

Decision Tree

In this session, our goal is to use Decision Tree to understand the relationship of alcohol in wine based on the cultivar of grape and additional chemical properties.

Data information

In this analysis we have data from the same region in Italy, but they are from 3 different cultivars of grape. The data contain measurements on 13 different constituents of grapes. Data is available at <https://archive.ics.uci.edu/ml/datasets/wine> and is cited as Dua, D. and Graff, C. (2019). UCI Machine Learning Repository <http://archive.ics.uci.edu/ml>. Irvine, CA: University of California, School of Information and Computer Science.

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The data contain the measured levels of 13 constituents found in each of the three types of wines. The attributes are (donated by Riccardo Leardi, riclea@anchem.unige.it)

Alcohol Malic acid Ash Alcalinity of ash Magnesium Total phenols Flavanoids Nonflavanoid phenols Proanthocyanins Color intensity Hue OD280/OD315 of diluted wines Proline Cultivar (added to original data source)

CART (Classification and Regression Tree)

Classification Tree

The target variable can take a class (discrete) set of values in tree models, which is used for classification-type problems.

Criterion

The function to measure the quality of a split. This parameter determines how the impurity of a split will be measured. The default value is 'gini', but you can also use 'entropy' as a metric for impurity.

gini

Gini impurity is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset. To compute Gini impurity for a set of items with J classes, suppose $i \in \{1, 2, \dots, J\}$ and let p_i be the fraction of items labeled with class i in the set.

$$IG(p) = J \sum_{i=1}^J (p_i) \sum_{k \neq i} (p_k) = J \sum_{i=1}^J p_i (1 - p_i) = J \sum_{i=1}^J (p_i - p_i^2) = J \sum_{i=1}^J p_i - J \sum_{i=1}^J p_i^2 = 1 - J \sum_{i=1}^J p_i^2$$
$$IG(p) = \sum_{i=1}^J J (p_i) \sum_{k \neq i} (p_k) = \sum_{i=1}^J J p_i (1 - p_i) = \sum_{i=1}^J J (p_i - p_i^2) = \sum_{i=1}^J J p_i - \sum_{i=1}^J J p_i^2 = 1 - \sum_{i=1}^J J p_i^2$$

Entropy

Entropy is defined as below, where p_1, p_2, \dots, p_J are fractions that add up to 1 and represent the percentage of each class present in child node that results from a split in the tree.

$$H(T) = IE(p_1, p_2, \dots, p_J) = - \sum_{i=1}^J p_i \log_2 p_i$$
$$H(T) = IE(p_1, p_2, \dots, p_J) = - \sum_{i=1}^J J p_i \log_2 p_i$$

Regression Tree

The target variable can take continuous values (real number) in decision tree, which is used for prediction-type problems.

Criterion

The function to measure the quality of a split.

Mean Square Error (MSE)

MSE is equal to variance reduction as feature selection criterion and minimizes the L2 loss using the mean of each terminal node.

Mean Absolute Error (MAE)

MAE is minimizing the L1 loss using the median of each terminal node.

Parameter Settings

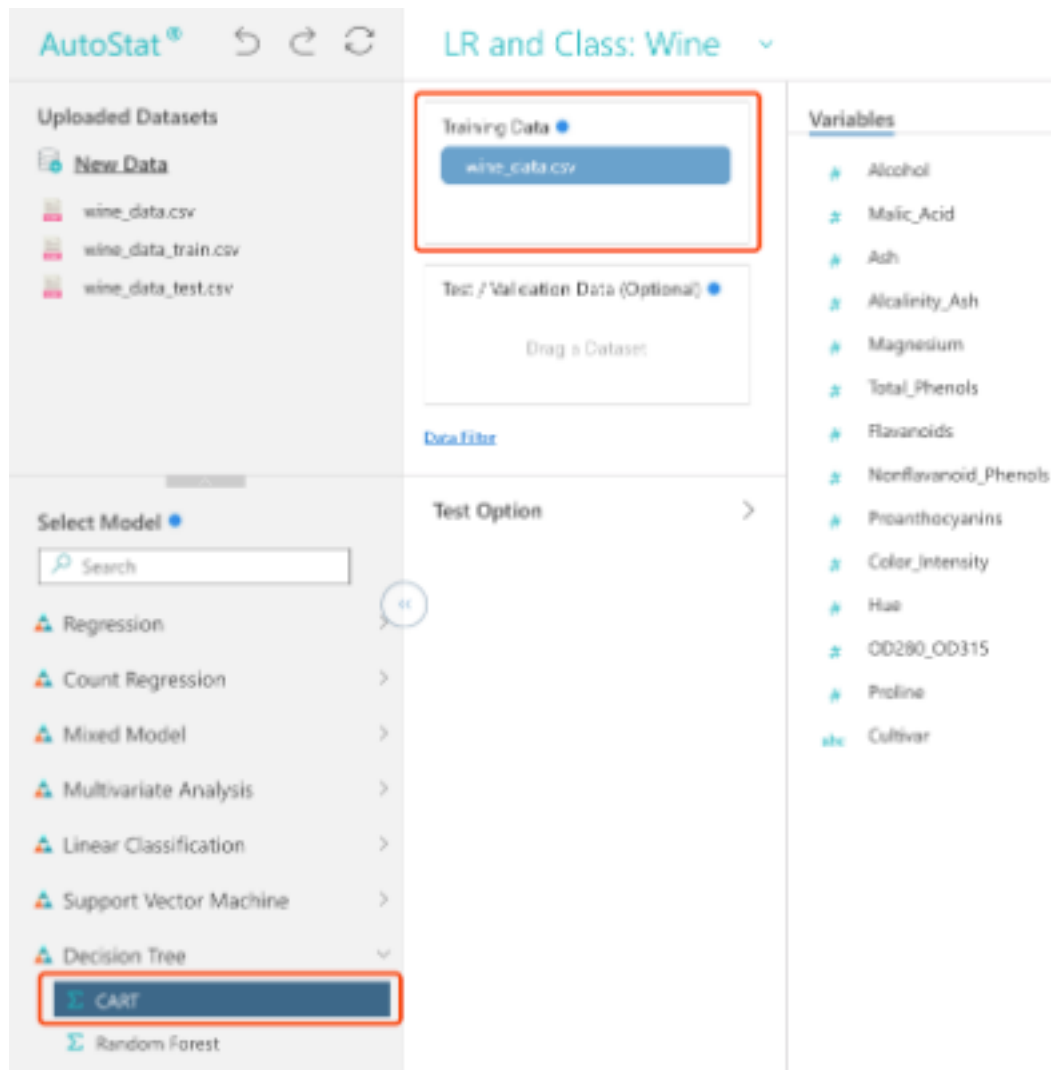
Manual replaces the exhaustive enumeration of all combinations by random selection.

Grid Search is the traditional way to perform hyperparameter optimization. It is simply an exhaustive searching through a manually specified subset of the hyperparameter space of the learning algorithm.

Bayesian Optimization is a global optimization method for noisy black-box functions. It iteratively evaluates promising hyperparameter configurations based on current model and updates them, thereby constructing a probabilistic model of function mapping from hyperparameter values to targets evaluated on the validation set.

Classifier

Select Dataset wine_data.csv and put it into training data.



Once data is selected, the available variables are shown.

