DMG ASSIGNMENT-1

Jatin Tyagi(2020381)
Pratyush Jain(2020396)
Ritvik Pendyala(2020096)
Vatsal Lakhmani(2020148)
Yatish Garg(2020162)

AIM

In this assignment, we aimed to modify and train a Decision Tree Classifier using:

- Single Attribute
- Double Attribute

We had to classify and separate three datasets based on the best feature we'd be getting on applying Logistic Regression and computing the feature with the least Entropy value, and we had to do this recursively.

Procedure

The steps that we followed are as below:

- Preprocessing data
- Modifying the Decision Tree
- Logistic Regression for Feature Selection
- Training and Fitting of the Model
- Cross Validation
- Computing the accuracy
- Compute and report the statistics.

DataSet - 1: Analysis

Cancer Prediction Dataset

This was basically a dataset where we had several features which pointed out to if a person had Cancer or not. The target column in this dataset is "Biopsy" There were about 800 change records, out of which we had 55 records with one and the rest with 0 as their result.

In this dataset we observed there were quite a few missing values.

We had to preprocess all the data present in the dataset and to achieve that. We did the following:

- We found that there were two columns(STDs: Time since first diagnosis, STDs: Time since last diagnosis) in the dataset that had over 90% missing values, and we thought that these two columns didn't contribute to the results as it was useless data, so we ended up dropping these two columns.
- In addition to the above, we tried to fill in the missing values with the mean of the columns, and a couple of variations, we tried to train our modded Decision Tree models with the various variations we decided to make.

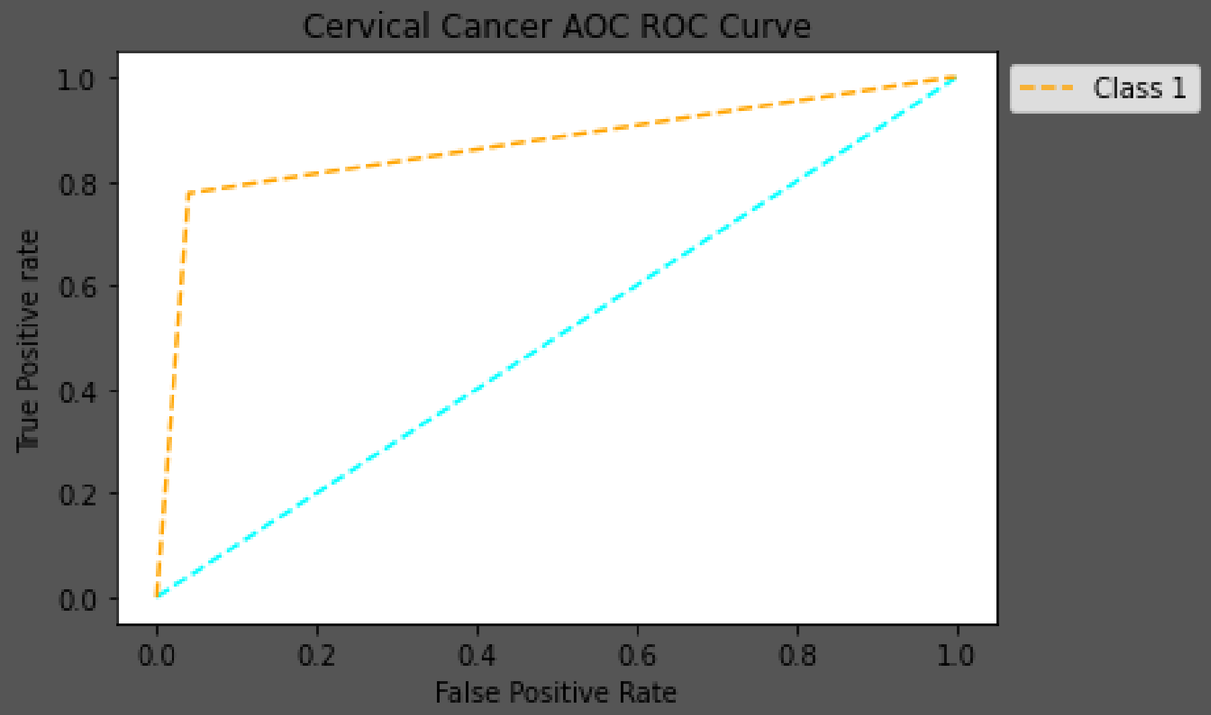
Results(single attribute)

	precision	recall	f1-score	support
0	0.98	0.96	0.97	125
1	0.58	0.78	0.67	9
accuracy			0.95	134
macro avg	0.78	0.87	0.82	134
weighted avg	0.96	0.95	0.95	134

Accuracy: 0.9477611940298507

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



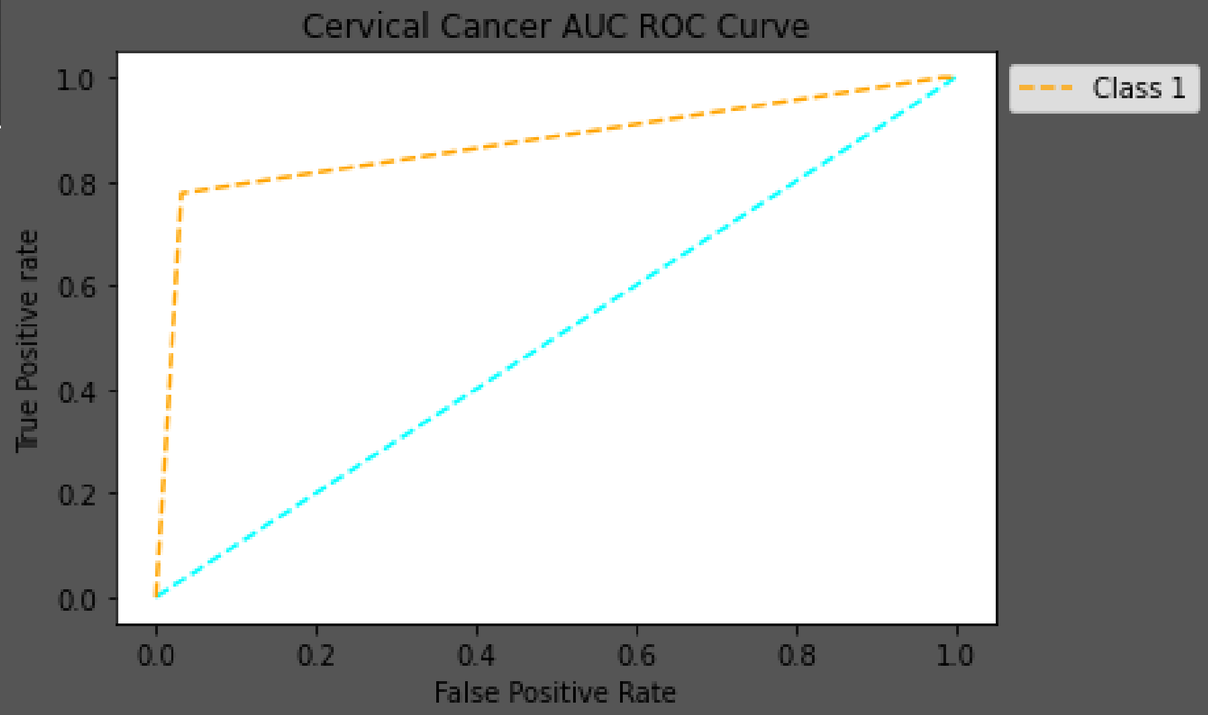
Results(double attribute)

	precision	recall	f1-score	support
0	0.98	0.97	0.98	125
1	0.64	0.78	0.70	9
accuracy			0.96	134
macro avg	0.81	0.87	0.84	134
weighted avg	0.96	0.96	0.96	134

