Final Report

Videos

- https://www.youtube.com/watch?v=p17C9q2M00Q (CART)

Additional Resource

- Bishop 14.4 (DT)

- https://www.saedsayad.com/decision_tree.html

1. Nice graph at top
2. Introduction from proposal
    a. Top-Down Greedy Approach: Theme
    b. Copy from before
3. https://medium.com/analytics-steps/understanding-the-gini-index-and-information-gain-in-decision-trees-ab4720518ba8 (will cover c4_5, cart, and id3)
4. Deep Analysis
    a. What is a decision tree?
        i. **Chpt 3 CMUSTUFF**
        ii. Chat 3.5 (CMUSTuff)
    b. What did you do?
        i. [We train our decision tree on car evaluation dataset and test its performance in the classification setting. We use accuracy as one of the measure .
        ii. https://archive.ics.uci.edu/ml/datasets/Car+Evaluation (READ)
            1. Talk about dataset. How it was made. Talk about attributes. Number of instances. Maybe create histogram or something on feature counts
            2. Target (What we are trying to predict)
                a. unacc - unacceptable, acc - acceptable, good, vgood
            3. Rest are features/attributes
    c. Why did we do it?
        i. We wanted to analyze Top-Down Greedy algorithm approaches.
            1. In this one, we use binary trees
            2. Talk about what top-down greedy approach is
                a. Why greedy?
                    i. At every node, we use either information gain/gini index to decide what to split on
            3. Talk about recursion
    d. Explanation of the algorithms
        i. ID3 (pre cursor to C4_5) - MitchellDT.pdf
            1. Hypothesis space of ID3
            2. Inductive bias of ID3
            3. Talk about statistical measures such as
                a. Entropy (add formula)
                b. Information gain (add formula)
            4. Contains the time complexity
            5. Requires discrete features
        ii. C4_5 - MitchellDT.pdf slide 21
            1. Extension to ID3. Quickly discuss. (2 sentences).
                a. Find something online to quickly discuss
            2. Handles both continuous (numerical) and discrete values and still uses entropy as splitting criterion
            3. OrderOfGrowth paper contains big O
            4. We didn't create this one algorithm because of time duration

iii. CART (OrderOfGrowth paper and MitchellDT.pdf, find one more that covers this deeper)
    1. Talk about gini-index (add in the formula)
    2. OrderOfGrowth paper contains the big O

e. Implementation
    i. Data Structures that you used
        1. ID3 vs Cart
    ii. Data transformations
        1. Binarized all features
            a. Buying -> med to low & vhigh -> high
            b. Maint -> med to low & vhigh -> high
            c. Doors -> all cars > 2 doors one bin, and cars w/ 2 doors is in other bin
            d. # Persons car can hold -> half of 4 persons goes into 2 persons and other half into more persons
            e. lug_boot: half of med goes into small and other half of med into big
            f. Safety: half of medium goes into low and other half into high
            g. Target: Unacceptable vs Acceptable car
            h.

```
[low     660
 high    636
 Name: 0, dtype: int64,
 low     648
 high    648
 Name: 1, dtype: int64,
 >2    963
 2     333
 Name: 2, dtype: int64,
 2       655
 more    641
 Name: 3, dtype: int64,
 big      657
 small    639
 Name: 4, dtype: int64,
 low     654
 high    642
 Name: 5, dtype: int64,
 unacc    910
 acc      386
 Name: 6, dtype: int64]
```

    iii. Mini step
        1. Show some code
            a. Possibly the calculation of or entropy/gini calculation or show the recursion itself
            b. Talk about what the mini step is.
    iv. Measuring performance
            a. ROC curve or show confusion matrix or both
    v. Algorithm benchmark
        1. Comparison of ID3, Cart, sklearn Cart
            a. Histogram of auc
    vi. Results
    vii. Limitations of DT

1. Robust to outliers but prone to overfitting (MitchellDT.pdf)
2. Low bias but high variance in bias-variance tradeoff
    a. A small change in the data can cause a large change in the structure of the decision tree causing instability.

