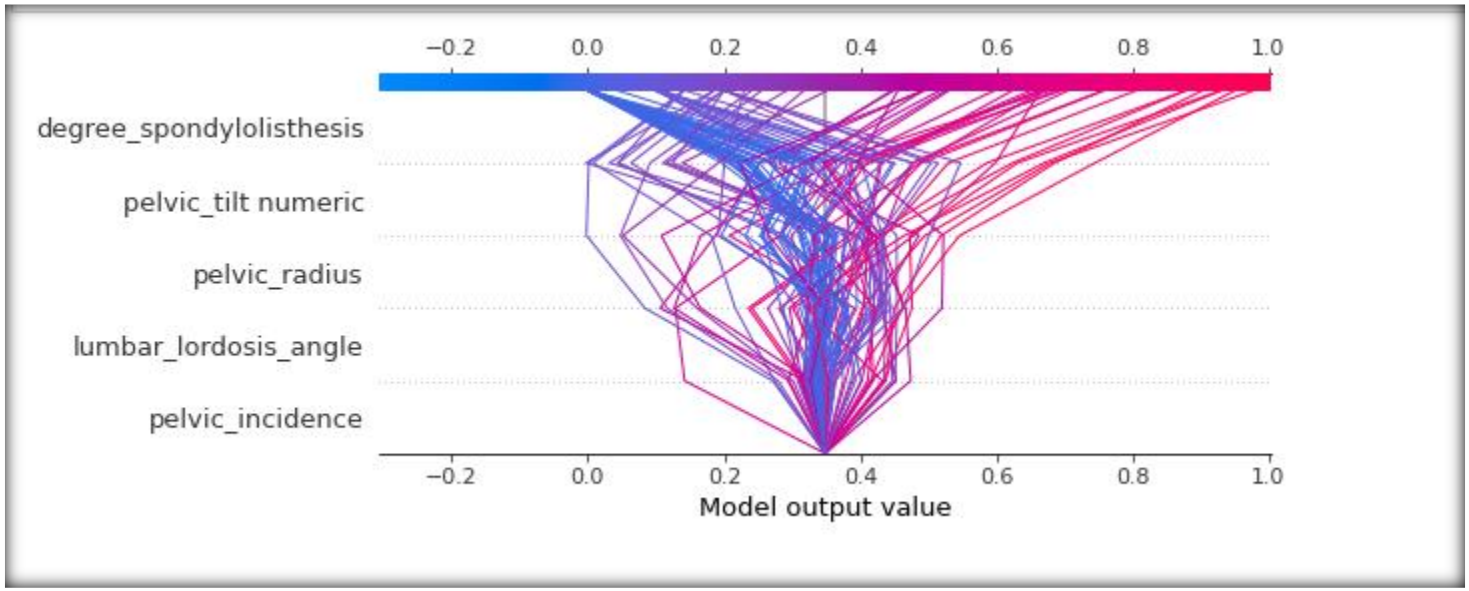
Develop a tree-based SHAP explainer and calculate the shap values. Shap values are arrays of a length corresponding to the number of classes in target. Here the problem is binary classification, and thus shap values have two arrays corresponding to either class.

Shap values are floating-point numbers corresponding to data in each row corresponding to each feature. Shap value represents the contribution of that particular data point in predicting the outputs. If the shap value is much closer to zero, we can say that the data point contributes very little to predictions. If the shap value is a strong positive or strong negative value, we can say that the data point greatly contributes to predicting the positive or negative class.
Force plots are suitable for row-wise SHAP analysis. It takes in a single row and shows in a rank order how each of the features contributed to the prediction. Wider a feature's block, more the contribution.



SHAP Decision Plot

Finally, we discuss the decision plot. As the summary plot, it gives an overall picture of contribution to prediction. From bottom to top of the decision plot, shap values are cumulatively added to the base value of the model in determining the output values. It can be observed that certain strings colored in blue resulted in final class value 0 and the remaining strings colored in red resulted in final class value 1.



