

# scikit-learn

Débora Bianca e Gabriel Arcanjo

# SCIKIT-LEARNING

Criado em 2007, anos depois surgiu no projeto **Google Summer of code** em 2010, e ainda continua em desenvolvimento.

É uma biblioteca em **python** que proporciona muitos algoritmos de **aprendizado** não-supervisionado e supervisionado.

Incluso em suas funcionalidades possui: Regressão, Classificação, Clustering, Seleção de Modelos e Pré-processamento.



# TUTORIAL UTILIZANDO SCIKIT-LEARNING

# IRIS FLOWER DATASET

Possuem o mesmo gênero, mas  
são de espécies diferentes



*Iris versicolor*



*Iris setosa*



*Iris virginica*



# IRIS FLOWER DATASET

Fisher's *Iris* Data [\[hide\]](#)

Dataset Order	Sepal length	Sepal width	Petal length	Petal width	Species								
1	5.1	3.5	1.4	0.2	I. setosa								
2	4.9	3.0	1.4	0.2	I. setosa								
3	4.7	3.2	1.3	0.2	I. setosa								
4	4.6	3.1	1.5	0.2	I. setosa								
5	5.0	3.6	1.4	0.3	I. setosa	4.7	1.4	I. versicolor					
6	5.4	3.9	1.7	0.4	I. setosa	4.5	1.5	I. versicolor					
7	4.6	3.4	1.4	0.3	I. setosa	4.9	1.5	I. versicolor					
8	5.0	3.4	1.5	0.2	I. setosa	4.0	1.3	I. versicolor					
9	4.4	2.9	1.4	0.2	I. setosa	4.6	1.5	I. versicolor					
10	4.9	3.1	1.5	0.1	I. setosa	4.5	1.3	I. versicolor					
			57	6.3	3.3	4.7	1.6	I. versicolor		6.0	2.5	I. virginica	
			58	4.9	2.4	3.3	1.0	I. versicolor		5.1	1.9	I. virginica	
			59	6.6	2.9	4.6	1.3	I. versicolor		5.9	2.1	I. virginica	
			60	5.2	2.7	3.9	1.4	I. versicolor		5.6	1.8	I. virginica	
									5.8	2.2	I. virginica		
								106	7.6	3.0	6.6	2.1	I. virginica
								107	4.9	2.5	4.5	1.7	I. virginica
								108	7.3	2.9	6.3	1.8	I. virginica
								109	6.7	2.5	5.8	1.8	I. virginica
								110	7.2	3.6	6.1	2.5	I. virginica

# CÓDIGO EM PYTHON

```
1# -*- coding: utf-8 -*-
2import numpy as np
3import graphviz
4from sklearn import tree
5from sklearn.datasets import load_iris
```

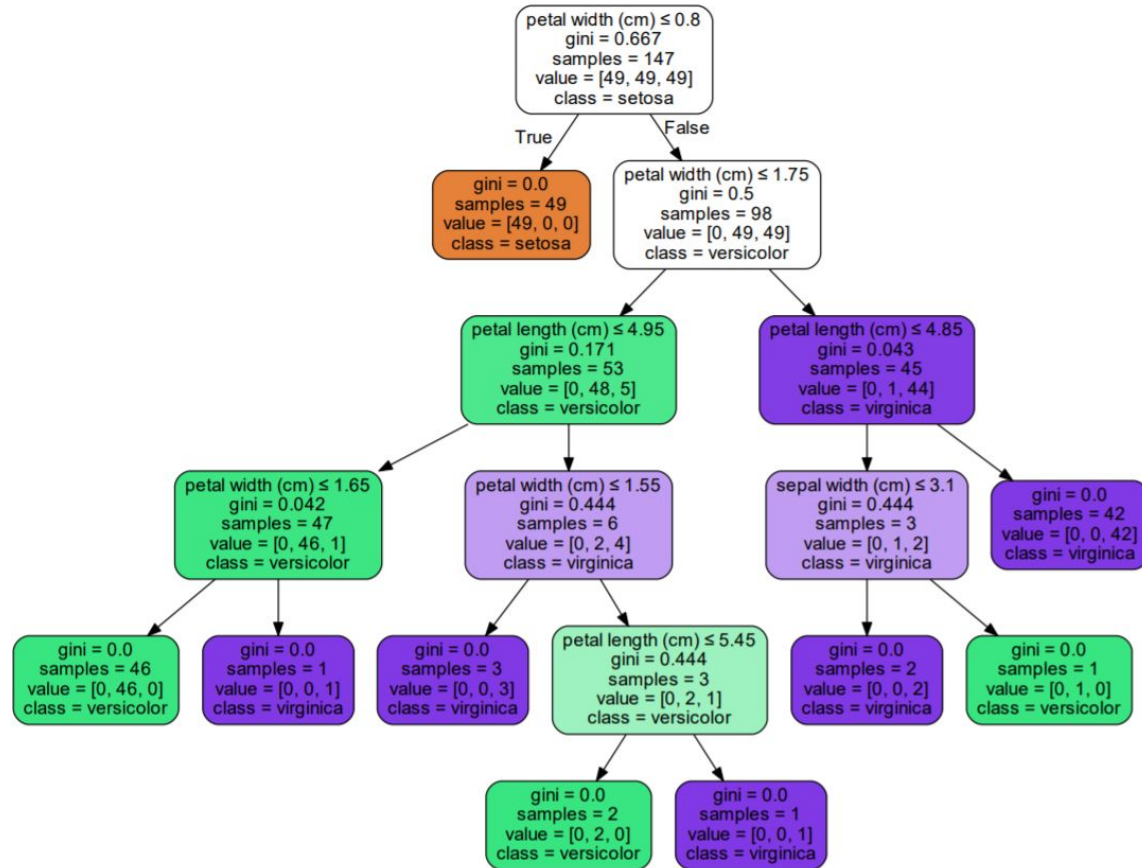
```
17test_idx = [10, 110, 71]
18
19#training data
20training_target = np.delete(iris.target, test_idx)
21training_data = np.delete(iris.data, test_idx, axis = 0)
22
23#testing data
24test_target = iris.target[test_idx]
25test_data = iris.data[test_idx]
26
27clf = tree.DecisionTreeClassifier()
28clf.fit(training_data, training_target)
```

```
30#predict label to a new flower
31print ("Data set ID:", test_idx)
32print ("Data set classification:", test_target)
33print ("Data test classification:", clf.predict(test_data))
34print ("[1]Iris setosa\t[2]Iris versicolor\t[3]Iris virginica")
```