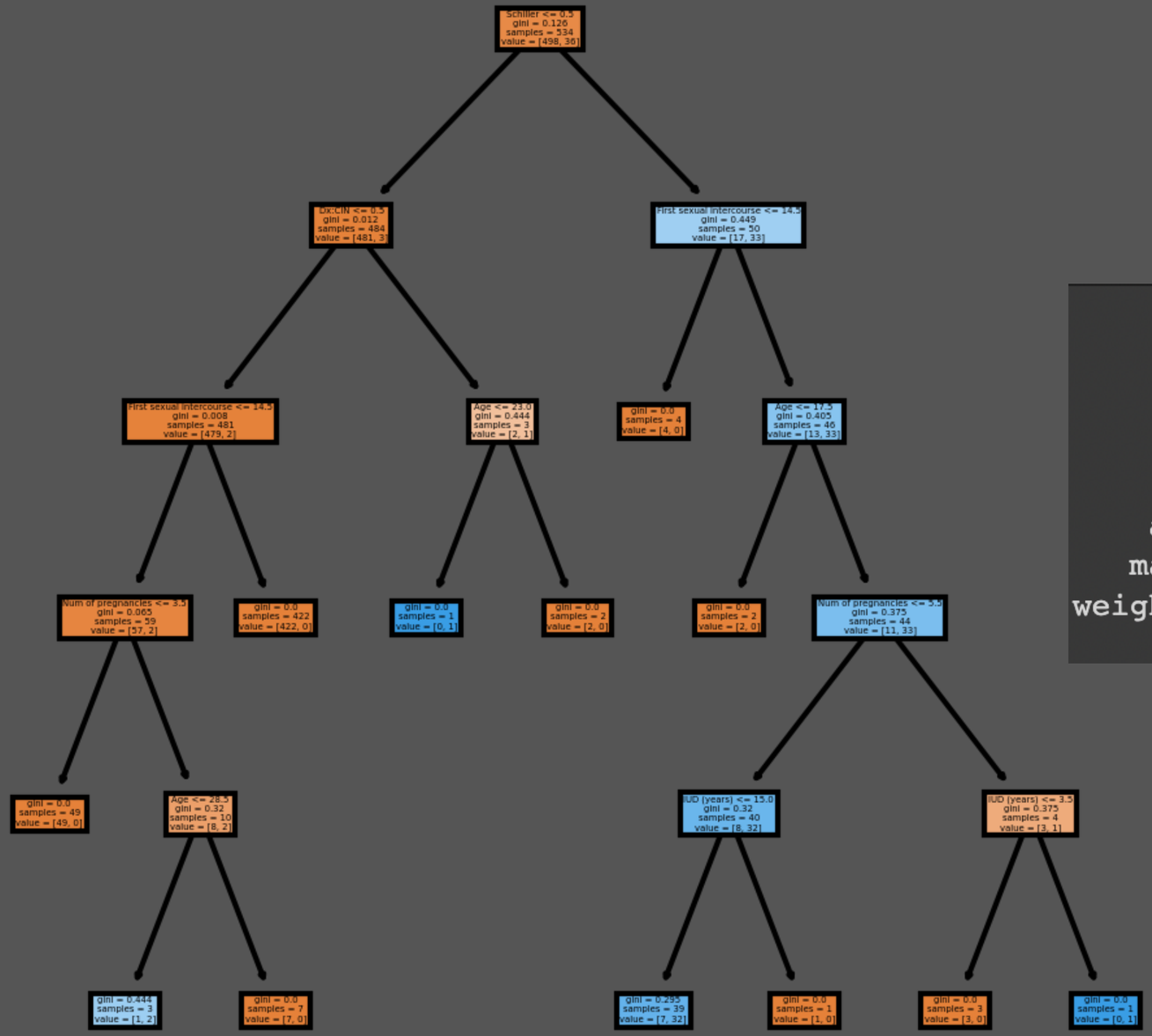
Accuracy: 0.9552238805970149

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



Scikit Learn Decision Tree

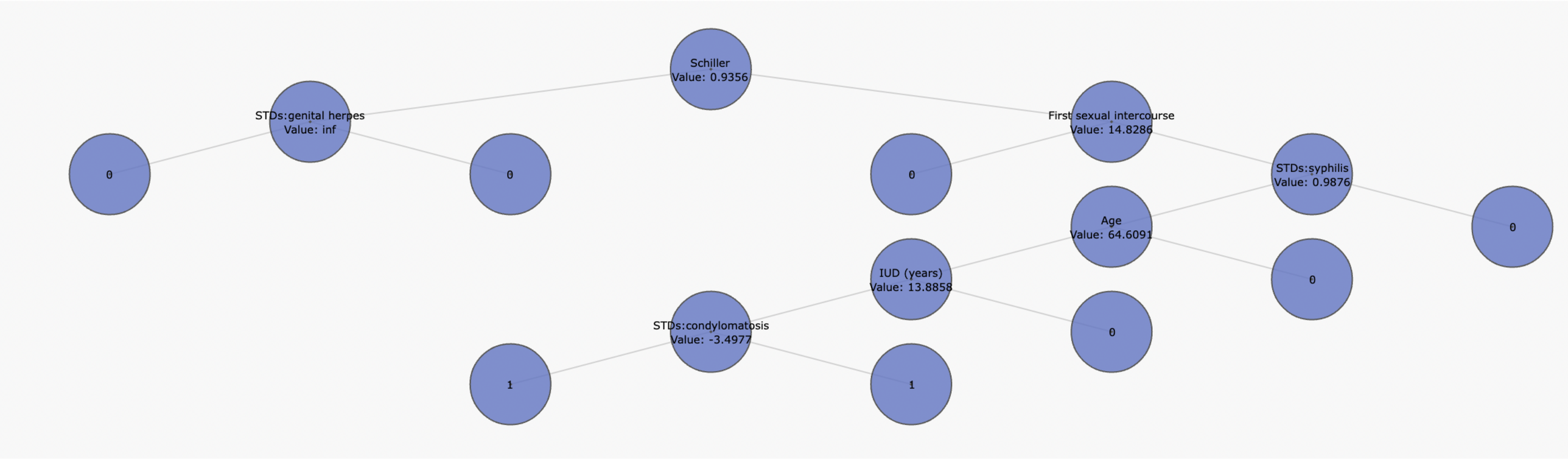


Statistics:

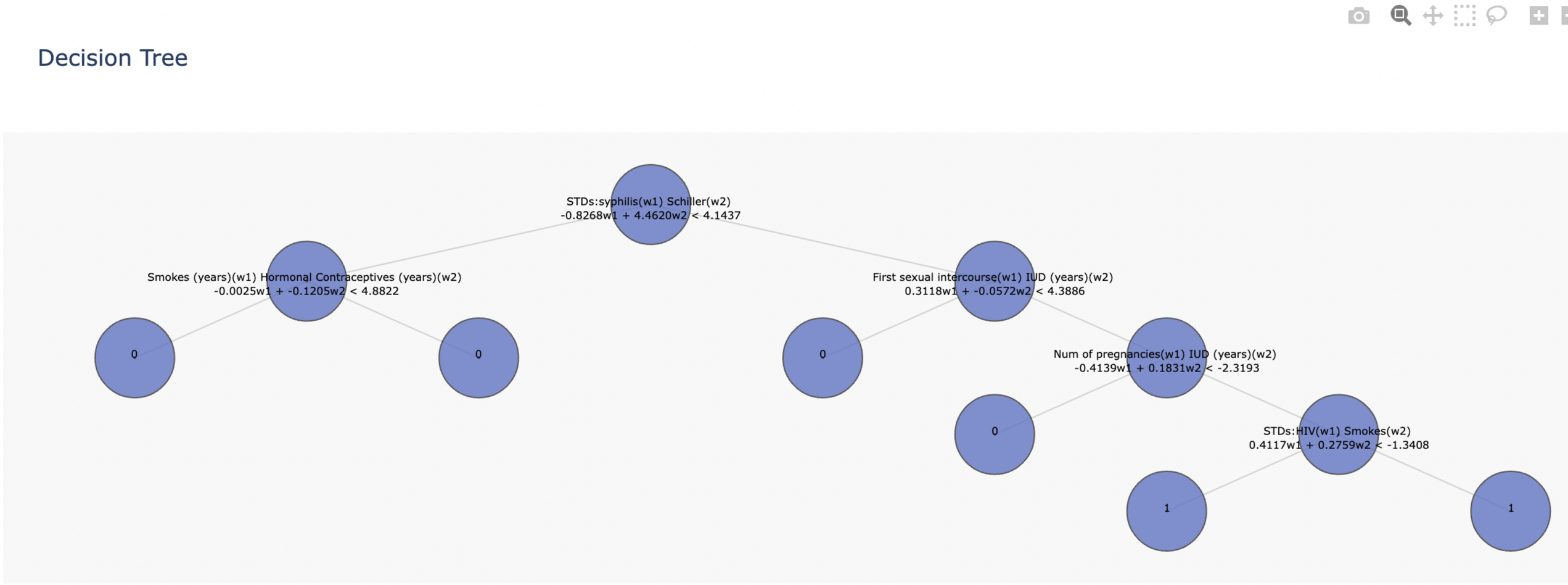
	precision	recall	f1-score	support
0	0.98	0.98	0.98	125
1	0.67	0.67	0.67	9
accuracy			0.96	134
macro avg	0.82	0.82	0.82	134
weighted avg	0.96	0.96	0.96	134

Our Decision Tree(single attribute)

Tree with Reingold-Tilford Layout



Our Decision Tree(double attribute)



K-Fold Cross Validation

Cervical Cancer Risk
1 Attribute for Split
Accuracy Score: 0.9593984962406015
Recall Score: 0.9087985898995594
F1 Score Score: 0.8448614225516037
Precision Score: 0.8112518450202977
2 Attibutes for Split
Accuracy Score: 0.9624060150375939
Recall Score: 0.9045486116292419
F1 Score Score: 0.8548667949434670
Precision Score: 0.8295538461538461

Student T-tests

Paired t-test Resampled
t statistic: -1.707212015047226, p-value: 1.901530479594624

One Attribute Split Accuracy: 0.9532338308457713
Two Attributes Split Accuracy: 0.9582089552238806
The models will work in the same way. (H0 True)

DataSet - 2: Analysis

Fetal Health Dataset

This dataset is all about determining a fetus's fetal health.
The target column of this dataset is "fetal_health."
The dataset was pretty well refined and needed no extra data preprocessing. Although we're given two datasets, they're identical in nature, so we ended up combining them and using them as one whole dataset.

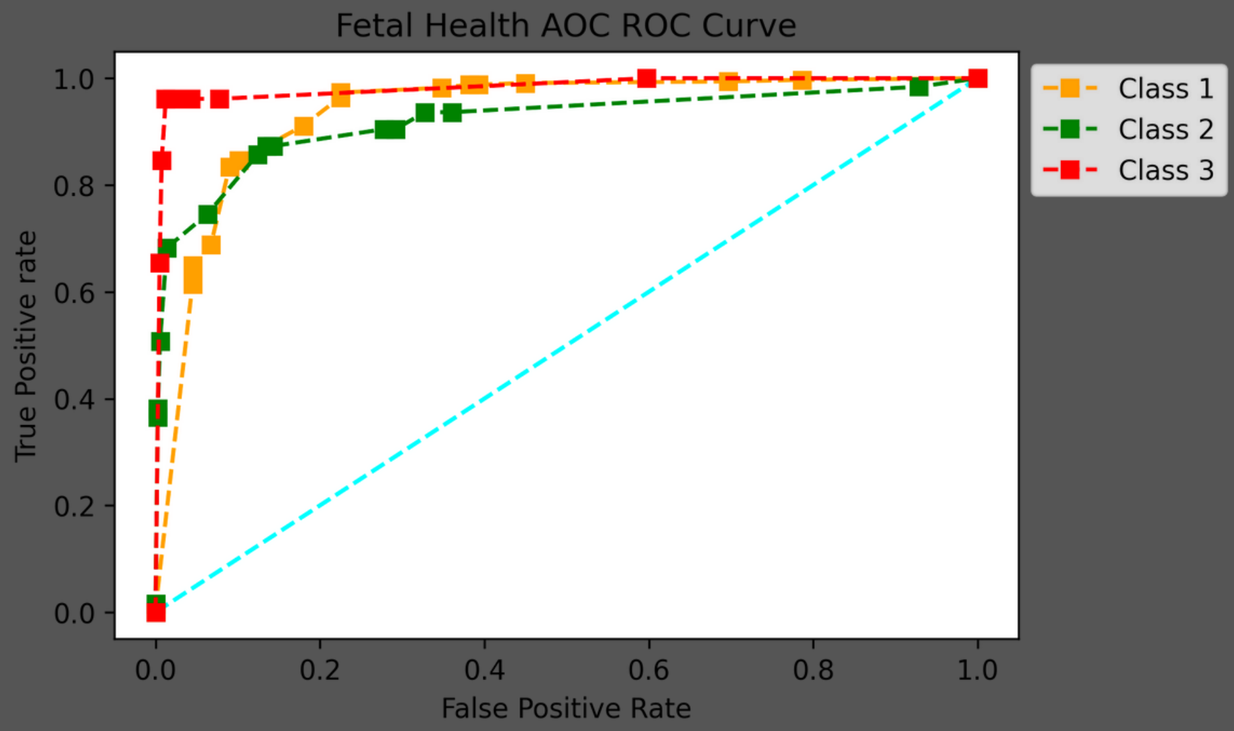
Results(single attribute)