Use Of ELI5 Library:

ELI5 is an acronym for **'Explain like I am a 5-year old'**. This aptly named Python library has the functionality to explain most machine learning models. Interpreting a machine learning model has two main ways of looking at it:

Global Interpretation: Look at a model's parameters and figure out at a global level how the model works Local Interpretation: Look at a single prediction and identify features leading to that prediction

```
import eli5
eli5.show_weights(rf) # clf is the model fitted
```

Weight	Feature
0.4268 ± 0.2188	x4
0.1922 ± 0.1582	x1
0.1400 ± 0.1485	x3
0.1226 ± 0.1669	x0
0.1184 ± 0.0677	x2

```
eli5.show_prediction(rf, X_train.iloc[1], feature_names = list(X_train.columns))
```

y=1.0 (probability **0.954**) top features

Contribution?	Feature
+0.348	<BIAS>
+0.281	degree_spondylolisthesis
+0.123	pelvic_tilt numeric
+0.101	pelvic_radius
+0.073	lumbar_lordosis_angle
+0.028	pelvic_incidence

ML- DASHBOARD

Interpret-ml - Explain Machine Learning Models And Their Predictions

The interpretation of machine learning models and their predictions has become quite important lately. The interpretation of models to better understand which features actually contributed to a particular prediction gives

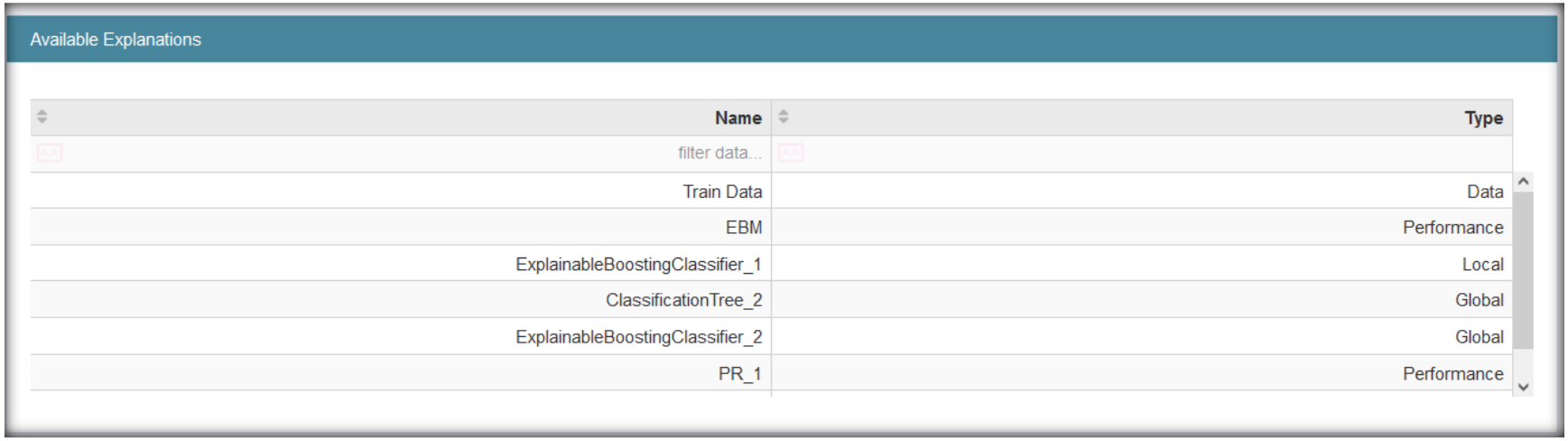
confidence to the model creator. It also helps to better explain why the model is behaving in a particular way. The python has many libraries (like **lime, shap, eli5, yellowbrick** etc.) which provide different ways to explain model predictions. We have already created tutorials on these libraries (See References section). As a part of this tutorial, we'll be explaining the library named interpret-ml which is designed by the Microsoft research team. The interpret-ml is an open-source library and is built on a bunch of other libraries (**plotly, dash, shap, lime, tree interpreter, sklearn, joblib, jupyter, salib, skope-rules, gevent, and pytest**). The interpret-ml creates an interactive dashboard of visualizations using plotly and dash which can explain data, model performance, and predictions from different perspectives. It has divided different explainer classes used for different purposes in different modules. Please make a note that interpret-ml is still in alpha release and constantly getting developed as of the date when this tutorial is created.