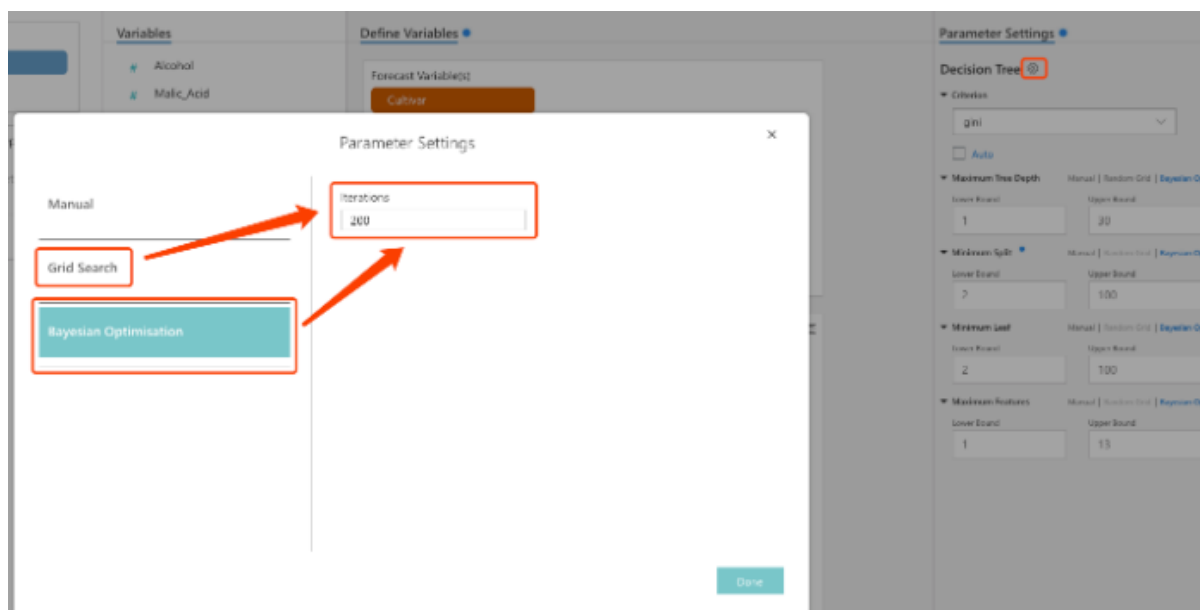
Now you need to define your model. To do this, drag your Forecast (outcome) variable into the appropriate space (see below). In this case, the forecast variable is Cultivar. Then drag the explanatory variables into the lower space or click the right arrow directly. In this case, there are all the other variables in the dataset.

The screenshot shows a web interface titled "Define Variables". It is divided into two main sections. The top section, labeled "Forecast Variable(s)", contains a single orange button labeled "Cultivar". The bottom section, labeled "Explanatory Variable(s)", contains a list of 14 teal buttons: "Alcohol", "Malic_Acid", "Ash", "Alcalinity_Ash", "Magnesium", "Total_Phenols", "Flavanoids", "Nonflavanoid_Phenols", "Proanthocyanins", "Color_Intensity", "Hue", and "OD280_OD315". To the right of this list are two small icons: a plus sign in a square and a minus sign in a square. The minus sign icon is highlighted with a red rectangular box.

If you are using all of the other variables, use this shortcut. If you want to remove some of the variables, simply click to make them disappear. Note, you should have you Forecast variable defined before you do this, or it will appear in your explanatory variables.

Now you can choose parameter settings.

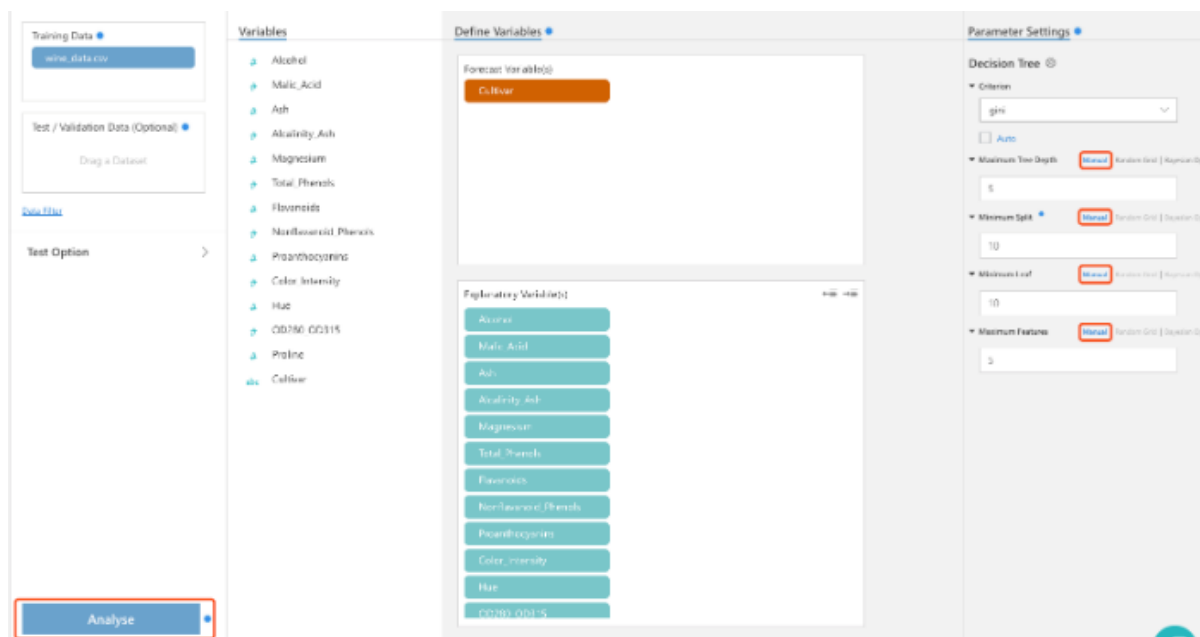
Firstly, I would like to change **Grid Search** and **Bayesian Optimisation** from default 30 to 200. It can improve the model accuracy.



Let's compare the differences among **Manual**, **Random Grid** and **Bayesian Opt** with default setting.

Manual

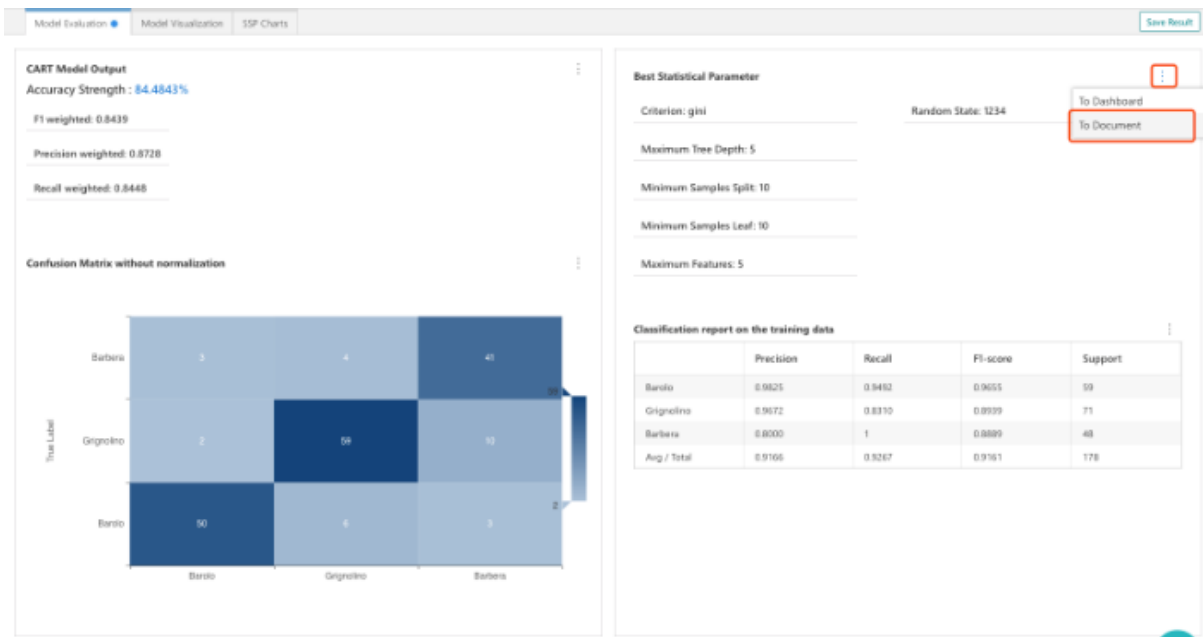
Change the setting from Bayesian Opt to Manual and click Analyse.



Once the Analysis is implemented, the Output Module will be shown with three tabs:

- Model Evaluation
- Model Visualization
- SSP Charts

All the elements in the three tabs can be sent to the Document Builder which is ready for importing into reports or papers. Clicking hamburger dots located in the top right corner of each elements. The three documents have been prepared for further steps.



if you want to keep your model you may like rename your model in more appropriate words, or you can delete as you want. You can even modify it by clicking Modify Model.