	precision	recall	f1-score	support
1	0.94	0.97	0.96	337
2	0.90	0.68	0.77	63
3	0.83	0.96	0.89	26
accuracy			0.93	426
macro avg	0.89	0.87	0.88	426
weighted avg	0.93	0.93	0.93	426

Accuracy: 0.9295774647887324

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



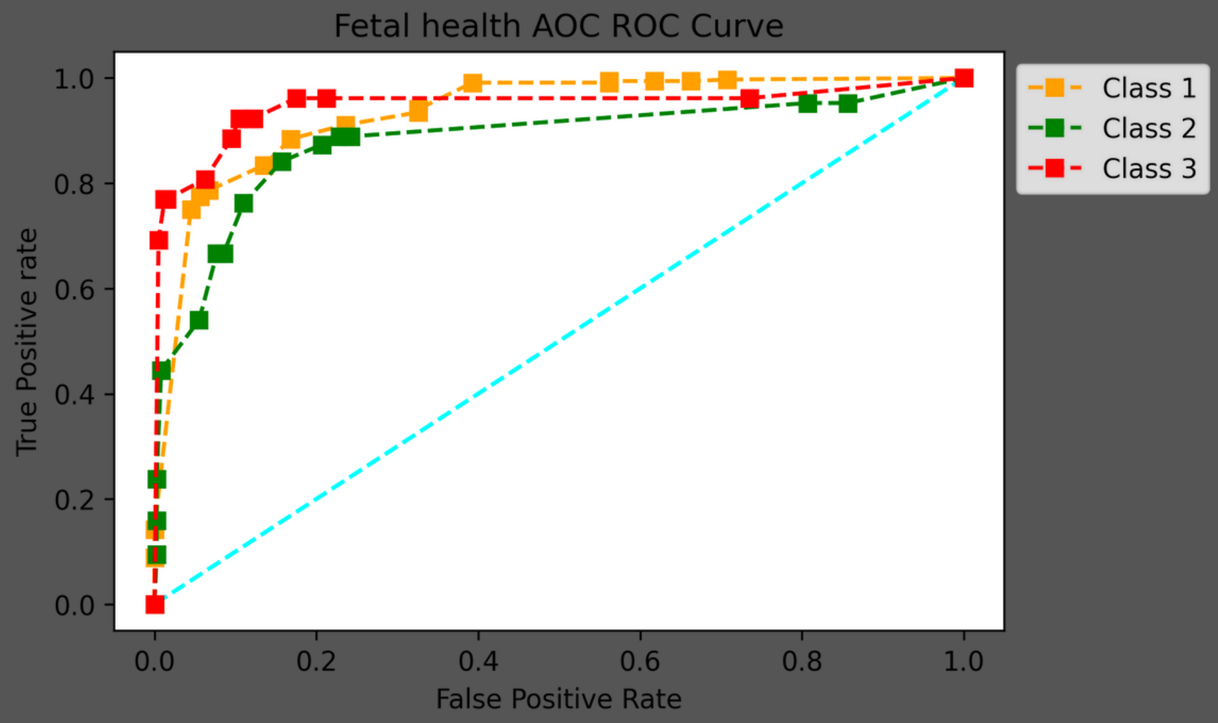
Results(double attribute)

	precision	recall	f1-score	support
1	0.85	0.99	0.92	337
2	0.67	0.19	0.30	63
3	0.94	0.62	0.74	26
accuracy			0.85	426
macro avg	0.82	0.60	0.65	426
weighted avg	0.83	0.85	0.82	426

