W3. Decision Tree, Random Forests with Python

Bio and Health Informatics Lab

SNU
BIOINFORMATICS
INSTITUTE

Table of Contents

- Building Decision Trees
- Building Random Forests

Decision Trees

- **Decision Trees (DTs)** are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

- Also by building trees and observing its hierarchy, you will learn what features are of high importance. This can be further used for feature selection

- The interpretation of the data is very helpful and easy.

Drawbacks of Decision Trees

- If data is highly unbalanced, the model may not predict well
- Errors within the training set may propagate to child nodes
- DTs are prone to overfitting

Installing python libraries for DT and RF analysis

- pip install sklearn
- pip install graphviz

Practice 1-1: Building DTs for classifying flowers using the Iris data

- Classes: 3={Iris-Setosa, Iris-Versicolour, Iris-Virginica}



Can you tell any difference?

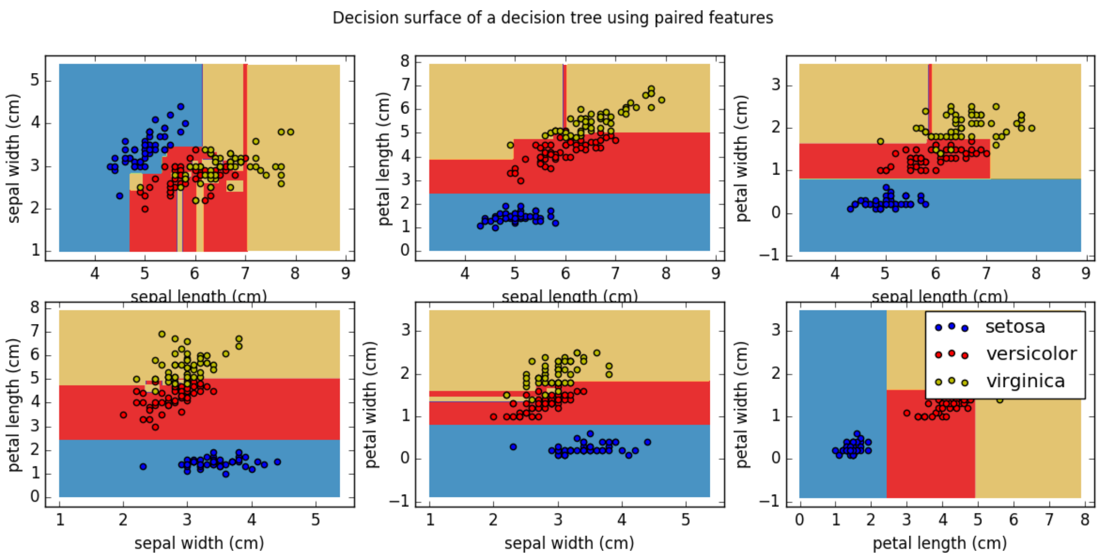
- Features: 4={Sepal length, sepal width, petal length, petal width}
- Data:

Fisher's *Iris* Data

Sepal length	Sepal width	Petal length	Petal width	Species
4.9	3.0	1.4	0.2	*I. setosa*
4.7	3.2	1.3	0.2	*I. setosa*
4.6	3.1	1.5	0.2	*I. setosa*
5.0	3.6	1.4	0.3	*I. setosa*
5.4	3.9	1.7	0.4	*I. setosa*
4.6	3.4	1.4	0.3	*I. setosa*
5.0	3.4	1.5	0.2	*I. setosa*
4.4	2.9	1.4	0.2	*I. setosa*
4.9	3.1	1.5	0.1	*I. setosa*
5.4	3.7	1.5	0.2	*I. setosa*
4.8	3.4	1.6	0.2	*I. setosa*
4.8	3.0	1.4	0.1	*I. setosa*
4.3	3.0	1.1	0.1	*I. setosa*
5.8	4.0	1.2	0.2	*I. setosa*