viii. Conclusion

1. Avenues for future research
    a. Modifications/Better Approaches
        i. Pruning used to avoid overfitting
        ii. Random Forest
            1. Bootstrap Aggegration (Bagging)
            2. Bad part: Loose interpretability
2. What we each learned
    a. Fluff

ix. Reference

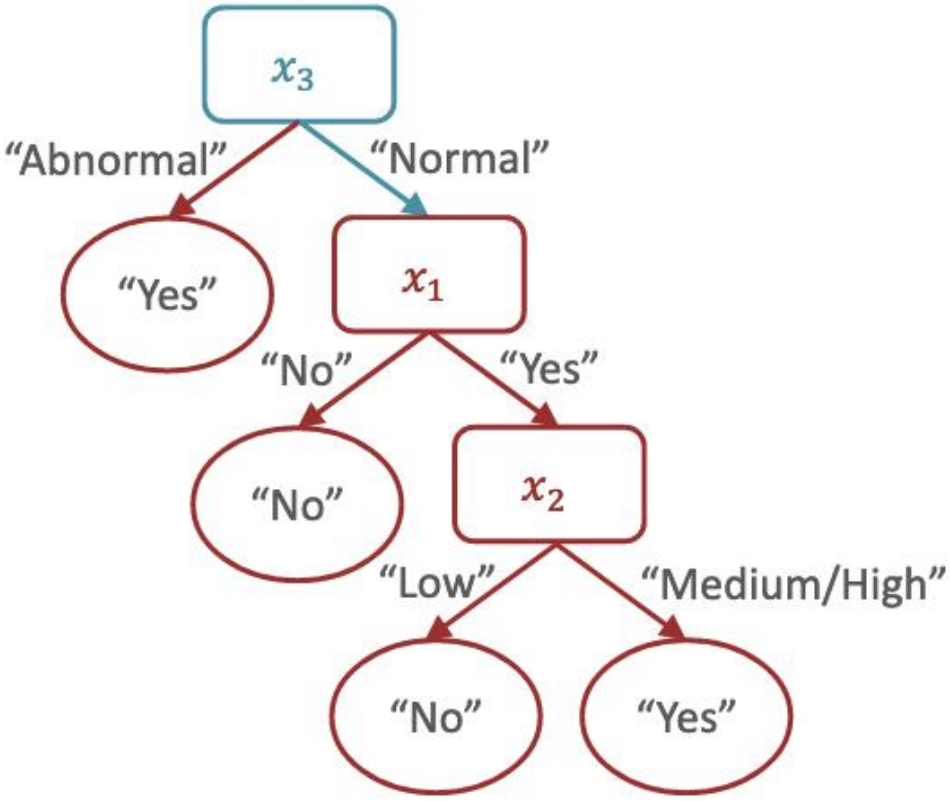
1. Put references everywhere

x. Appendix

1. Code belongs here
2. Use monospace

Using Top-Down Greedy Strategy algorithms to build, compare, and apply decision trees to different problem

sets.



Project Context

After a few weeks of thought process, our team members gathered a wide range of intriguing subjects for our algorithm's final project. During the discussion, we talked about each of the topics, and in the end, we agreed that a core understanding we wanted to leave with from this course was recursion, trees, graph traversals, and greedy approaches. A natural fit that falls right within this domain includes Decision Trees. Therefore, we will take a deep dive into what a Decision tree is, different algorithms commonly used to build a Decision tree, and their applications.

1. Karan Shah: My background is Data Science at Carnegie Mellon University. I worked in industry as a Data Scientist as well, and in order to take many machine learning algorithms from conception and deploy them in end-to-end production with MLOPs requires having a core understanding of algorithms (aside from knowledge needed in machine learning theory/application, distributed systems, API's, containerization, cloud vs On-Prem architecture, etc.). Algorithms will be a solid stepping stone for more advanced areas I plan to go into which may include Advanced Algorithms, Robotics, Comp. Vision, etc. Therefore, in this project, I will supervise the rest of the group with my background knowledge in this domain along with the material from the Machine Learning course I am currently taking at NEU. I will provide the mathematical foundations needed to understand CART/ID3 (precursor of C4.5 algorithm) and attempt to bridge the gap between the math and implementation.

