Assignment1

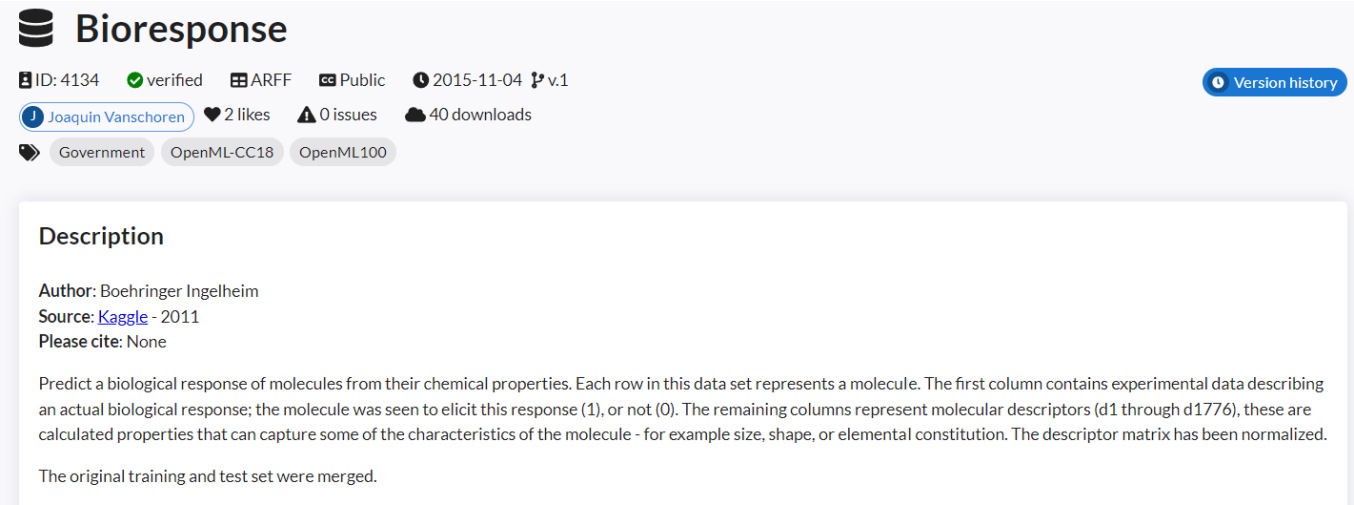
Min Wei,Li

Data 1

Data 1

- data_id = 4134 (name = Bioresponse)
- Data mission is to predict a biological response of molecules from their chemical properties.
- Each row in this data set represents a molecule.
- the molecule was seen to elicit this response (1), or not (0).

https://www.openml.org/search?type=data&sort=qualities.NumberOfNumericFeatures&status=any&qualities.NumberOfClasses=%3D_2&qualities.NumberOfInstances=between_1000_10000&id=4134



Bioresponse

ID: 4134 verified ARFF Public 2015-11-04 v.1 Version history

Joaquin Vanschoren 2 likes 0 issues 40 downloads

Government OpenML-CC18 OpenML100

Description

Author: Boehringer Ingelheim
Source: Kaggle - 2011
Please cite: None

Predict a biological response of molecules from their chemical properties. Each row in this data set represents a molecule. The first column contains experimental data describing an actual biological response; the molecule was seen to elicit this response (1), or not (0). The remaining columns represent molecular descriptors (d1 through d1776), these are calculated properties that can capture some of the characteristics of the molecule - for example size, shape, or elemental constitution. The descriptor matrix has been normalized.

The original training and test set were merged.