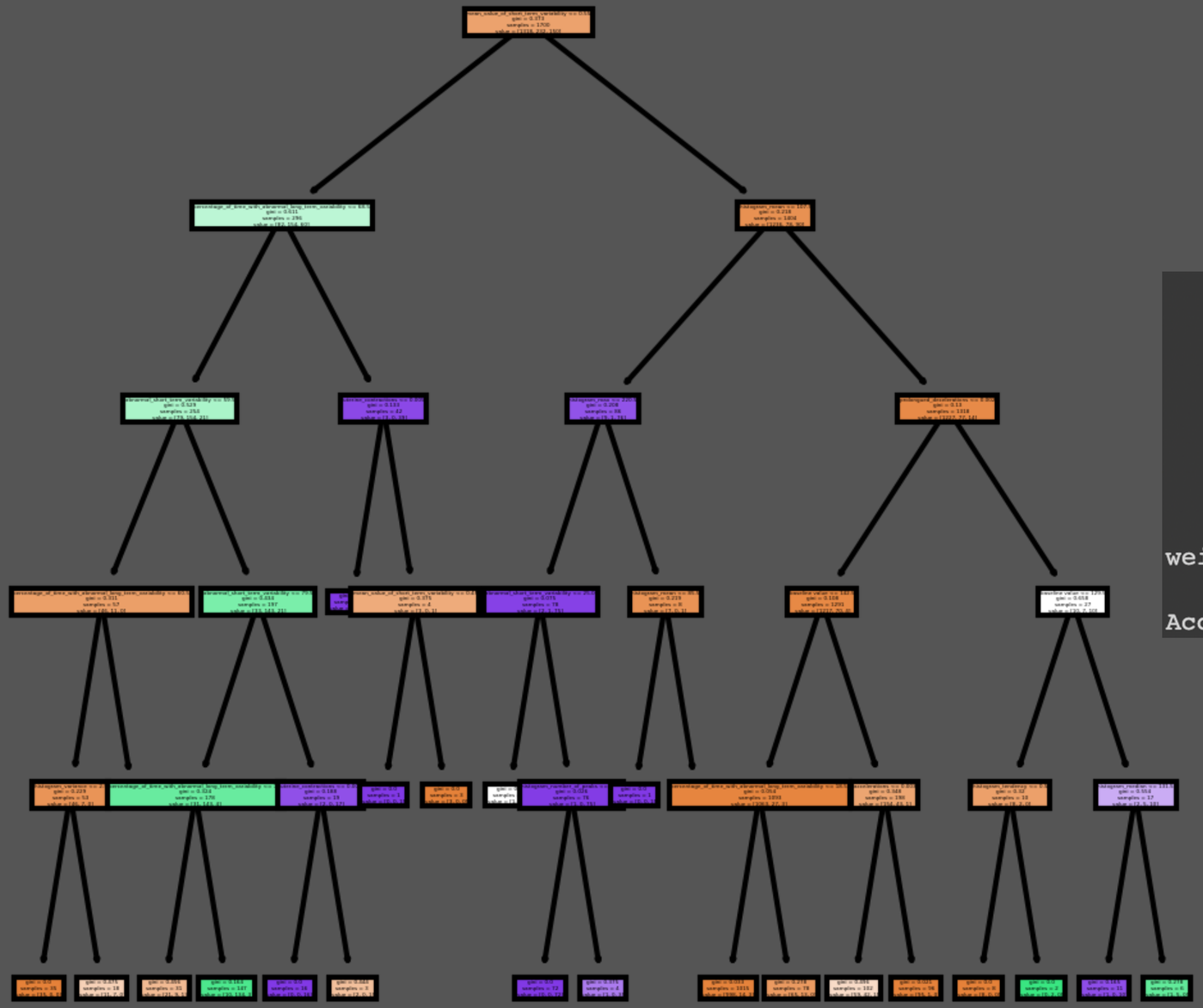
Accuracy: 0.8497652582159625

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



Scikit Learn Decision Tree



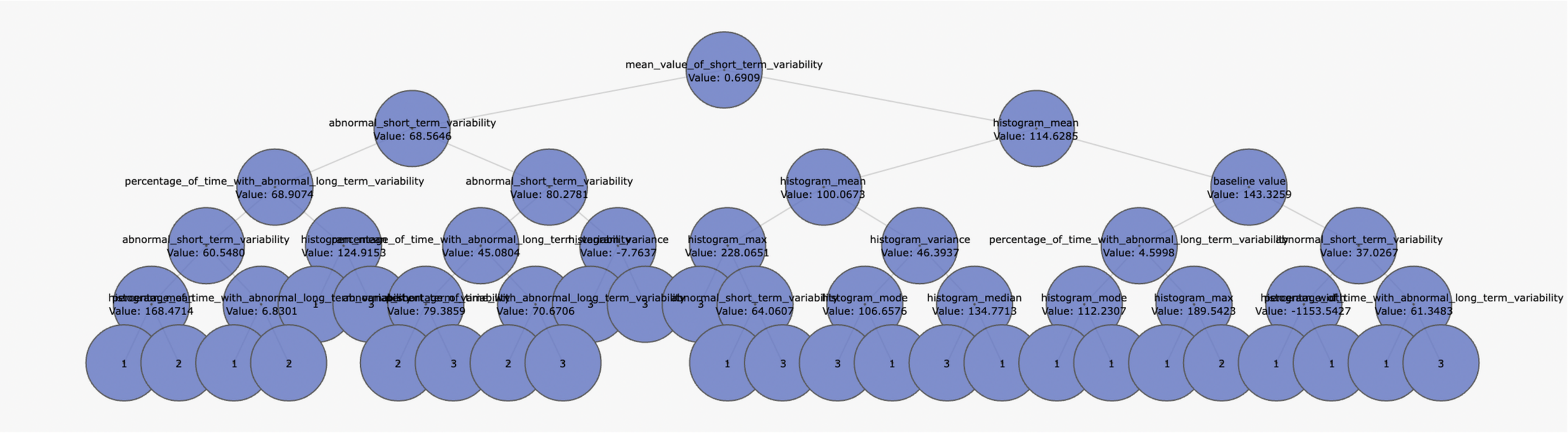
Statistics:

	precision	recall	f1-score	support
1	0.94	0.98	0.96	337
2	0.88	0.67	0.76	63
3	0.93	1.00	0.96	26
accuracy			0.93	426
macro avg	0.91	0.88	0.89	426
weighted avg	0.93	0.93	0.93	426

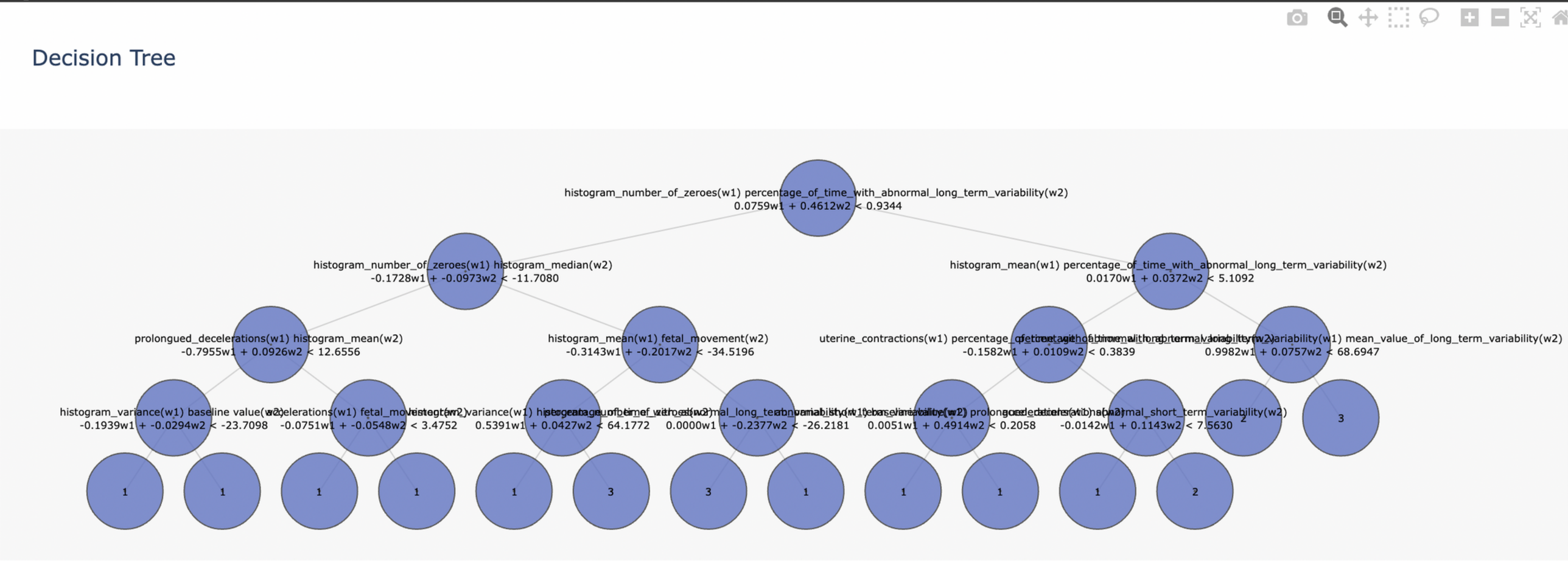
Accuracy: 0.9319248826291080

Our Decision Tree(single attribute)