Result List

Pipeline

None

Pipeline Run

None

CART - Tue 11th Feb 20, 13:12

CART - Tue 11th Feb 20, 11:33

Model Evaluation

CART Model Output

Accuracy Strength

F1 weighted: 0.8439

Precision weighted:

Recall weighted: 0.8

Rename

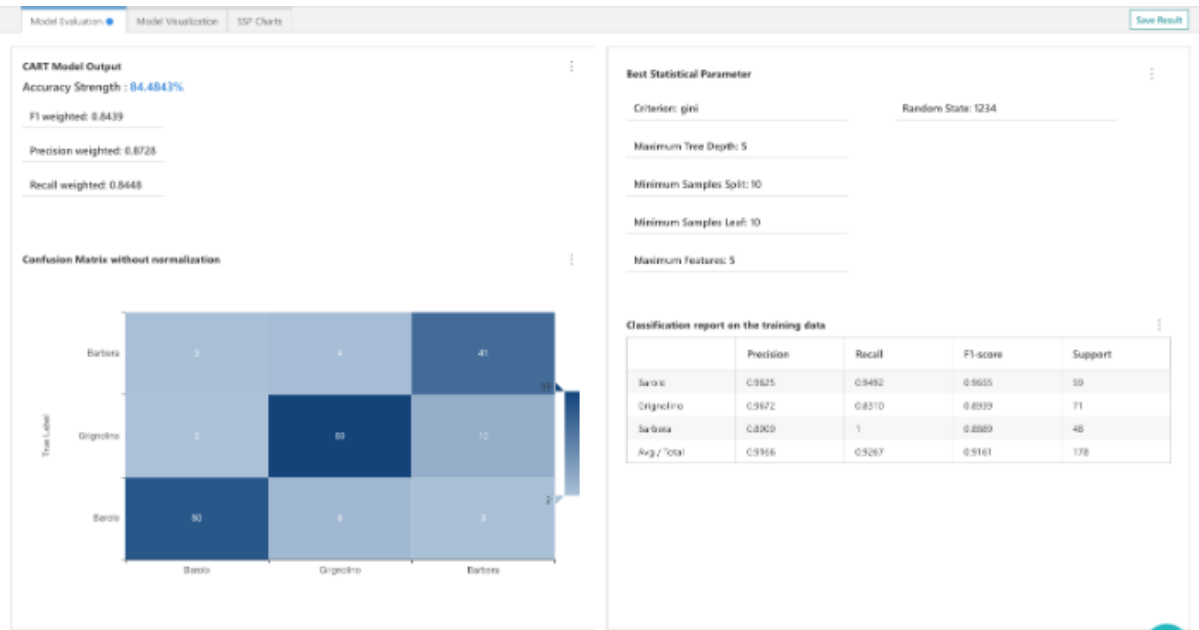
Delete

Modify Model

Create Group

Move to

Go to **Model Evaluation** and have a look at the CART Model Output



It shows that the Accuracy Strength is 84.4843%. The confusion matrix loss 28 value. Let's do the other two settings for comparison.

Random Grid

Change the setting from Manual to Random Grid and click Analyse.

Parameter Settings

Decision Tree

▼ Criterion

gini

☐ Auto

▼ Maximum Tree Depth Manual **Random Grid** Bayesian Opt

Start Step Size Num of Step

5 5 5

▲ Minimum Split Manual **Random Grid** Bayesian Opt

▼ Minimum Leaf Manual **Random Grid** Bayesian Opt

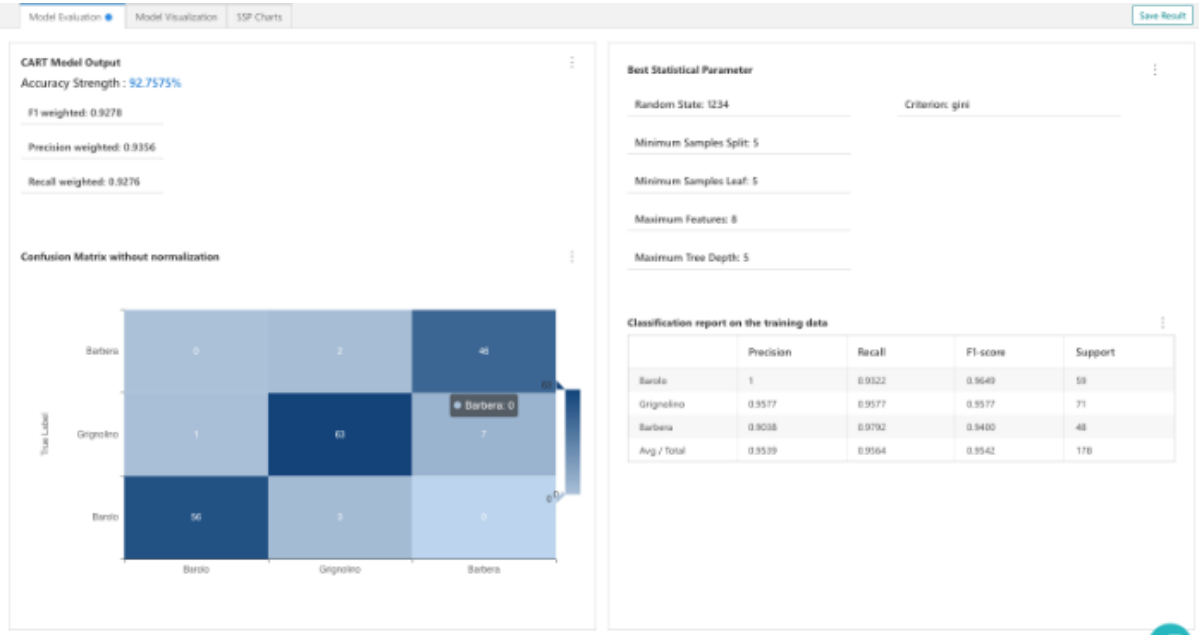
Start Step Size Num of Step

5 5 5

▼ Maximum Features Manual **Random Grid** Bayesian Opt

Start Step Size Num of Step

1 1 13



The Model Evaluation shows that the Accuracy Strength is 92.7575%. The confusion matrix shows that it loss 13 value.

Bayesian Opt

Change the setting from Random Grid to Bayesian Opt and click Analyse.