Feature of Data 1

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	...
0	0.000000	0.497009	0.10	0.0	0.132956	0.678031	0.273166	0.585445	0.743663	0.243144	...
1	0.366667	0.606291	0.05	0.0	0.111209	0.803455	0.106105	0.411754	0.836582	0.106480	...
2	0.033300	0.480124	0.00	0.0	0.209791	0.610350	0.356453	0.517720	0.679051	0.352308	...
3	0.000000	0.538825	0.00	0.5	0.196344	0.724230	0.235606	0.288764	0.805110	0.208989	...
4	0.100000	0.517794	0.00	0.0	0.494734	0.781422	0.154361	0.303809	0.812646	0.125177	...
...	...	...	...	...	...	...	...	...	...	...	...
3746	0.033300	0.506409	0.10	0.0	0.209887	0.633426	0.297659	0.376124	0.727093	0.308163	...
3747	0.133333	0.651023	0.15	0.0	0.151154	0.766505	0.170876	0.404546	0.787935	0.192527	...
3748	0.200000	0.520564	0.00	0.0	0.179949	0.768785	0.177341	0.471179	0.872241	0.122132	...
3749	0.100000	0.765646	0.00	0.0	0.536954	0.634936	0.342713	0.447162	0.672689	0.372936	...
3750	0.133333	0.533952	0.00	0.0	0.347966	0.757971	0.230667	0.272652	0.854116	0.140316	...