Tree with Reingold-Tilford Layout



Our Decision Tree(double attribute)



K-Fold Cross Validation

Fetal Health
1 Attribute for Split
Accuracy Score:
0.9593984962406015
Recall Score: 0.9087985898995594
Fl Score Score: 0.844861422551603 7
Precision Score: 0.811251845020297 7
2 Attibutes for Split
Accuracy Score:
0.9624060150375939
Recall Score: 0.9045486116292419
F1 Score Score: 0.8548667949434670
Precision Score: 0.829553846153846 1

Statistics(Student T- tests)

Resampled T-test

```
Paired t-test Resampled
t statistic: 0.9595581490997213, p-value: 0.3452101059323076

One Attribute Split Accuracy: 0.8717527386541472
Two Attributes Split Accuracy: 0.8658841940532083
The models will work in the same way. (H0 True)
```

Statistics(Student T- tests)

Cross Validated Pair T-test

```
Cross Validated Paired t-test
t statistic: 0.3591527515448929, p-value: 0.727762651433001

One Attribute Split Accuracy: 0.8725152803614138
Two Attributes Split Accuracy: 0.8706462042696430
The models will work in the same way. (H0 True)
```

DataSet - 3: Analysis

Banking Dataset

This dataset was all about the database a bank has.
The target column is the last column "y".
Here in this dataset there was a lot of preprocessing we had to perform such that we could use its data for our classification and splitting.

Data Preprocessing Steps:

The steps we used to make the data feasible for our use are as below:

- In the education feature, there are eight different types of fields, and they could be used as a major classification metric in the dataset, but when they're in text format, we can't make use of them. That's why we've got to give them a ranking base system where you can signify the rating of the value to be how far they've been educated. We used a -1 to 6 ranking system where -1 meant illiterate, and 6 meant university degree.
- A similar issue occurred in the columns "married", "poutcome", "contact", "loan", "housing" and "default"
- Also, there was a non-quantifiable column where we couldn't assign any hierarchy to them; in this case, we used One hot encoding where we would just make those number of columns = unique number of values in the column.. This would essentially increase the number of columns where each would have a 1 or 0, signifying whether that person has that job
- In addition to that, we have unknown tags in some of the features in the dataset, here we have to examine and correspondingly decide whether they'd be missing values or if they'd be actual meaningful data.

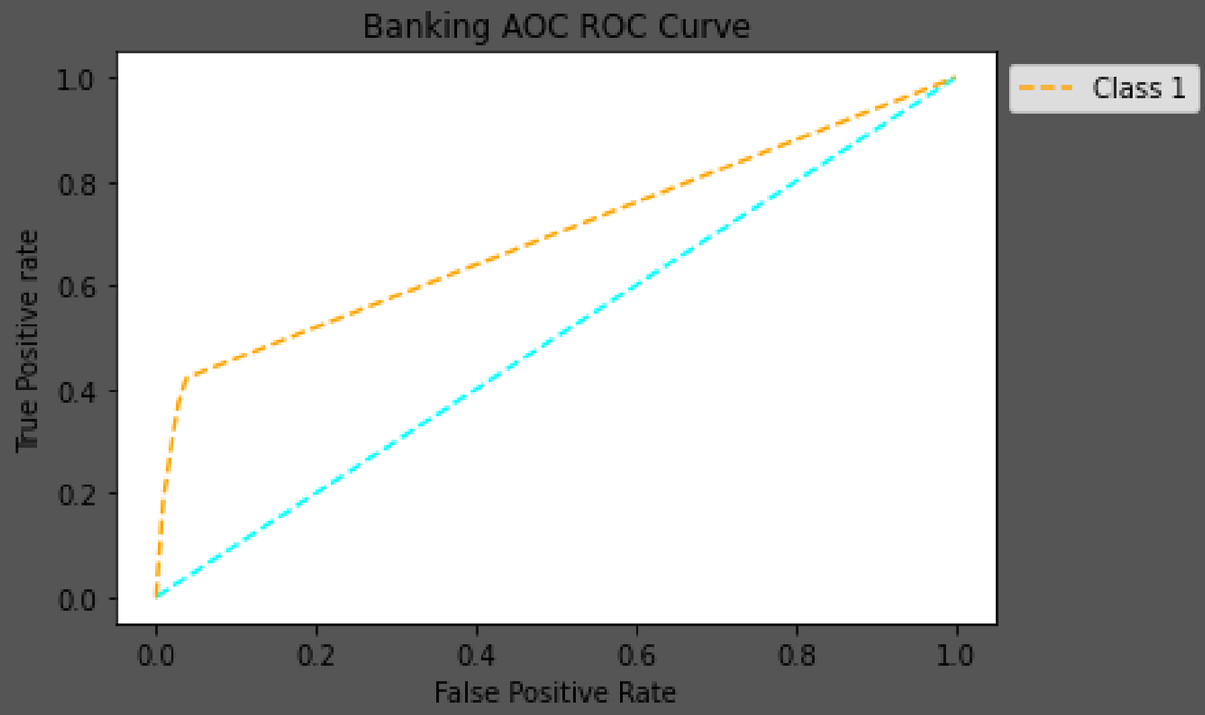
Results(single attribute)

	precision	recall	f1-score	support
0	0.92	0.97	0.95	7301
1	0.63	0.37	0.46	937
accuracy			0.90	8238
macro avg	0.78	0.67	0.71	8238
weighted avg	0.89	0.90	0.89	8238