Parameter Settings

Decision Tree

▼ Criterion

gini

☐ Auto

▼ Maximum Tree Depth

Manual **Random Grid** Bayesian Opt

Start

5

Step Size

5

Num of Step

5

▲ Minimum Split

Manual **Random Grid** Bayesian Opt

▼ Minimum Leaf

Manual **Random Grid** Bayesian Opt

Start

5

Step Size

5

Num of Step

5

▼ Maximum Features

Manual **Random Grid** Bayesian Opt

Start

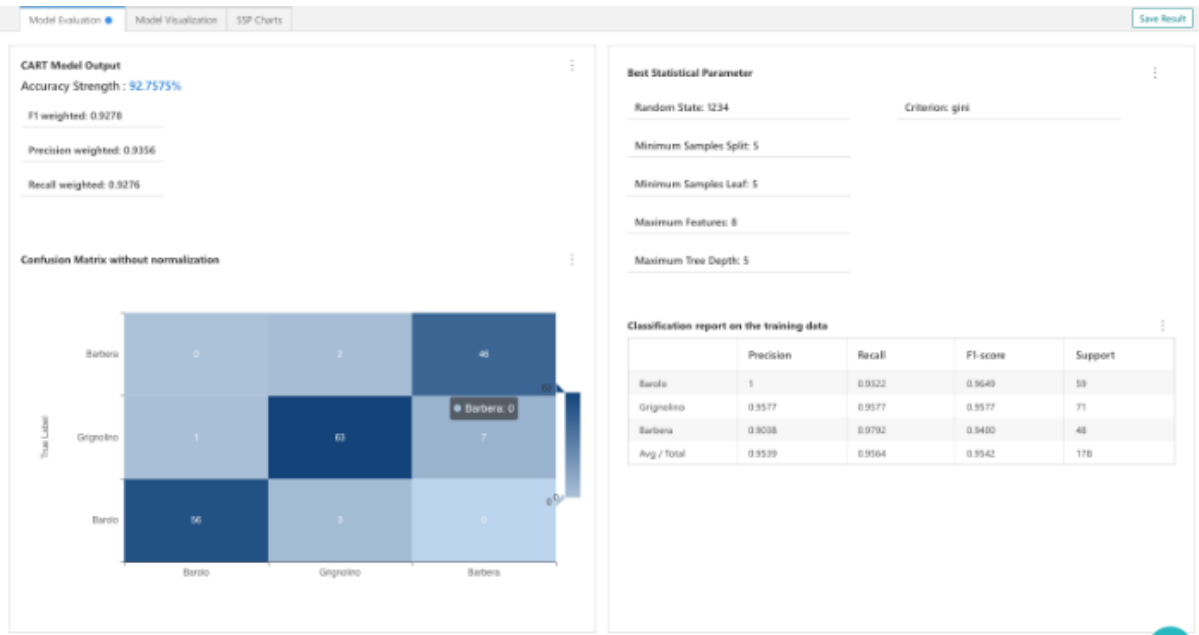
1

Step Size

1

Num of Step

13



The Model Evaluation shows that the Accuracy Strength is 92.2020%. The confusion matrix loss 14 value.

Parameter Settings

Decision Tree

▼ Criterion

gini

☐ Auto

▼ Maximum Tree Depth Manual | Random Grid **Bayesian Opt**

Lower Bound: 1 Upper Bound: 50

▼ Minimum Split Manual | Random Grid **Bayesian Opt**

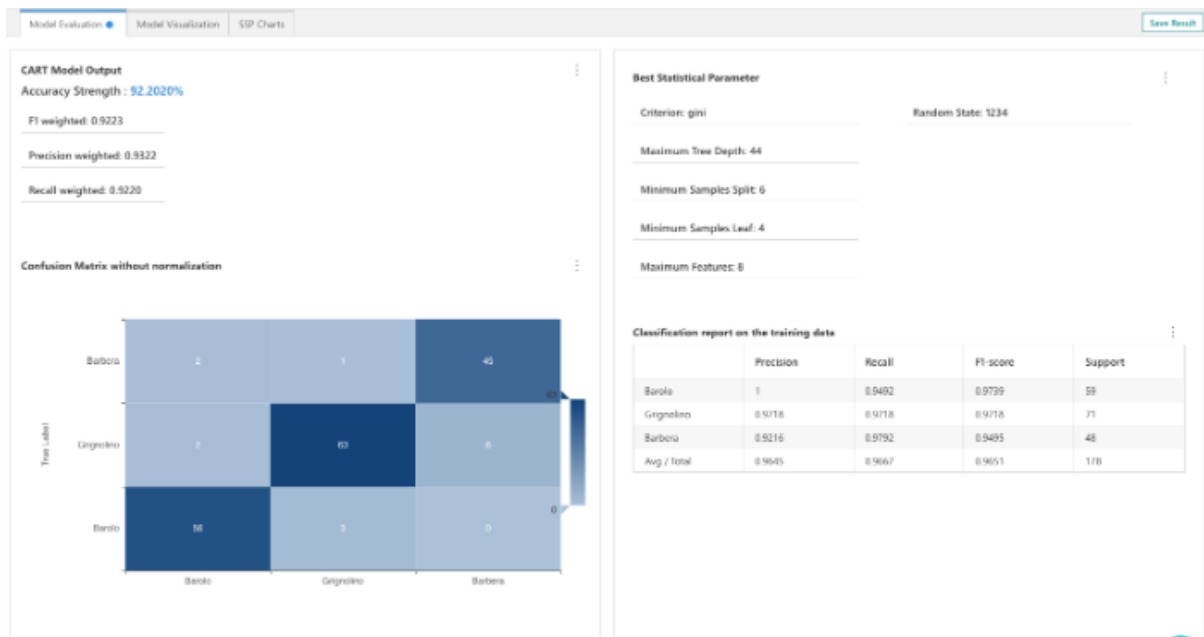
Lower Bound: 2 Upper Bound: 100

▼ Minimum Leaf Manual | Random Grid **Bayesian Opt**

Lower Bound: 2 Upper Bound: 100

▼ Maximum Features Manual | Random Grid **Bayesian Opt**

Lower Bound: 1 Upper Bound: 13



By comparing the three accuracy strength, it seems that there has not too much different between **Random Grid** and **Bayesian Opt**.

Advanced settings:

Maximum Tree Depth

The maximum depth of the tree. Without limitaion, the nodes will expanded until all leaves are pure or until all leaves contain less than minimum split. Not limiting the growth of a decision tree may lead to over-fitting.

Minimum Split

The minimum number of samples a node for splitting. The default value is two. This parameter can be used to regularize the tree.

Minimum Leaf Samples

The minimum number of samples for leaf nodes. The default value is set to two. This parameter can be used to limit the growth of the tree.

Max Features

The number of features to consider when looking for the best split. If the value is not set, the decision tree will consider all features available to make the best split.

The Bayesian Opt seems have high maximum tree depth, which may be overfitting. Let's see if we can lower the maximum tree depth without reducing the accuracy strength.

Change Upper Bound from 50 to 30 in Maximum Tree Depth and click Analyse.