We'll be explaining the usage of three main modules as a part of this tutorial.

data - It has classes that can help us explain our dataset from a different perspective.
glass box - It has classes that can help us explain our model predictions.
perf - It has classes that can help us visualize metrics like proc curve, precision-recall curve, etc.

Explaining the Results Using Interpret ML

Available Explanations :

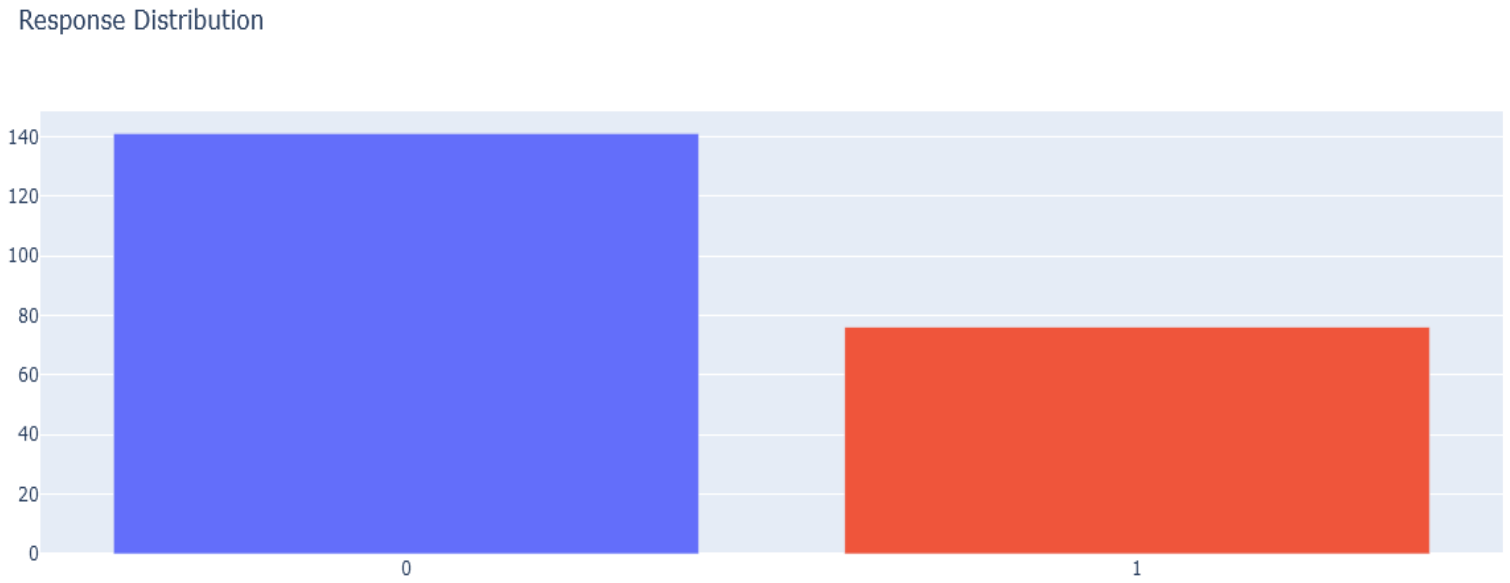


ML Dashboard