3751 rows × 1776 columns

Target of Data 1

	target
0	1
1	1
2	1
3	1
4	0
...	...
3746	1
3747	1
3748	0
3749	1
3750	0

3751 rows × 1 columns

dtype: category

AUC score for each parameters

- min_samples_leaf = 1

  AUC score = 0.73

- min_samples_leaf = 5

  AUC score = 0.77

- min_samples_leaf = 10

  AUC score = 0.79

- min_samples_leaf = 50

  AUC score = 0.81

- **min_samples_leaf = 110 (the best)**

  AUC score = 0.82

- min_samples_leaf = 200

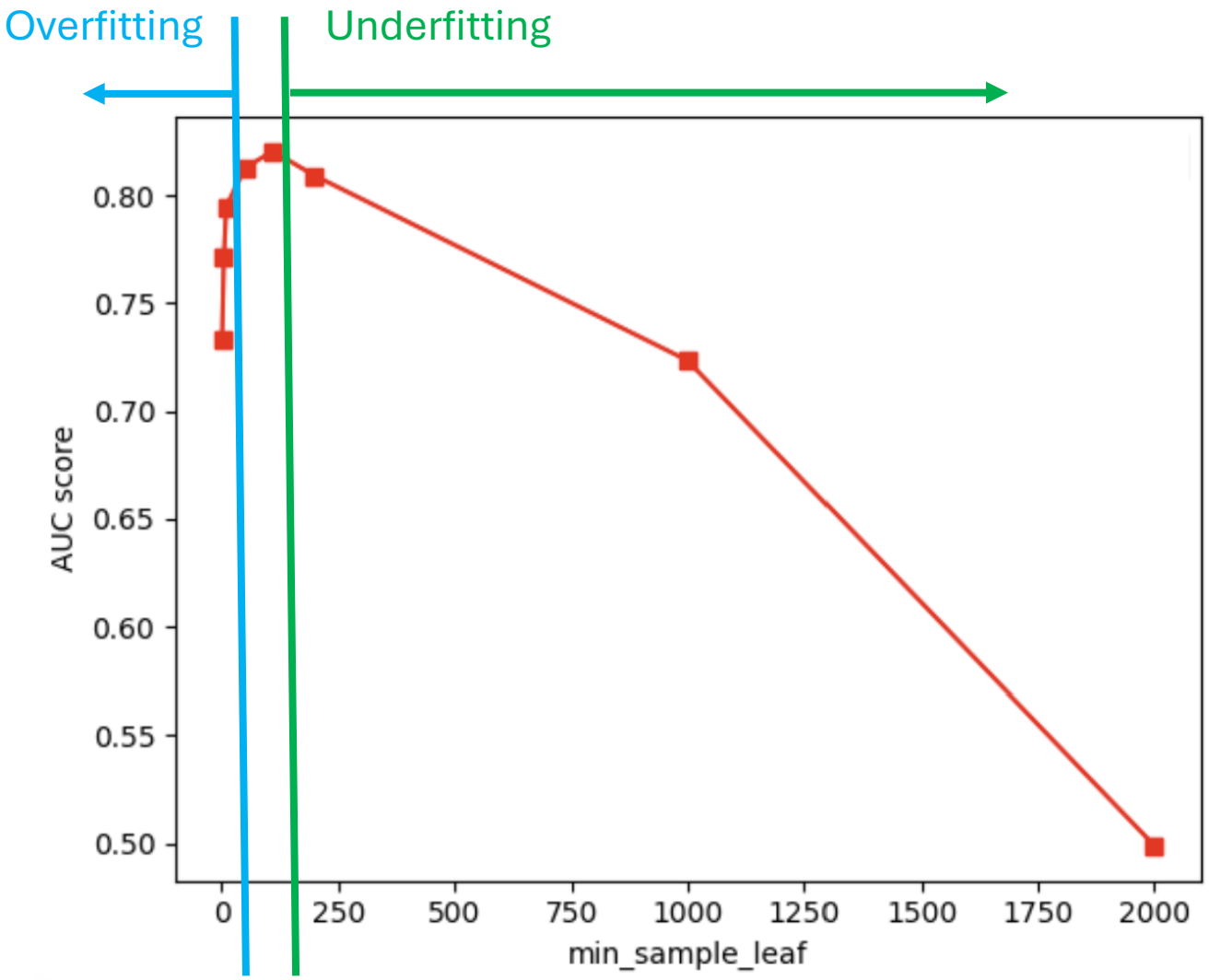
  AUC score = 0.81

- min_samples_leaf = 1000

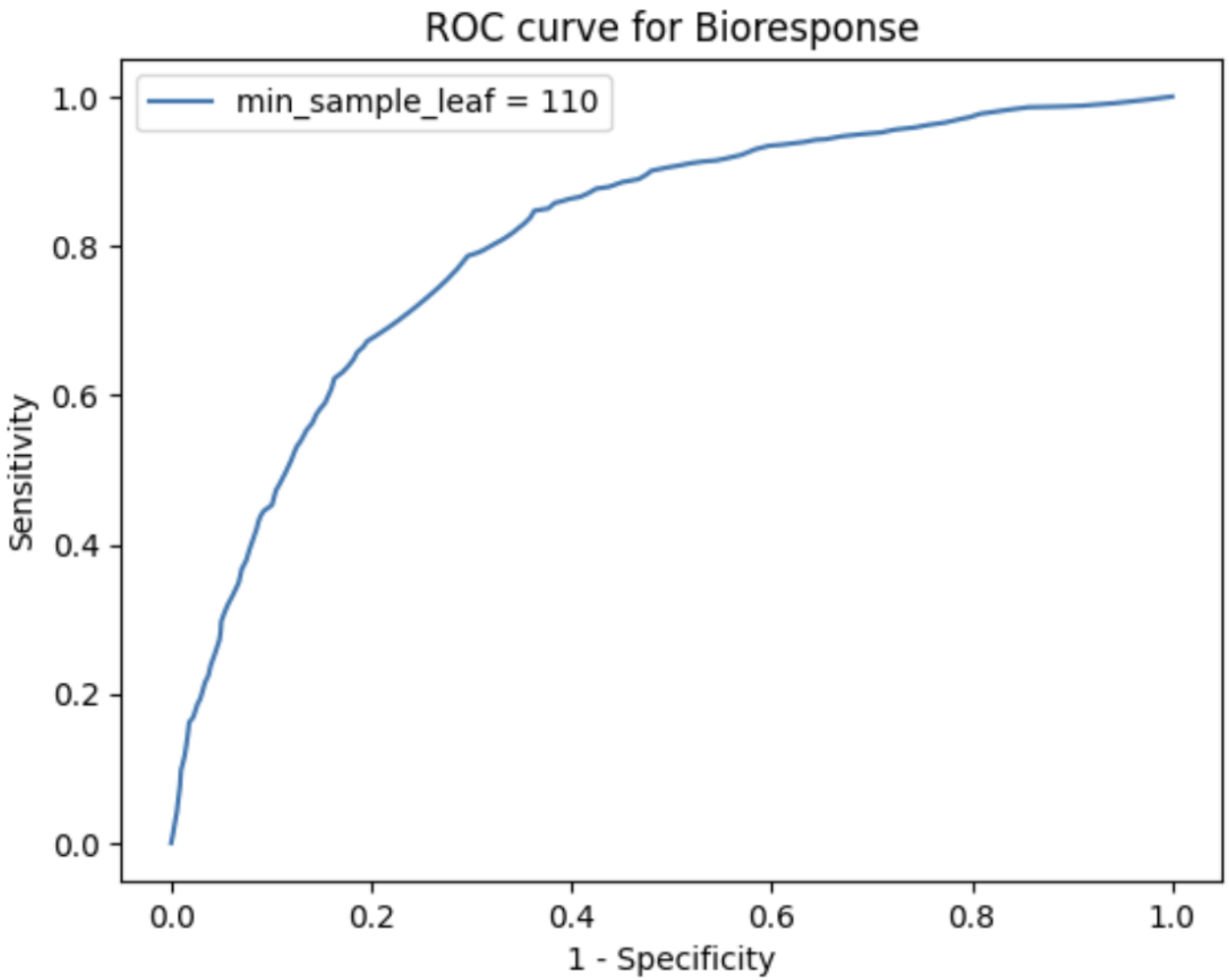
  AUC score = 0.72

- min_samples_leaf = 2000

  AUC score = 0.5



ROC curve for Bioresponse



Discussion of the results for Data 1

1. In this experiment, we first use a for loop to identify the regions where overfitting and underfitting occur. When the number of min_samples_leaf is too small, the decision tree tends to overfit due to the excessive depth of the tree. On the other hand, when the number of min_samples_leaf is too large, the decision tree becomes too shallow, leading to underfitting.

2. The large number of features in Data 1 results in longer training times. Additionally, the high number of features causes the Decision Tree's AUC score to peak around 0.82, indicating that this dataset may be more suitable for training with more complex models.

Data 2

Data 2

- data_id = 41964 (name = USPS)
- Data mission is to classify 6 and 9 from datasets.