Parameter Settings

Decision Tree

▼ Criterion

gini

☐ Auto

▼ Maximum Tree Depth

Manual | Random Grid | Bayesian Opt

Lower Bound: 1

Upper Bound: 30

▼ Minimum Split

Manual | Random Grid | Bayesian Opt

Lower Bound: 2

Upper Bound: 100

▼ Minimum Leaf

Manual | Random Grid | Bayesian Opt

Lower Bound: 2

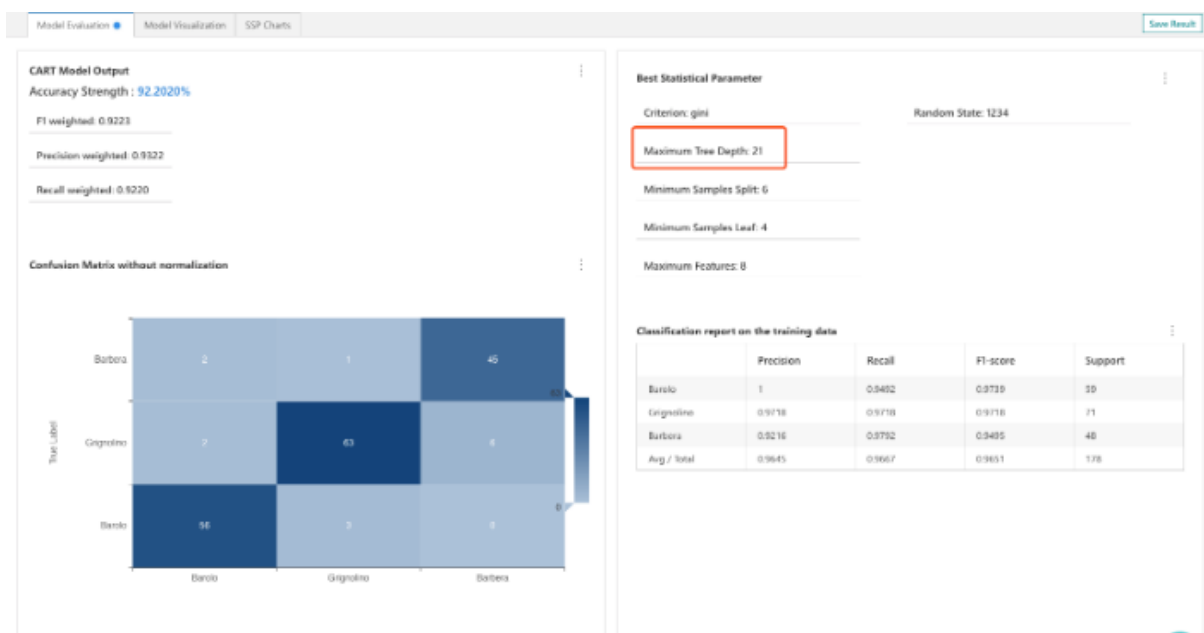
Upper Bound: 100

▼ Maximum Features

Manual | Random Grid | Bayesian Opt

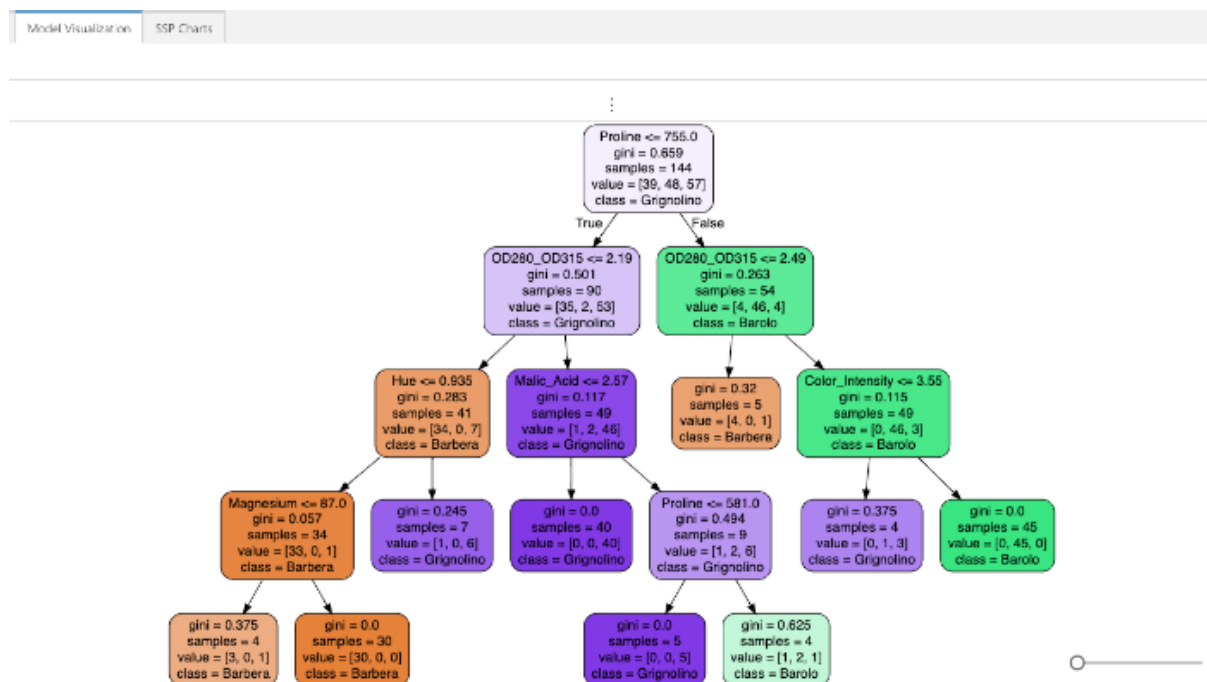
Lower Bound: 1

Upper Bound: 13



The Accuracy Strength are the same. But the Maximum Tree Depth decrease from 44 to 21, which can reduce overfitting.

Go to **Model Visualization**. Let's see how to interpret this tree.



The tree above grows to a depth of 5, and there have 8 nodes and 9 leaves. Each box contains several characteristics. Let's describe from the top node, also referred as the root node. The root node is at a depth of zero. A node is a point along the decision tree where a question is asked. This question divides the data into smaller subsets.

Proline <=755.0 : The first question the decision tree ask is if the Proline is less than 755. Based on the result, it is either True or False.

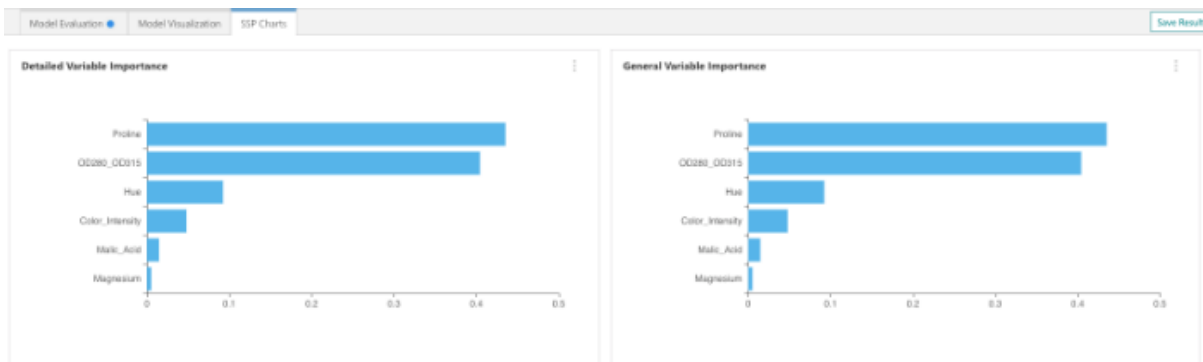
gini = 0.659 : The gini score is a metric that quantifies the purity of the node/leaf. A gini score greater than zero means that samples contained within that node belong to different classes. A gini score of zero means that the node is pure. Only a single class of samples exist with in that node. we have gini score greater than sero. Thereby, we know that the samples contained in the root node belong to different classes.

Value =[39,48,57][39,48,57]: The value list tells you the number of samples at the given node fall into each class. The first element of the list shows the number of samples belonging to Barbera class. The second element of the list shows the number of samples belonging to Barbera class. The third element of the list shows the number of samples belonging to Grignolino classes.

class = Grignolino : The class value shows the prediction a given node will make and it can be determined from the value list. Whichever class occurs the most within the node will be selected as the class value. If the decision tree were to end at the root node, it will predict that all 144 samples belonged to the Grignolino class. If the value are the same, the decision tree will choose the first class on the list by default.

Go to **SSP Charts**.

The Chart shows the importance of variables.



How about we change the criteria from gini to entropy. We can check if the accuracy will be better.

Parameter Settings

Decision Tree

Criterion

entropy (selected)

gini

entropy (selected)