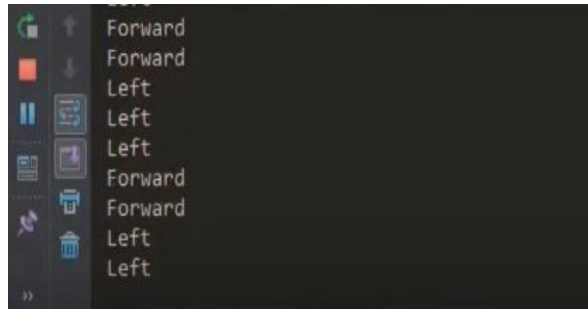
```
36#Gera o arquivo em pdf da árvore de decisões para classificar as flores
37dot_data = tree.export_graphviz (clf, out_file = None,
38                                feature_names = iris.feature_names,
39                                class_names = iris.target_names,
40                                filled = True, rounded = True,
41                                special_characters = True)
42graph = graphviz.Source(dot_data)
43graph.render("iris")
```

Data set ID: [10, 110, 71]  
Data set classification: [0 2 1]  
Data test classification: [0 2 1]

# ÁRVORE DE DECISÃO

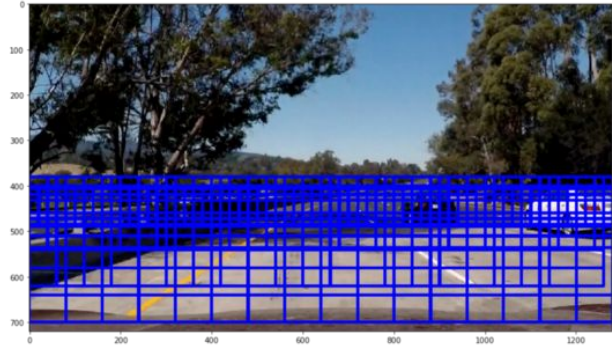
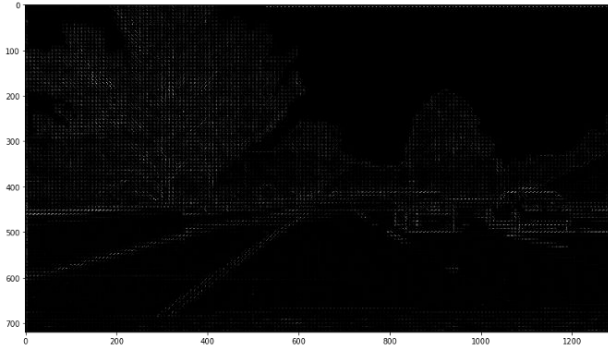


# SCIKIT-LEARN E SISTEMA EMBARCADO

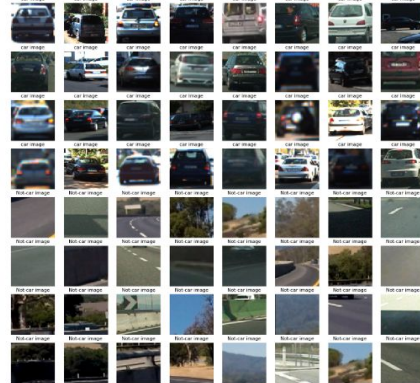




# SCIKIT-LEARN E SISTEMA EMBARCADO



Your function returned a count of 8792 cars and 8968 non-cars  
of size: (64, 64, 3) and data type: uint8



# REFERÊNCIAS

**CODE ACADEMY:** <https://www.codecademy.com/articles/scikit-learn>

**SCIKIT-LEARN:** <http://scikit-learn.org/stable/modules/tree.html>

**YOUTUBE - Google Developers:** <https://www.youtube.com/watch?v=tNa99PG8hR8>

**YOUTUBE - Track following RC car:** <https://www.youtube.com/watch?v=wKVF2ItrGkI>

**MEDIUM - Self-driving Cars:**

<https://medium.com/@ricardo.zuccolo/self-driving-cars-opencv-and-svm-machine-learning-with-scikit-learn-for-vehicle-detection-on-the-bf88860e055a>

**WIKIPEDIA - Iris flower dataset:** [https://en.wikipedia.org/wiki/Iris\\_flower\\_data\\_set](https://en.wikipedia.org/wiki/Iris_flower_data_set)