- encoded as 0 (original class 6) and 1 (original class 9).

https://www.openml.org/search?type=data&sort=qualities.NumberOfNumericFeatures&status=any&qualities.NumberOfClasses=%3D_2&qualities.NumberOfInstances=between_1000_10000&id=41964

Feature of Data 2

	double1	double2	double3	double4	double5	double6	double7	double8	double9	double10	...
0	-0.999927	-0.993644	-0.900309	-0.632621	-0.443145	-0.454436	-0.474872	-0.431176	-0.494539	-0.583648	...
1	-0.995450	-0.936326	-0.808753	-0.824952	-0.922331	-0.791464	-0.355341	-0.041017	0.234386	0.446180	...
2	-1.000000	-0.999996	-0.999957	-0.999762	-0.998096	-0.977190	-0.753359	-0.190280	0.060797	-0.192678	...
3	-0.999998	-0.999672	-0.984040	-0.783646	-0.236214	0.155985	0.223880	0.133327	-0.128543	-0.339083	...
4	-1.000000	-1.000000	-1.000000	-0.999993	-0.999807	-0.997746	-0.986723	-0.929268	-0.755894	-0.416145	...
...	...	...	...	...	...	...	...	...	...	...	...
1419	-1.000000	-1.000000	-1.000000	-0.999970	-0.999287	-0.991834	-0.946654	-0.823924	-0.624319	-0.330996	...
1420	-0.999974	-0.999339	-0.993963	-0.981067	-0.971815	-0.959439	-0.920520	-0.858829	-0.766829	-0.595252	...
1421	-0.999971	-0.999114	-0.986055	-0.880407	-0.555724	-0.115656	0.237533	0.404396	0.305455	0.031487	...
1422	-1.000000	-1.000000	-0.999907	-0.991894	-0.867838	-0.436423	0.113779	0.445012	0.522597	0.471361	...
1423	-0.999921	-0.994707	-0.916358	-0.631715	-0.299447	-0.276286	-0.427615	-0.486219	-0.489630	-0.486181	...