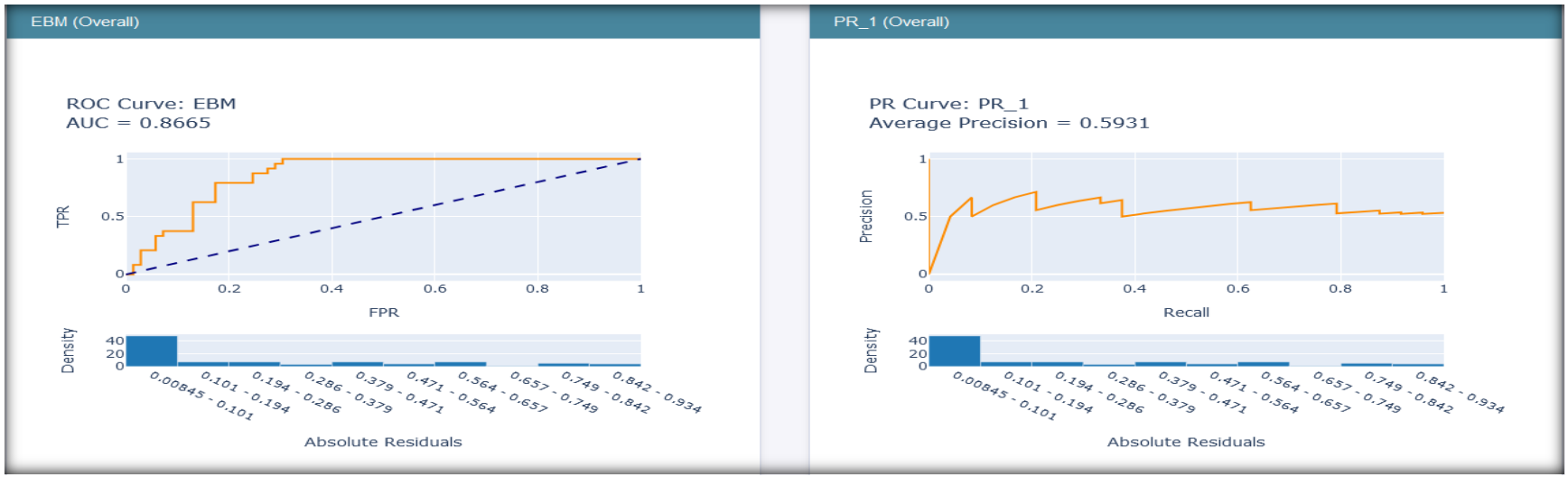
The explanations available are split into tabs, each covering an aspect of the pipeline.

- **Data** covers exploratory data analysis, designed mostly for feature-level.
- **Performance** covers model performance both overall and user-defined groups.
- **Global** explains model decisions overall.
- **Local** explains a model decision for every instance/observation.

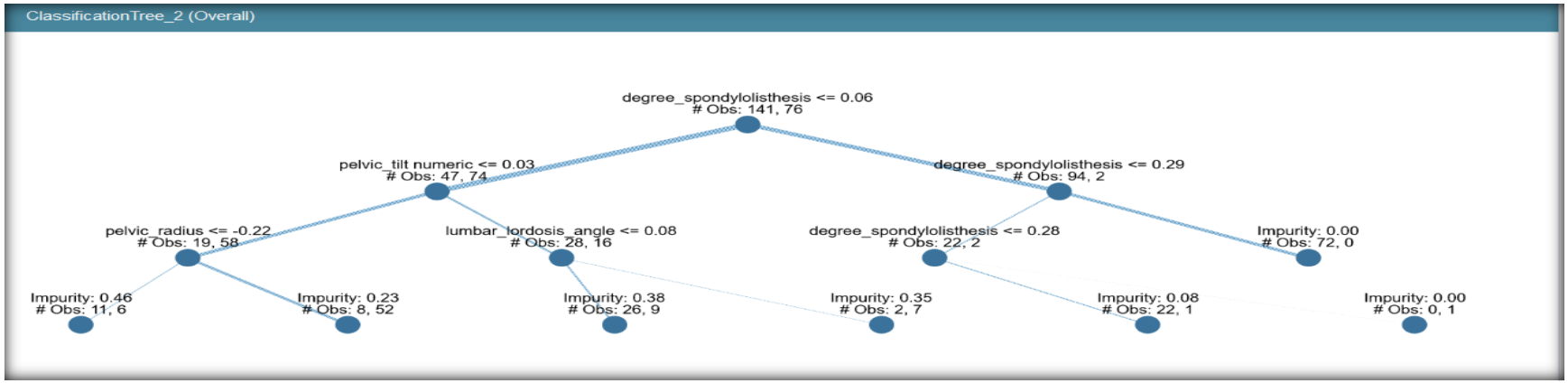
Train Data (Overall)

