Assignment 2

ML Class: CS 6375

October 9, 2024

1 Assignment Policies for CS 6375

The following are the policies regarding this assignment.

1. This assignment needs be done individually by everyone.
2. You are expected to work on the assignments on your own. If I find the assignments of a group of (two or more) students very similar, the group will get zero points towards this assignment. You may possibly also be reported to the judiciary committee.
3. Please use Python for writing code. You can submit the code as a Jupyter notebook
4. For the theory questions, please use Latex
5. This Assignment is for 25 points.
6. This will be due on October 25th.
7. Late policy: We will deduct two points per day the assignment is submitted late.

2 Questions

1. **Nearest Neighbor Methods (9 Points):** This question is on Nearest Neighbor Methods

   (a) **(2 points) Nearest Neighbor Regression:** In class, we looked at k-Nearest Neighbor Classification. Can you extend this to regression? What will the algorithm look like if we do Nearest Neighbor Regression? Does it make sense to do regression with nearest neighbors?

   (b) **(2 points) Irrelevant Attributes:** Please give an example of a dataset where a nearest neighbor algorithm would perform very poorly when we have irrelevant features.

   (c) **Implementation (5 points):** Implement a k-Nearest Neighbor Algorithm from scratch on the dataset considered in the demo. Compare your results with what we obtained in the demo using sklearn. The link to the demo we did in class is `https://colab.research.google.com/drive/1coQx_fGMyi0hli6xjETUaXfy3AX1PqWX`.

2. **Decision Trees (6 Points):** There are three parts to this question. Each part is for 2 points.

   (a) Part 1: In the class, we saw that finding the split which maximizes the information gain $I(X_i; Y)$ [which is also called the mutual information] was a good strategy to greedily build the decision tree. Explain why this is a good strategy with an example. What if instead of the information gain, we were to use the conditional gain $H(Y|X_i)$. Would it make sense to maximize or minimize this?

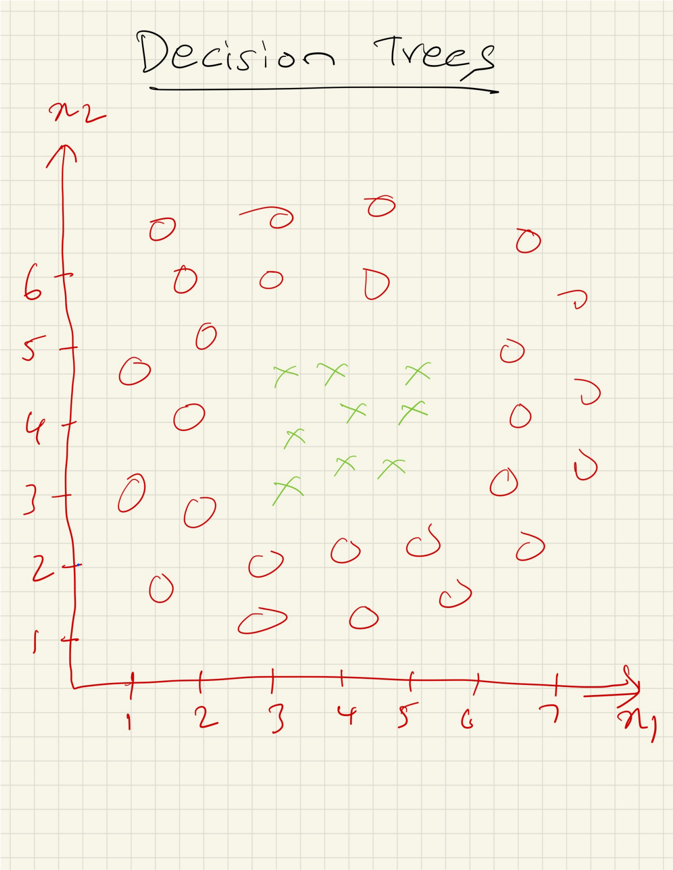

Figure 1: 2 Dimensional Data for DT Classification

(b) Part II: Why does the Gini Split condition that we saw in class make sense as a good splitting condition? What about using $G(S) = 1 - \sum_{i=1:N} p_i^3$?

(c) Part 2: Consider the data points shown in Figure 1. Draw the approximate decision boundry obtained by the a decision tree algorithm. Also provide an approximate solution obtained by the decision tree algorithm (provide the final tree which the DT algorithm would most likely give). I am **not** expecting you to do this programmatically. Argue based on the rough position of the data points, how the tree learning algorithm will grow. Next, assume you have no limit on the depth of the tree. How would the solution look in that case?

3. **MLE and MAP (5 Points):** Let us consider the Dice Roll problem we discussed briefly in class. Let X denote a Random Variable equal to the number that appears on the top of the dice. Answer the following questions:

- What are the parameters of the Dice roll process? How many parameters do we have?
- Let $\alpha_1, \cdots, \alpha_6$ denote the number of times we see $1, 2, \cdots, 6$ in a sequence of N dice rolls. Calculate the Likelihood function.
- Without solving for the Likelihood, can you tell us what the MLE estimates of the parameters should be?
- If we were to use Bayesian methods, how will you set the prior distribution?
- Calculate the Posterior Distribution and the MAP estimates.

4. **Implement a Decision Tree from Scratch (5 Points):** Implement a Decision Tree from scratch. We will implement the classification variant. Implement the simple decision tree algorithm with two variants: (i) use the conditional entropy $H(Y|X)$ as a splitting criteria, and (ii) use the Gini Coefficient as a splitting criteria. Assume the features are numerical. Compare your performance with sklearn for a comparable choice of hyper-parameters (depth of the tree, number of leaf nodes, etc). Also, contrast between the Conditional Entropy and Gini Coefficient as Splitting criteria. You can use the same dataset we considered for the demo in class (`https://colab.research.google.com/drive/1RKC12-hsxRx7GqogkQjN0ylqFy6ay2As`).