1424 rows × 256 columns

Target of Data 2

	int0
0	0
1	1
2	1
3	1
4	0
...	...
1419	0
1420	0
1421	1
1422	0
1423	0

1424 rows × 1 columns

dtype: category

AUC score for each parameters

- min_samples_leaf = 1

  AUC score = 0.93

- min_samples_leaf = 10

  AUC score = 0.96

- **min_samples_leaf = 40 (the best)**

  AUC score = 0.97

- min_samples_leaf = 110

  AUC score = 0.95

- min_samples_leaf = 200

  AUC score = 0.94

- min_samples_leaf = 400

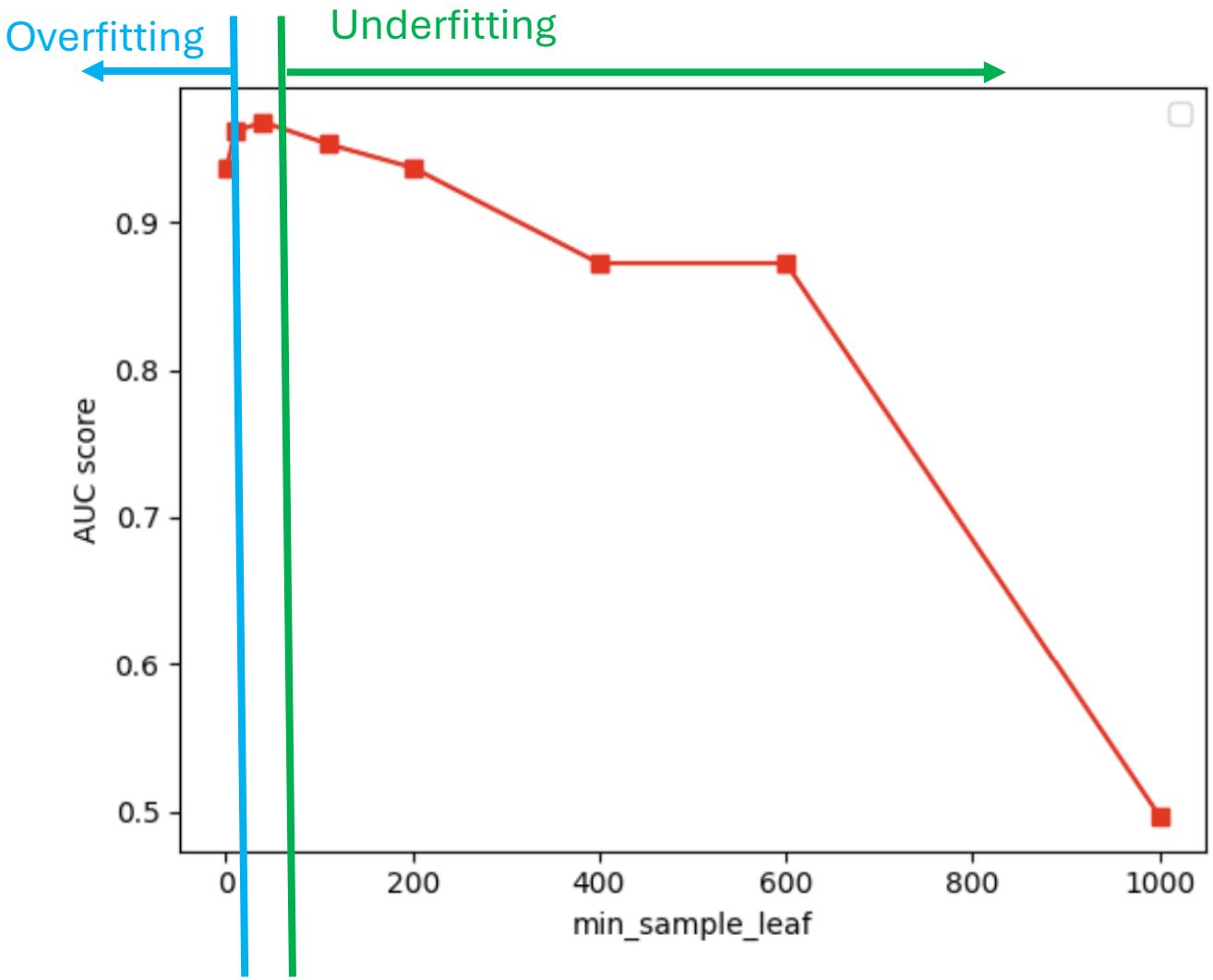
  AUC score = 0.87

- min_samples_leaf = 600

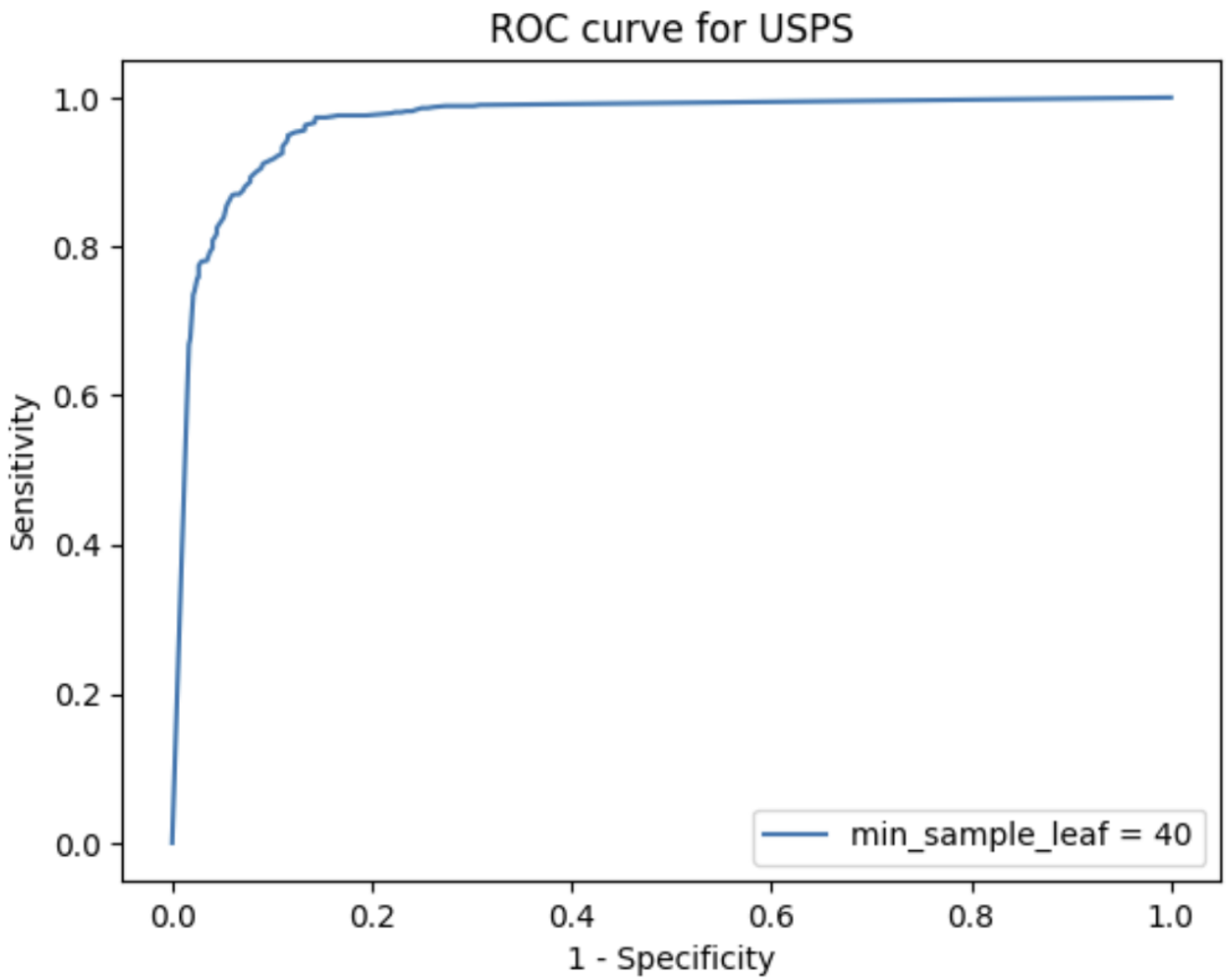
  AUC score = 0.87

- min_samples_leaf = 1000

  AUC score = 0.5



ROC curve for Data 2



Discussion of the results for Data 2

In the case of Data 2, since there are fewer features, the model's training time is relatively shorter. Additionally, it can be observed that when the min_samples_leaf value is below a certain threshold, the decision tree's AUC scores are high, typically above 0.9. This indicates that the decision tree model is well-suited for handling datasets with fewer features, although care must be taken to avoid overfitting.