Graded Assignment 2 - Decision Trees

Overview

For this assignment, you will need to implement an algorithm that can create a Decision Tree (DT) from a given dataset, demonstrating its functionality on the weather data discussed in the course material and also a dataset of your choosing, discussing and explaining the results.

You will need to provide:

1. A Jupyter notebook file (.ipynb) containing:
   - A working Python implementation of an algorithm (for example CART or ID3) that can produce a DT from a given dataset.
   - An account of the algorithmic and technical choices made to implement your DT.

2. A 3-page report (.pdf) containing:
   - Details of the dataset you have chosen to use for this assignment, and which feature you are predicting using the DT.
   - An assessment of the success of your decision tree, in terms of accuracy on test data, how well the decision tree helps interpret the data and possibly other means.
   - A critical exploration of the recent advancements and a real-world application of DTs

Guidelines

The implementation of the DT algorithm should be your own work, and only rely on basic Python packages such as NumPy. Use of "off-the-shelf" packages or code to implement a DT algorithm will be heavily punished.

For importing and manipulating datasets, you are permitted to use a wider range of Python packages. Therefore, **please keep the import statements for the implementation and data handling sections separately**. In particular, we suggest you implement your DT algorithm at the top of the notebook.

You should only use public datasets licensed for use in this context, clearly referencing their source. A good place to look for datasets is Google Dataset Search or Kaggle, although any appropriate sources can be used. These datasets should certainly not include any personal identifying information. Any other external sources should be fully referenced.

Marking Criteria

Implementation (.ipynb file)

Task 1 (20%): How successful is the implementation?

Does the code produce and display clearly a correct decision tree for the weather data? (10%)
Does the code produce and display clearly a correct decision tree for your chosen dataset? (10%)

Task 2 (20%): How technically sophisticated is the implementation?

Does the code implement a basic algorithm, or use one or more that go beyond the material presented in the course material? Are the choices well explained and justified? (10%)
How wide a range of datasets is the implementation able to handle? What types of features are admissible for inputs and outputs? (10%)

Report (.pdf file)

Task 3 (10%): How well motivated is the choice of database?

Is the choice of database and feature predicted with the decision tree well justified and explained? (5%)
Is the database chosen suitable for a DT approach? Beyond accuracy, are the benefits of using a DT approach explained? (5%)

Task 4 (20%): How well are the results explained and contextualised?

Are clear hypotheses about the data made and tested using the DT produced? (10%)
Are the results intelligently reflected upon? Are the results used to make observations and conclusions about the dataset and its source? (10%)

Task 5 (30%): Recent advances and a real-world application of DTs?

Does the report provide an in-depth overview of the recent advances in DT research? This could be related to the fitting of training data, generalisation, and interpretability of the DT. (20%)
Does the report contain a discussion on the practical application of DT, for instance, how DT models could be useful to solve real-world problems? (10%)