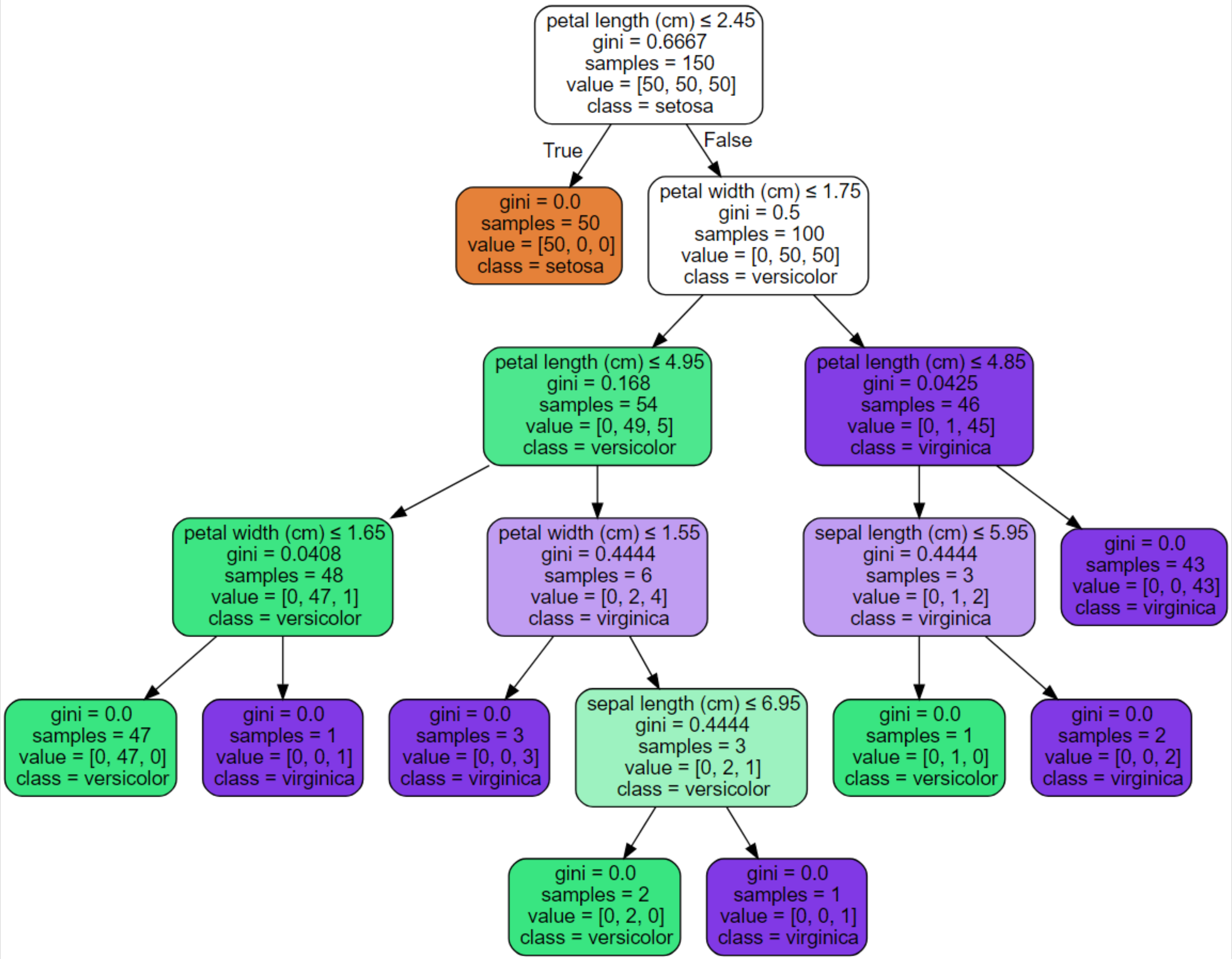
Practice 1-1: Draw the DT for this data

- Load Iris data and observe the data (data, target or labels)
- Create a decision tree and fit the data
- Perform prediction on input data
- Draw a more interpretable DT using the graphviz package
- Visualize the pair-wise decision surface of the DT



SNU BIOINFORMATICS INSTITUTE

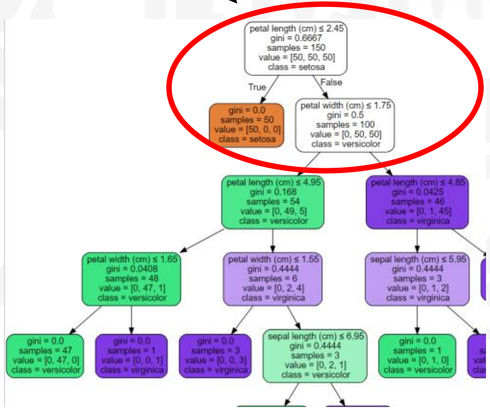
Decision Tree of the Iris data



SNU BIOINFORMATICS INSTITUTE

Practice 1-2: Decide where to split the DT

- Implement the GINI index formula
- Implement the $GINI_{split}$ formula
- Calculate the $GINI_{split}$ for the first branch
  - How should we split the data?
- What are the $GINI_{split}$ values of each feature?



SNU BIOINFORMATICS INSTITUTE

Practice 2-1:
Building Random Forests using the Iris data

- Load Iris data
- Create a data frame using pandas package
- Split the Iris data to training and testing sets
  - Ratio 75:15
- Create Random Forest(RF) classifier
- Fit data into RF
- Predict the species of the testing data
- Check accuracy
- What are the variable importance values of each feature?
- Visualize the first estimator tree of the RF
  - How many tress did RF generate?

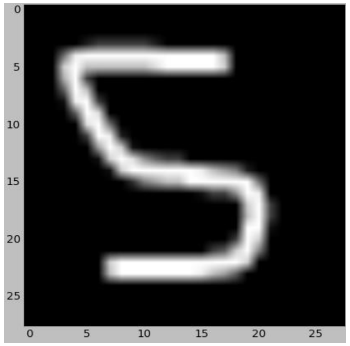
Practice 2-2:
Build a Digit Recognizer using RF

- Load MNIST digit train data ("train.csv")
- Split data into train, test data
- Generate and fit a RF using train data
- Measure accuracy using test data

- Use the whole train data
- Load the test data ("test.csv")
- Predict the digit of each image in "test.csv"
- Check the prediction

Practice 2-3:
Classify your own handwritten digits

- Draw your digit using https://sketch.io/sketchpad/
- Save your drawing as .png file
- Convert it into computable format (np.array format)
  - image2data.py imagefile
- Load it as a test data and classify the image



SNU
BIOINFORMATICS
INSTITUTE