2. Ameya Santosh Gidh: I have a background in mechanical engineering. I love the subject of algorithms and how it is used to solve the daily tasks. Finding the appropriate data structures and solving complex problems efficiently is what I have learnt from this project. I am interested in probabilities and statistics and on how the kind of algorithms I learnt in this course, I can use to solve real life problems. One of the fundamental work of a computer is to make decisions and thus learning one of the fundamental algorithm of decision trees is important for us to understand how the trees we studied in our algorithm class can help us in real life situations. For me I love to analyze data on historical data like politics and education.

3. Mattia Contestabile: My previous studies were concentrated on Philosophy and Political Economy, and my interests for algorithms that optimizes a wide range of tasks has grown naturally during this algorithms class,. The core understanding how we choose the best set of actions has real implications not only on our individual life but as also in our collective lives. Using graph traversal algorithms such as Breadth First Search, Dijkstra's algorithm, Kruskal's Minimum Spanning Tree provides us powerful tools to explore solutions that can optimize real life scenarios like increasing delivery efficiency, finding the shortest greedy algorithms for allocating just enough resources to achieve that. My aim is to be able to delve into the decision tree application by understanding how to build a CART algorithms that is required for building a decision tree. This is not only for the sake of this project but also for my interest decision analysis. At the same time I aim to work with the team knowing that we have to be time sensitive due to the ongoing homework deadlines for CS5800 and other courses coupled with the upcoming synthesis.

4. Jose Lou: I have a degree in Business Management(with a focus on finance) and MIS. From the time that I graduated back in 2007, I have held many different posts and wore many different hats in various different industries. I had worked as a business developer,

financial planning and analysis, and risk analysis for multinational corporations and banks. When I finally made enough money and had an opportunity to create my own business, I did so at the age of 25. Two years after business has stabilized, I felt like I still could do more with my time and went to Med School while running my business. Now, I am yet again embarking on a new academic journey, which could be my last stint in academia as I am not getting any younger and the older you get, the more responsibilities and obligations life thrusts at you. This time in the field of technology, Computer Science to be more precise. With my background in various different fields, I have gleaned a unique understanding and perspective of their unique characteristics. Thus, I believe that I am in a unique position to use these various different angles to provide unconventional methods and solutions to problems and hopefully use and leverage the newfound skills I am gaining from this degree and tie it in to my previous work and life experiences. Algorithms is essential to CS because it provides us with various different techniques or approaches to solving a specific problem in the most efficient time. This is very important to aspiring software engineers or computer scientists as we develop software that is powering the modern world, and research ways to further improve upon existing knowledge. I am very interested in working and developing AI products after graduation. Novel things from self driving cars, autonomous robots, and AI assistants. I feel like automation is our future, and I want to be part or a catalyst to that change. The topic my group wanted to investigate is Dijkstra and Kruskal's algorithm which I agreed with because it fit in with my goal. These algorithms are considered to be one of the best and most used when it comes to find the Shortest Path. The concept is simplistic yet very powerful and has a lot of practical applications in the real world.

Clearly Defined Questions

Decision trees are very interesting in nature because they are a non-parametric supervised learning technique that can be used for regression and classification settings. We will be using them in the classification setting. Applications for these probabilistic in nature trees include (but not limited to) diagnosis of diseases, cost analysis, etc. Specifically, we will be investigating the ID3 (pre-cursor to C4.5 algorithm) and CART algorithm. Given our time constraint, we will not be able to address all questions on this subject and all the different algorithms available as well. We selected ID3 and CART because these are commonly taught and used in industry. We will conduct research on these algorithms in order to answer some of the following questions:

- What is a decision tree?