Lower Bound: 1 Upper Bound: 30

Minimum Split

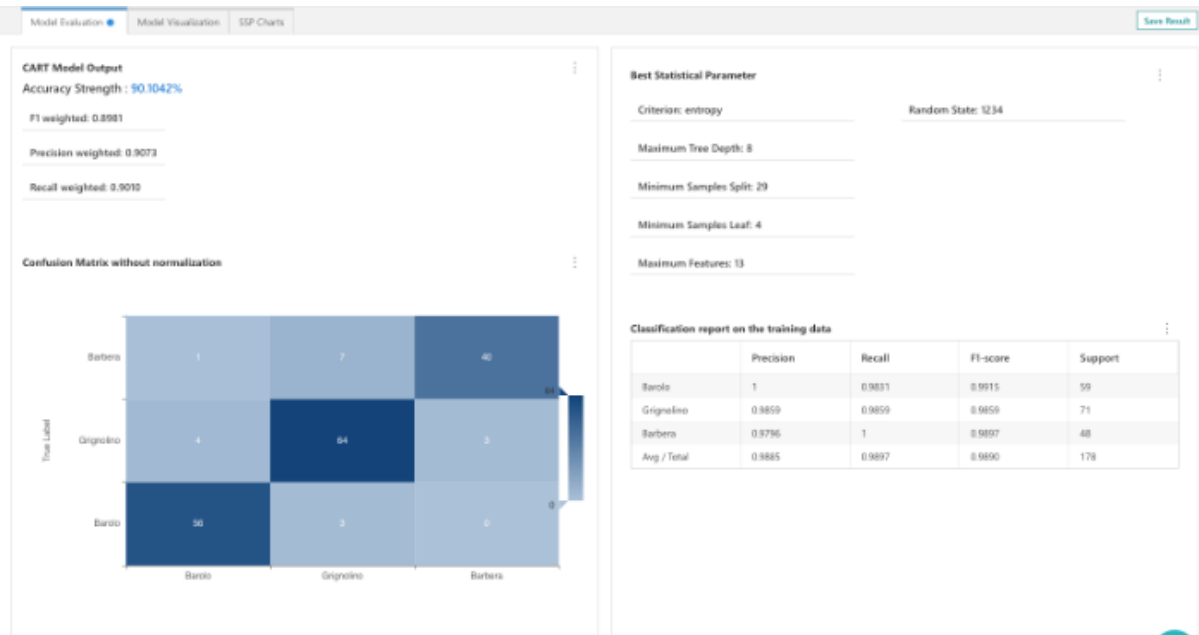
Lower Bound: 2 Upper Bound: 100

Minimum Leaf

Lower Bound: 2 Upper Bound: 100

Maximum Features

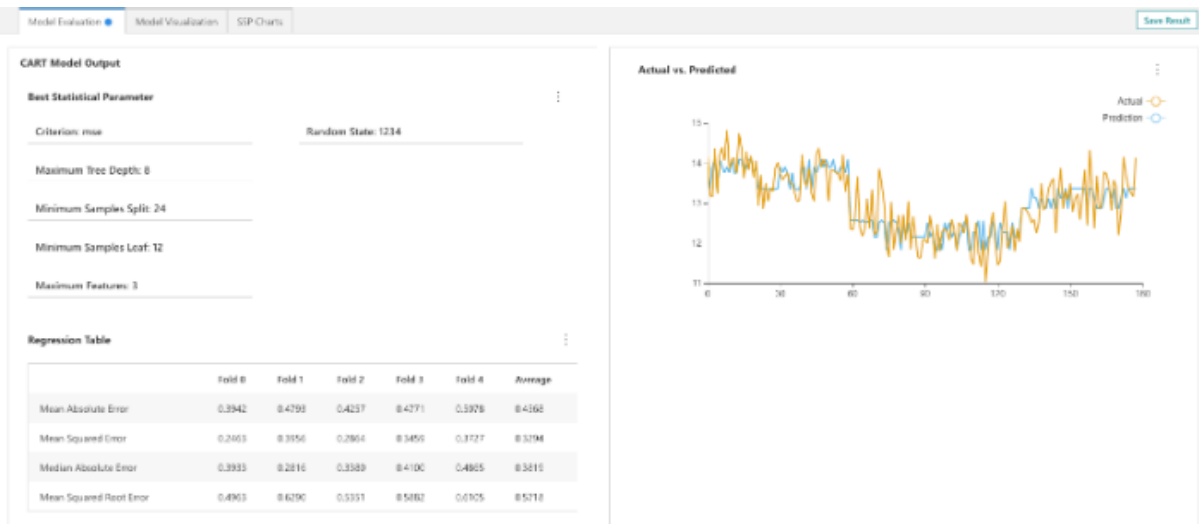
Lower Bound: 1 Upper Bound: 13



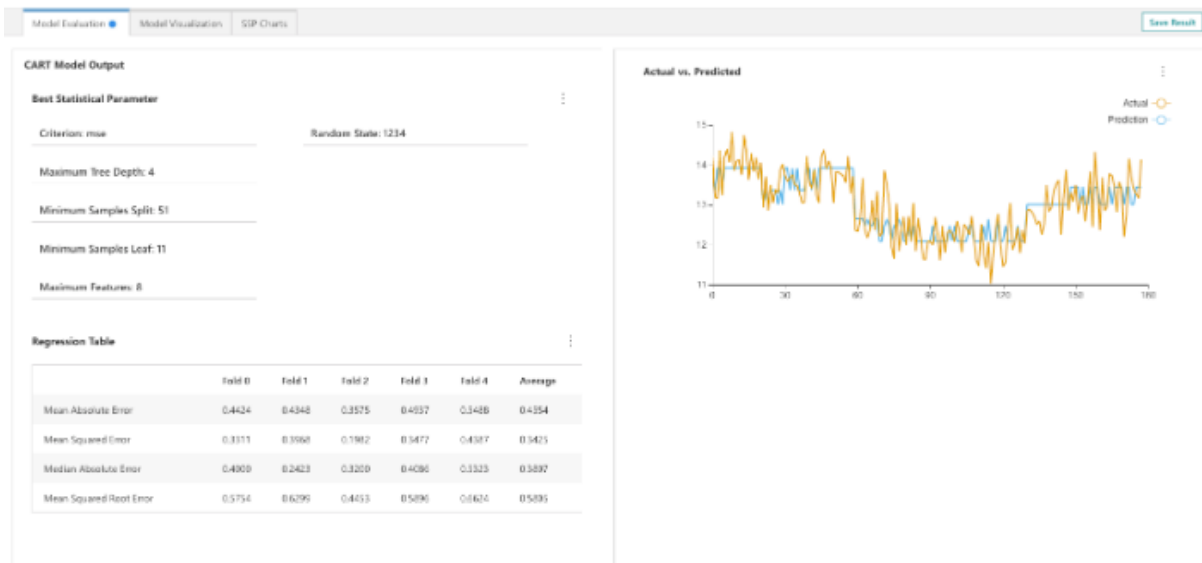
The accuracy decrease from 92.2020% to 90.1042%. It just small dataset, there are not large different between gini and entropy criteria.

Regressor

In this case, the Forecast variable is Alcohol, and the explanatory variables are all the other variables in the dataset. The parameter settings set as default.



It shows that Mean Absolute Error (MAE) is 0.4554 and Mean Squared Error (MSE) is 0.3425. MAE and MSE are low, which means it perform good. We can try if there has any lower MAE and MSE.



Let's change the default iteration from 30 to 200 in Bayesian Optimization.

Parameter Settings

Manual

Grid Search

Bayesian Optimisation

Iterations

Done

It shows that Mean Absolute Error (MAE) is 0.4568 and Mean Squared Error (MSE) is 0.3294. There just have a slightly change.

Try to enter different values into parameter settings, and see if there has any better performance.

Parameter Settings

Decision Tree

▼ Criterion

Mean Square Error (MSE)



☐ Auto

▼ Maximum Tree Depth

Manual | Random Grid | Bayesian Opt

Lower Bound

1

Upper Bound

100

▼ Minimum Split

Manual | Random Grid | Bayesian Opt

Lower Bound

2

Upper Bound

200

▼ Minimum Leaf Samples

Manual | Random Grid | Bayesian Opt

Lower Bound

2

Upper Bound

200

▼ Max Features

Manual | Random Grid | Bayesian Opt

Lower Bound

1

Upper Bound

13

Model Evaluation

Model Visualization

SSP Charts

CART Model Output

Best Statistical Parameter

Criterion: mse

Maximum Tree Depth: 37

Minimum Samples Split: 3

Minimum Samples Leaf: 12

Maximum Features: 2

Random State: 1234

Regression Table


	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
Mean Absolute Error	0.4142	0.4040	0.3984	0.4867	0.4396	0.4286
Mean Squared Error	0.2518	0.3171	0.2271	0.3629	0.2846	0.2887
Median Absolute Error	0.3425	0.2321	0.3143	0.4606	0.3617	0.3423
Mean Squared Root Error	0.5018	0.5631	0.4765	0.6024	0.5335	0.5355

The performance seems better. However, the Maximum Tree Depth looks little bit high. Repeat to change the parameters, and see which can show better.

Parameter Settings

Decision Tree

▼ Criterion

Mean Square Error (MSE) 

☐ Auto


▼ Maximum Tree Depth Manual | Random Grid | Bayesian Opt

Lower Bound

1

Upper Bound

120


▼ Minimum Split  Manual | Random Grid | Bayesian Opt

Lower Bound

2

Upper Bound

120

▼ Minimum Leaf Samples  Manual | Random Grid | Bayesian Opt

Lower Bound

2

Upper Bound

120

▼ Max Features Manual | Random Grid | Bayesian Opt

Lower Bound

1

Upper Bound

13

Finally, the parameters perform better than before With 0.4376 MAE and 0.2932 MSE.