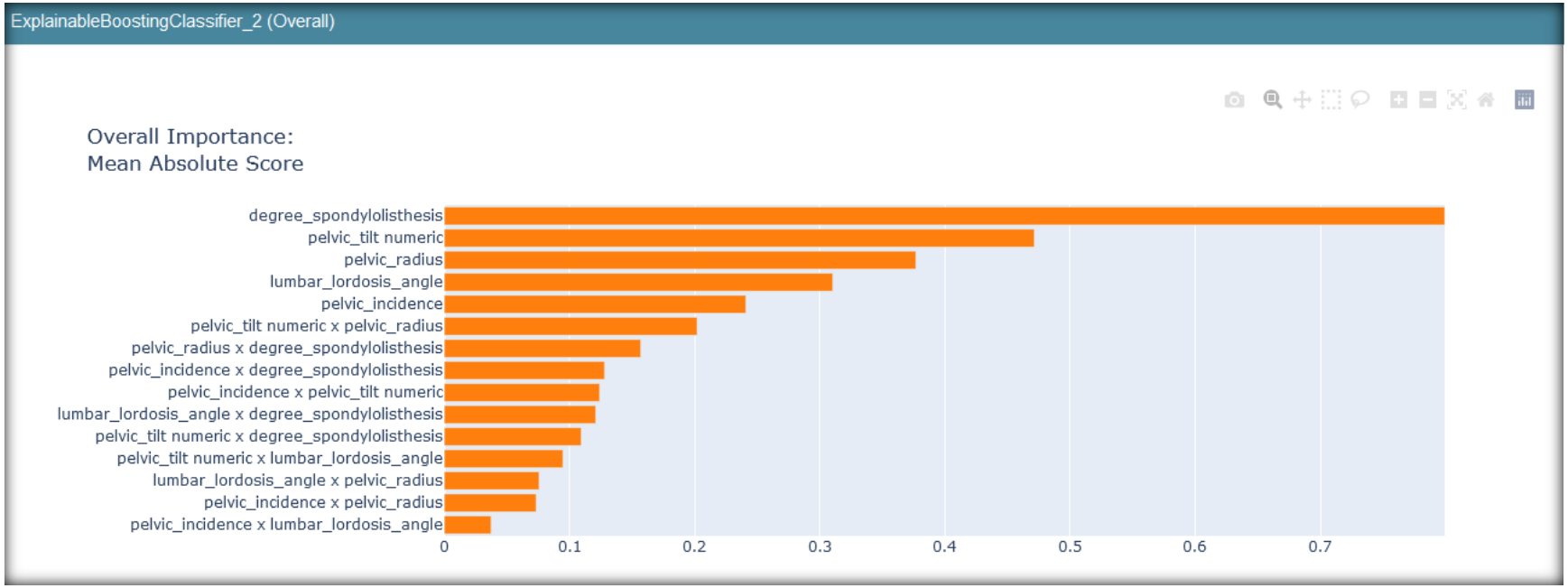


Performance Tab In ML Dashboard:



Global Tab in ML Dashboard:





Conclusion and Expected Outcomes

xAI can able to successfully interpret the Model performance and influence of feature variables in predicting the desired results.