- What are the main differences between the two algorithms? (i.e. traversal, choosing attributes, etc.)
- What are the benefits and drawbacks involved in each algorithm?
- Are these algorithms resulting in a global optimized solution? If not, we will research if there is anything relevant in this space.
- How can we extend the concept of decision trees for better results?

Project Scope

The scope of our project is to analyze Top-Down Greedy algorithms used to make decision trees. We will maintain these trees to essentially be binary trees. Furthermore, as aforementioned above, we will investigate what a decision tree is truly composed of, how these algorithms work (similarities/differences and benefits/drawbacks), extension of these approaches, and if this will result in a global optimal solution. Aside from trying to answer these questions, our aim also is to explore possible direct applications of decision trees using these algorithms on real world datasets. For now, we have a politicians and educations dataset, but this may be altered near final submission. While we evaluate these techniques on the dataset, we will consider whether one was better or worse ultimately.

Project Progress Description

So far, we have hopped on teams and done the following:

- Determine the topic of the project
- Discuss the clearly defined questions (may add or remove 1 or 2)
- Narrowed the scope and will continue narrowing further if need be
- Outlined the roles everyone will play
- Reviewed key strengths/weaknesses of different members in group
- Reviewed background context needed for project As we move forward, this will be our plan of attack:
- Split the tasks up between the different members
- Perform additional research investigating decision trees, ID3, and CART
- Analyze the benefits and drawbacks of the two algorithms along with their similarities and difference. Further analyze the correctness and complexity of the two algorithms.
- Evaluate performance of each algorithm on difference datasets and present results
- Make sure we have answered the previous questions and stayed consistently within scope of problem

- Discuss further extension of decision trees by altering possibly the learning algorithm but still remaining within confines of hypothesis space. If time permits, we will code up, apply, and present results this as well.
- Gather all information together, complete final report, submit presentation videos/slides, and any additional information.

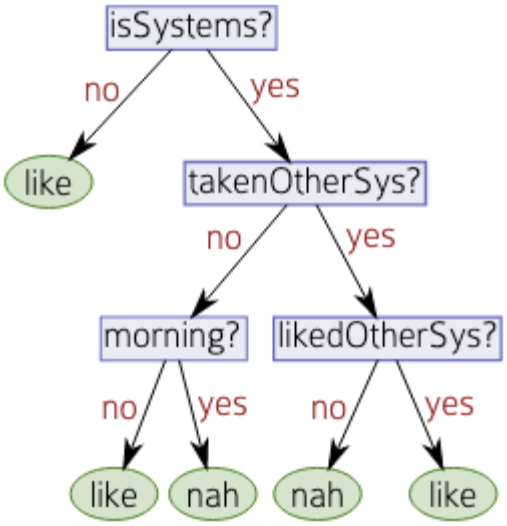
Deep Analysis

What is a decision tree?

-) It's a Tree model based on Divide and Conquer Algorhitm
-)It is a powerful tool to help in decision-making by corroborating existing dataset to elaborate possible the best possible set of decisions.
- Part of the process is understanding how to ask the right set of questions, this will contribute to refining the dataset with elite splits that can eventually lead to refining a decision result or an action based on this decision tree classification.
- A decision Tree is a tool that can be used in an everyday setting, we are constantly making decisions in our life but that does not mean those decisions are necessarily correct. A decision tree takes a dataset and needs carefully designed questions that

can help the algorithm ‘split’ the **features** by using the right **features values**

- **Features:** Are the questions are we feeding to our Decision Tree
- **Features Values**: These are the responses/answers to this question
- **Label:** This refers to the rating, ideally to classify the accuracy.
- It is impossible to anticipate the right questions for a given dataset, snce the nature of the features could be very different and vary from different angles and interepretations. To avoid this problem, Data scientists use a greedy method to choose the right question to ask for each layer when it comes to building the decision tree.
-
-



What did we do?