CART Model Output

Best Statistical Parameter

Criterion: mse

Random State: 1234

Maximum Tree Depth: 8

Minimum Samples Split: 20

Minimum Samples Leaf: 3

Maximum Features: 1

Regression Table

	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
Mean Absolute Error	0.3858	0.4104	0.4561	0.5002	0.4354	0.4376
Mean Squared Error	0.2210	0.3119	0.2997	0.3614	0.2720	0.2932
Median Absolute Error	0.3369	0.2482	0.4267	0.4012	0.4325	0.3691
Mean Squared Root Error	0.4701	0.5585	0.5474	0.6012	0.5215	0.5397

Random Forest

CART work great with the data used to create them, but they are not flexible when it comes to classifying new samples. Random Forests combine the simplicity of decision trees with flexibility resulting in a vast improvement in accuracy.

Random Forest has one more advanced setting compared to CART:

Number of Estimators

The number of trees in the forest. The default value is set to 100.

Out-of-Bag-Evaluation

Whether to use out-of-bag samples to estimate the generalization accuracy. The default value is set to True.

Bootstrap

Whether bootstrap samples are used when building trees. If False, the whole dataset is used to build each tree.

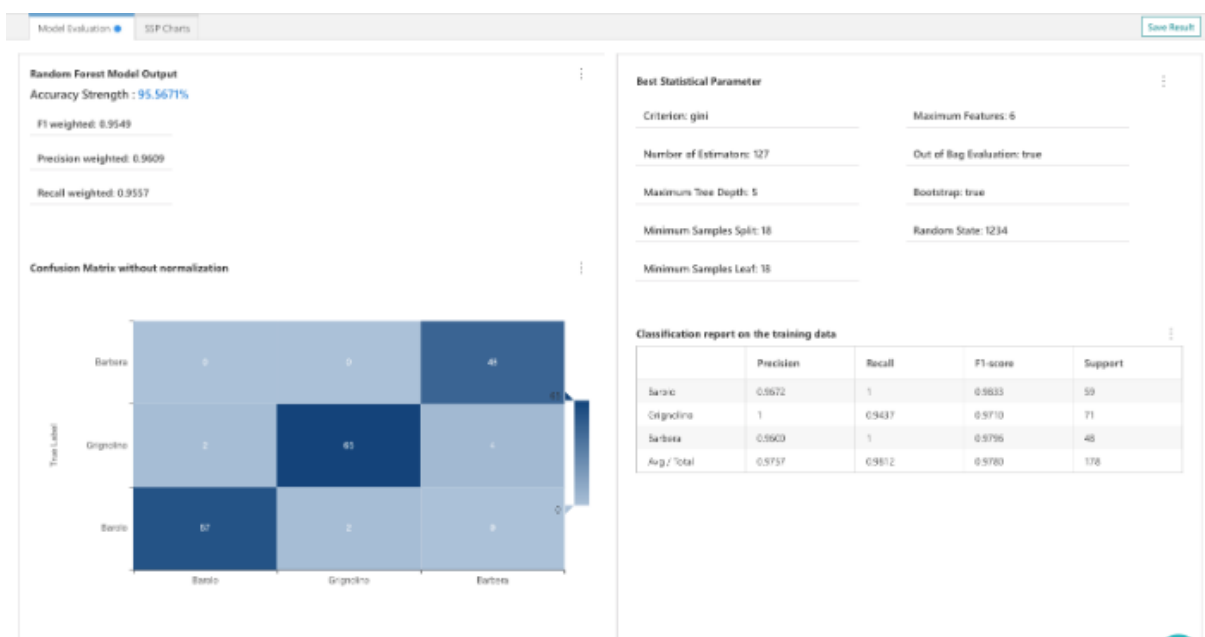
Classifier

Select Dataset wine_data.csv and put it into training data, which is the same as before. The Forecast (outcome) variable is Cultivar as before. The explanatory variables are all the other variables in the dataset. The parameter settings set as default at first.

The screenshot shows the machine learning interface with the following settings:

- Uploaded Datasets:** 'wine_data.csv' is selected as 'Training Data'.
- Define Variables:** 'Cultivar' is the 'Forecast Variable(s)'. All other variables (Alcohol, Malic_Acid, Ash, Alkalinity_Ash, Magnesium, Total_Phenols, Flavanols, Nonflavanoid_Phenols, Proanthocyanins, Color_Intensity, Hue, OODBG_ODDTG, Purity, Cultivar) are listed as 'Explanatory Variables'.
- Parameter Settings:**
 - Criterion: gini
 - Number of Estimators: 100
 - Maximum Tree Depth: 5
 - Minimum Samples Split: 18
 - Minimum Samples Leaf: 18
 - Out of Bag Evaluation: True

The Random Forest Model Output shows below:



With same parameter default setting, the accuracy is 92.2020% in CART while the accuracy is 95.5671%. Random forest perform better than CART in prediction with same settings.

How about we change **Number of Estimators** upper bound from 150 to 300?