Accuracy: 0.9033746054867686

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



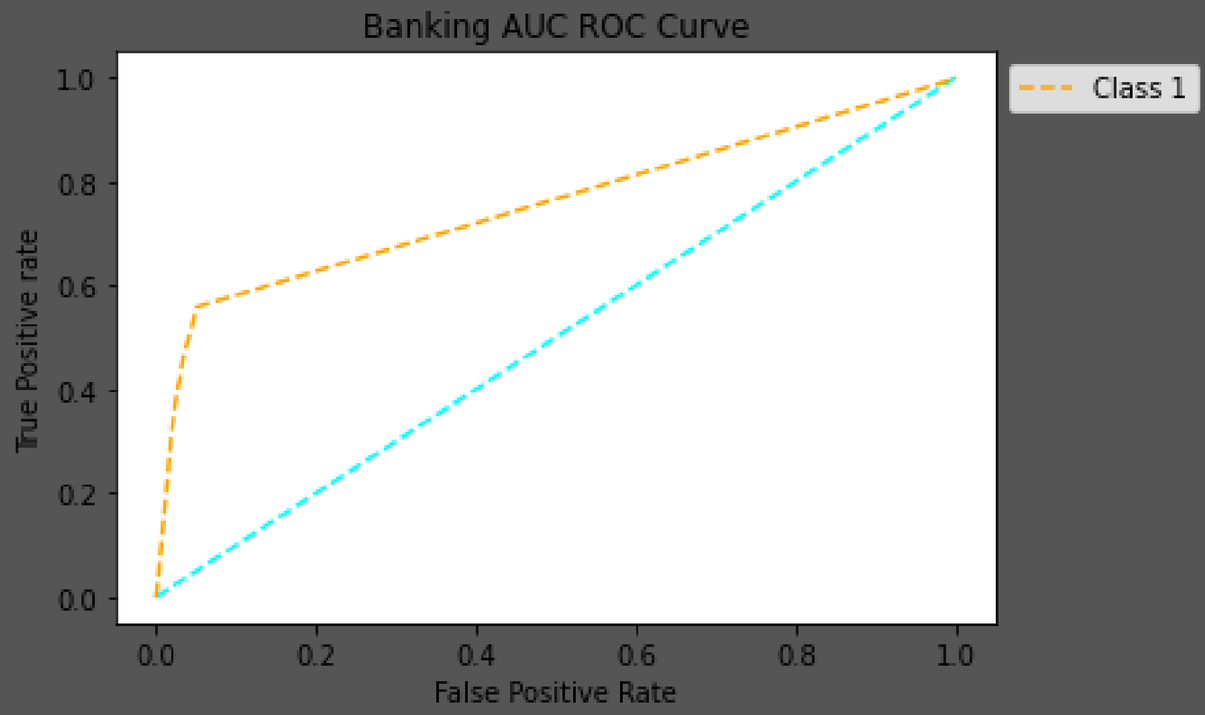
Results(double attribute)

	precision	recall	f1-score	support
0	0.93	0.97	0.95	7301
1	0.64	0.42	0.51	937
accuracy			0.91	8238
macro avg	0.78	0.69	0.73	8238
weighted avg	0.90	0.91	0.90	8238

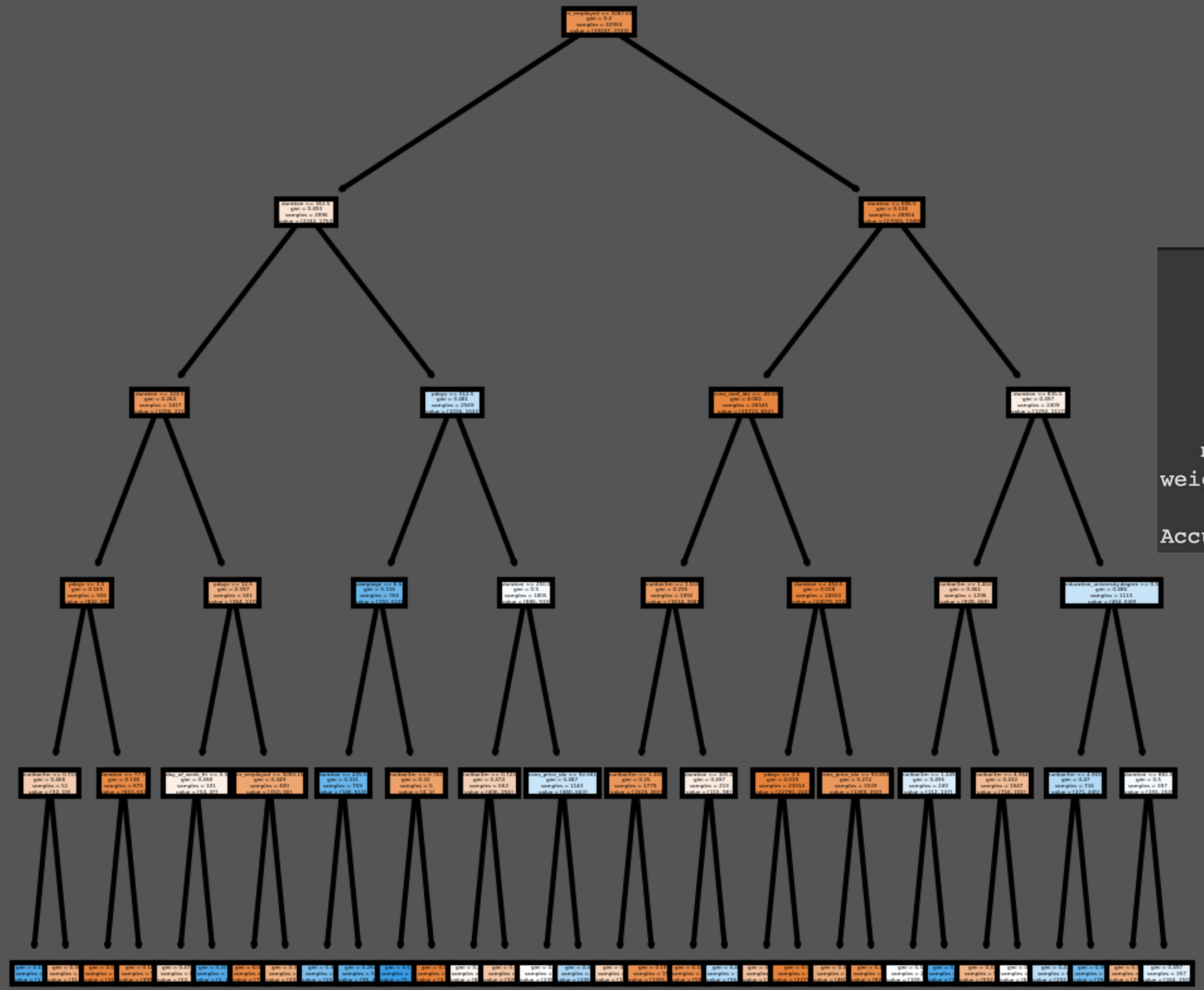
Accuracy: 0.9071376547705754

Precision,Recall,Accuracy and the F1 Score

AUC-ROC Curve:



Scikit Learn Decision Tree



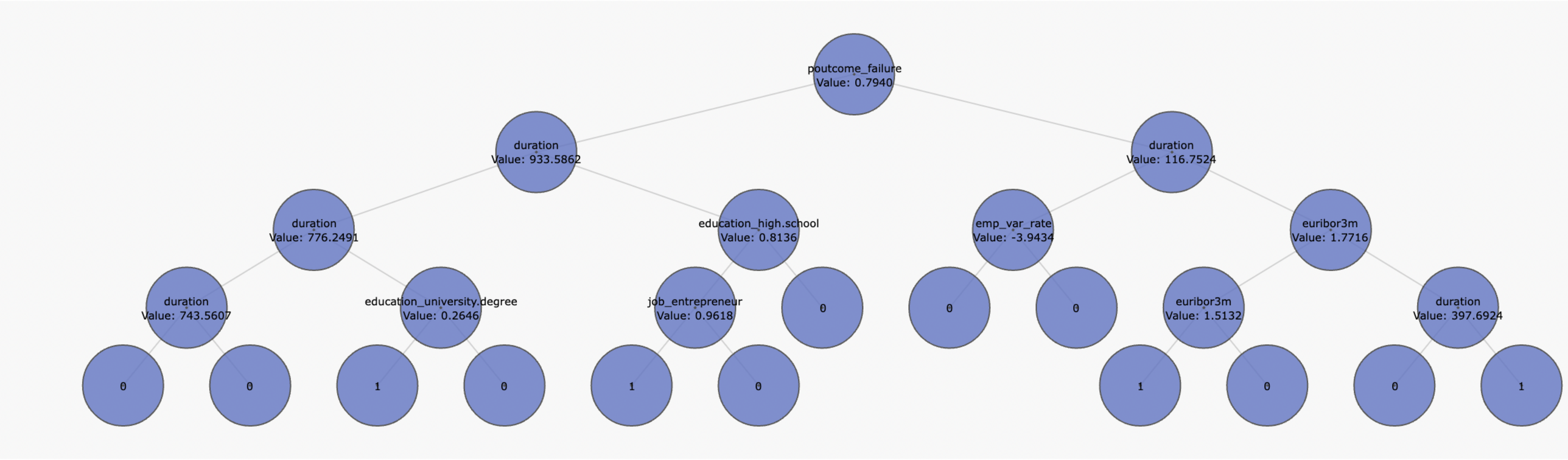
Statistics:

	precision	recall	f1-score	support
0	0.94	0.97	0.95	7301
1	0.67	0.51	0.58	937
accuracy			0.92	8238
macro avg	0.81	0.74	0.77	8238
weighted avg	0.91	0.92	0.91	8238

Accuracy: 0.9159990288905074

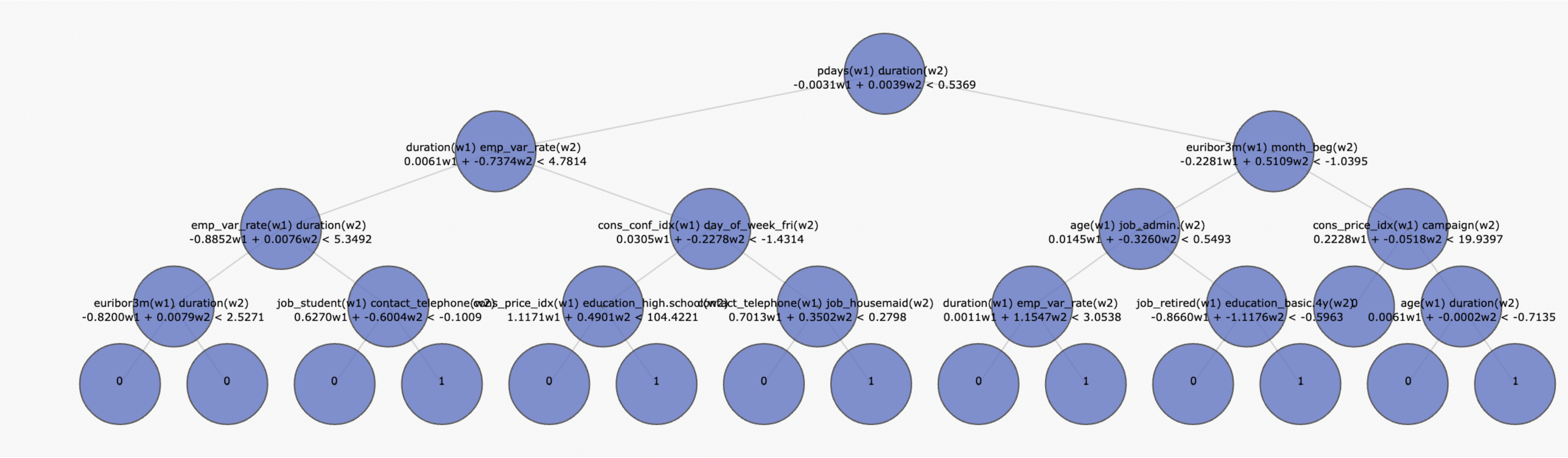
Our Decision Tree(single attribute)

Tree with Reingold-Tilford Layout



Our Decision Tree(double attribute)

Decision Tree



K-Fold Cross Validation

Banking

1 Attribute for Split

Accuracy Score: 0.960902256390977

Recall Score: 0.9154652565662260

F1 Score Score: 0.8493585511611993

Precision Score: 0.813744239631336 5

2 Attibutes for Split

Accuracy Score: 0.962406015037593 9

Recall Score: 0.9045486116292419

Fl Score Score: 0.8548667949434670

Precision Score: 0.8295538461538461

Decision Tree Rules Comparision

1. The normal DT classifier determines the split value at each node by looking at all the unique values of a particular attribute.

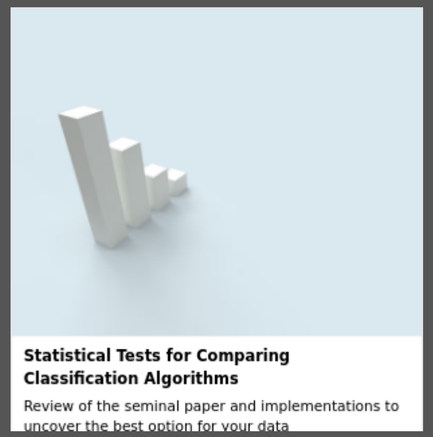
2. In our implementation, the split value is determined by conducting logistic regression for each attribute/ pair of attributes at each node. Hence, the split values are different for the rules from the normal DT classifier and our implementation.

3. For splitting a node with a pair of attributes, we see slightly better accuracy compared to the normal DT classifier because our implementation of 2 attributes for split at each node unlocks the degree of freedom for the decision boundary to split the data. Instead of parallel lines to axes such as $x > 3$, now the decision boundary can have lines such as $ax + by > c$, where a, b, c are determined from logistic regression. Hence, its better performance is justified.

Libraries Used

- Numpy
- Sklearn
- Scipy
- Weka
- Pandas
- MatplotLib.pyplit

References Used



https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html

https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262