MTU
Ollscoil Teicneolaíochta na Mumhan
Munster Technological University

MACHINE LEARNING.

Lab 07: Decision trees

BACKGROUND.

The goal of this lab exercise is to evaluate the Decision Tree classifier on the Iris dataset and to understand how the classifier decides between the different types of flowers.

Task 1.

In this task you will analyse the performance of a decision tree classifier for the Iris dataset.

A. Load the Iris dataset from the dataset repository of SciKit-Learn.
B. Create a loop to evaluate the performance of different Decision Tree Classifiers of increasing maximum depth ranging from a minimum of max_depth=1 to a maximum of max_depth=5.
C. For each of these Decision Tree Classifiers create a leave-one-out cross validation procedure and calculate the average performance on the test data as well as on the training data.
D. Plot the average performance on both the test and the training data against the maximum tree depth parameter. What can you observe? What maximum depth should you choose to obtain optimal results?

Task 2.

In this task you will visualise and analyse the decision process of the decision tree classifier for the Iris dataset.

A. Load the Iris dataset from the dataset repository of SciKit learn.
B. Create a loop and train Decision Tree Classifiers of increasing maximum depth ranging from a minimum of max_depth=1 to a maximum of max_depth=5. Visualise the resulting decision trees for each maximum depth.
C. Use these visualisations to determine which two features are the most relevant for the Iris classification problem.

Task 3.

In this task you will visualise the decisions boundaries of the decision tree classifier for the Iris dataset similar to task 4 of Lab 02 and compare the result with the visualisation obtained in task 2.

A. For the two most important features determined in the previous task determine the minimum and maximum value to create two vectors containing 100 equally spaced points

between the minimum and maximum (Hint: use `np.arange()`). Create a 2d mesh-grid from these two vectors (Hint: use `np.meshgrid()`).

B. For the remaining two features determine the average value and create two 100x100 matrices containing this average matching the two matrices generated in the previous subtask (Hint: use `np.ones()`).
C. Using the four matrices created in the previous subtasks create a 10000x4 matrix of feature vectors representing all choices of the most important features in the value range and the average value for the two less important features (Hint: use `np.flatten()`).
D. Now use the five Decision Tree Classifiers trained in the previous task to predict the classes for each of these feature points generated in the previous sub-task. Reshape the results into a regular 100x100 grid (Hint: use `np.reshape()`).
E. Visualise the grids calculated in the previous subtask as image (Hint: use `plt.imshow()`) together with the points of the Iris dataset in one figure (Hint: use `plt.scatter()`). Compare the decision boundaries to the corresponding decision trees visualised in the previous task.