Parameter Settings

Random Forest

▼ Criterion

☐ Auto

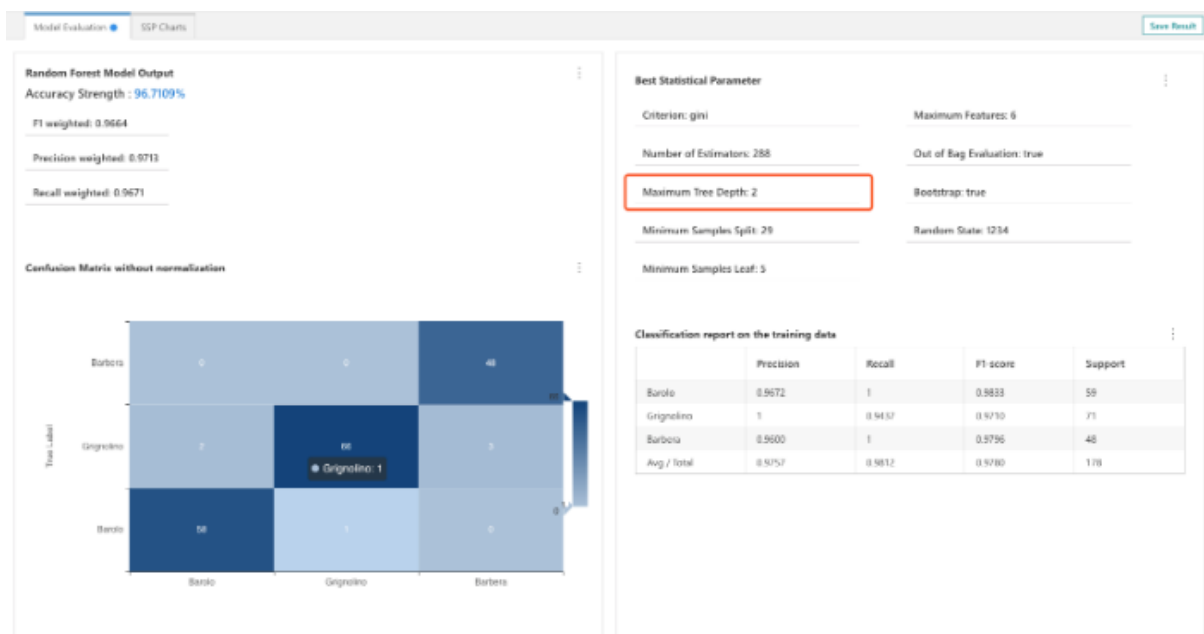
▼ Number of Estimators

Manual | Random Grid | Bayesian Opt

Lower Bound

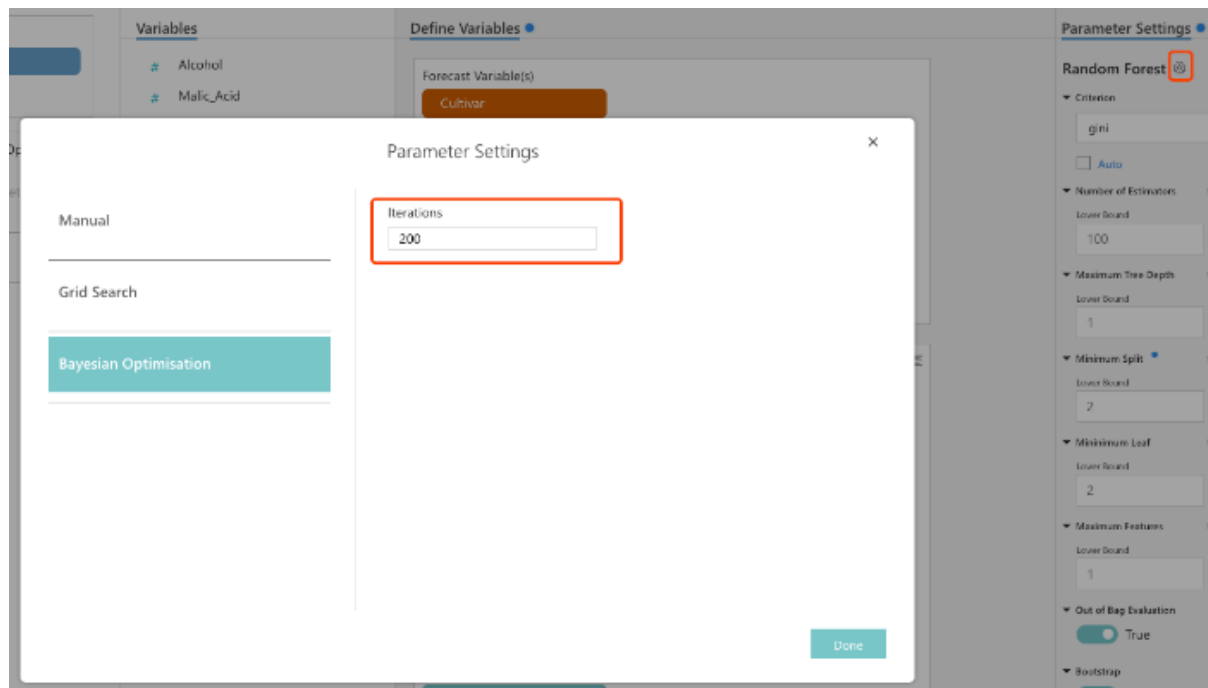
Upper Bound

It seems the accuracy increase from 95.5671% to 96.7109%. Isn't it better?

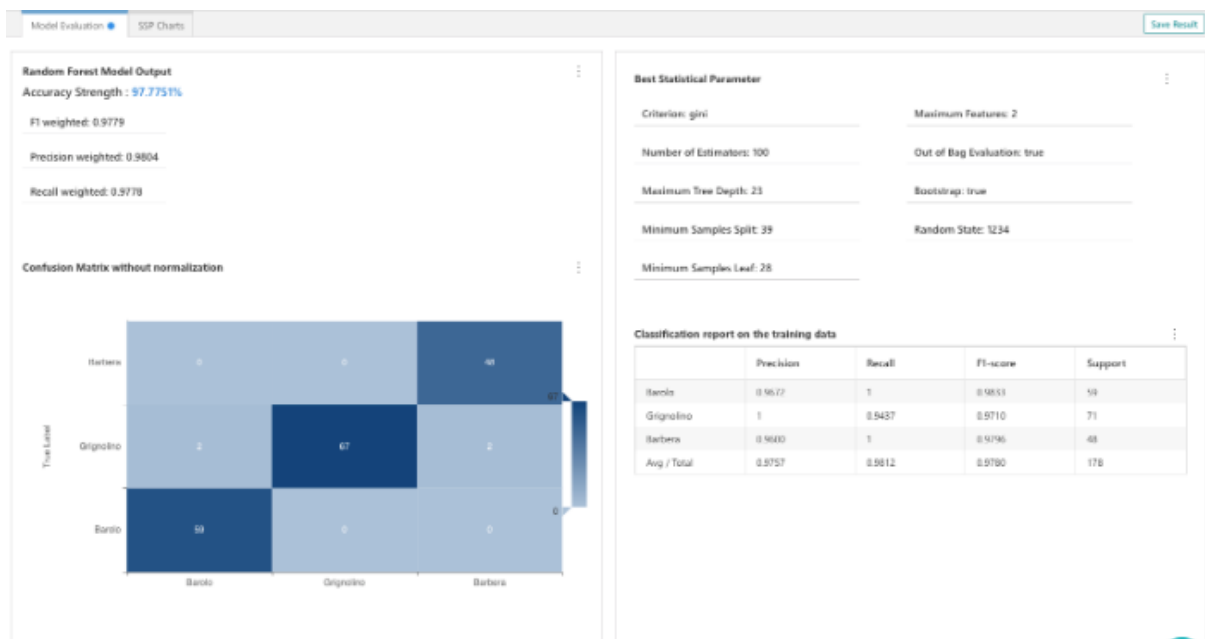


Carefully, please have a look at the Maximum Tree Depth. It only have 2 Depth. This depth is too small, which may result underfitting. For these two models, although the 95.5671% is lower accuracy. The Maximum Tree Depth is not that low. Therefore, the upper bound set as 150 is better in this case.

Do we have other way to improve the accuracy? How about we try to improve the Parameter Settings of Bayesian Optimization from default 30 to 200? Let's have a look.



Remember setting the Upper Bound to the default value 150 in Number of Estimators.



The accuracy increase from 95.5671% to 97.7751%, which performs better.

Regressor

We Just use the best perform in CART to see if the regressor will increase performance in Random Forest with the same parameter settings.

Random Forest Model Output

Best Statistical Parameter

⋮

Criterion: mse

Maximum Features: 12

Number of Estimators: 136

Out of Bag Evaluation: true

Maximum Tree Depth: 81

Bootstrap: true

Minimum Samples Split: 37

Random State: 1234

Minimum Samples Leaf: 5

Regression Table

⋮

	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Average
Mean Absolute Error	0.3935	0.3544	0.3491	0.4989	0.6662	0.4524
Mean Squared Error	0.2312	0.2812	0.1699	0.3526	0.6295	0.3329
Median Absolute Error	0.3649	0.2213	0.3636	0.3939	0.6859	0.4059
Mean Squared Root Error	0.4808	0.5303	0.4122	0.5938	0.7934	0.5621

It seems MAE and MSE in **Random Forest** are higher than **CART**, and the Maximum Tree Depth is overfitting.

Let's exploring why!

Go back to **Linear Regression Output**

Linear Regression Model Output

Variable	Coefficient	Standard Error	t-value	p-value
CONSTANT	12.1651	0.5873	20.7148	1.6111e-47
Malic_Acid	0.0745	0.0437	1.7040	0.0903
Ash	-0.2858	0.2130	-1.3419	0.1815
Alcalinity_Ash	0.0044	0.0181	0.2424	0.8087
Magnesium	-0.0001	0.0031	-0.0326	0.9741
Total_Phenols	0.0988	0.1245	0.7932	0.4288
Flavanoids	-0.0505	0.1153	-0.4381	0.6619
Nonflavanoid_Phenols	-0.0373	0.4048	-0.0921	0.9267
Proanthocyanins	-0.0671	0.0912	-0.7357	0.4629
Color_Intensity	0.1243	0.0300	4.1419	5.5156e-5
Hue	0.3522	0.2631	1.3388	0.1825
OD280_OD315	0.0308	0.1091	0.2821	0.7782
Proline	0.0002	0.0002	0.7263	0.4687
Cultivar_Barolo	0.7246	0.2978	2.4335	0.0160
Cultivar_Grignolino	-0.3944	0.2384	-1.6541	0.1000

It seems just Cultivar_Barbera and Color_Intensity are important enough based on p-value.

When the number of variables is large, but the infraction of relevant variables small, random forests are likely to perform poorly with small m . When the number of relevant variables increases, the performance of random forests is surprisingly robust to an increase oin the number of noise variables. With 2 relevant and 12 noise variables, assume $m=\sqrt{(2+12)}\approx 4(2+12)\approx 4$. This does not hurt the performance of random forests compared with boosting. This robustness islarely due to the relative insensitivity of misclassification cost to the bias and variance of the probability estimates in each tree.