Machine Learning
CAP 5610

Instructor: Dr. Mengxin Zheng
TA 1: Shenyang Liu
TA 2: Mayank Kumar
Department of Computer Science
University of Central Florida (UCF)

Spring, 2025

January 17, 2025

Assignment #1

Submission Instructions

Please submit your solutions via Canvas. You should submit your assignment as a PDF file. Please do not include blurry scanned/photographed equations as they are difficult for us to grade.

Late Submission Policy

The late submission policy for assignments will be as follows unless otherwise specified:

1. 75% credit within 0-48 hours after the submission deadline.
2. 50% credit within 48-96 hours after the submission deadline.
3. 0% credit after 96 hours after the submission deadline.

Decision Tree

1 Decision Tree Basics [30 pts]

The goal of this assignment is to test and reinforce your understanding of Decision Tree Classifiers.

(a) [5 pts] How many unique, perfect binary trees of depth 3 can be drawn if we have 5 attributes? By depth, we mean the depth of the splits, not including the nodes that only contain a label (see Figure 1). So, a tree that checks just one attribute is a depth one tree. By perfect binary tree, we mean every node has either 0 or 2 children, and every leaf is at the same depth. Note also that a tree with the same attributes but organized at different depths is considered "unique". Do not include trees that test the same attribute along the same path in the tree.

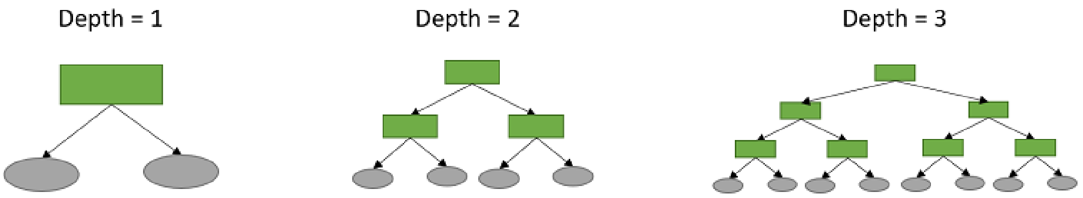

Figure 1: Example of perfect binary trees with different depths.

[5 pts] In general, for a problem with A attributes, how many unique perfect D depth trees can be drawn? Assume A >> D

(b) [10 pts] Consider the following dataset from Table 1 for this problem. Given the five attributes on the left, we want to predict if the student got an A in the course. Create 2 decision trees for this dataset. For the first, only go to depth 1. For the second go to depth 2. For all trees, use the ID3 entropy algorithm from class. For each node of the tree, show the decision, the number of positive and negative examples and show the entropy at that node.

Hint: There are a lot of calculations here. You may want to do this programatically.

Early	Finished HMK	Senior	Likes Coffee	Liked The Last Jedi	A
1	1	0	0	1	1
1	1	1	0	1	1
0	0	1	0	0	0
0	1	1	0	1	0
0	1	1	0	0	1
0	0	1	1	1	1
1	0	0	0	1	0
0	1	0	1	1	1
0	0	1	0	1	1
1	0	0	0	0	0
1	1	1	0	0	1
0	1	1	1	1	0
0	0	0	0	1	0
1	0	0	1	0	1

Table 1: Toy Data-set for Task 1: Decision Tree Basics.

(c) [5 pts] Make one more decision tree. Use the same procedure as in (b), but make it depth 3. Now, given these three trees, which would you prefer if you wanted to predict the grades of 10 new students who are not included in this data-set? Justify your choice.

(d) [5 pts] Consider a new definition of a "realizable" case: "For some fixed concept class C, such as decision trees, a realizable case is one where the algorithm gets a sample consistent with some concept c ∈ C. In other words, for decision trees, a case is realizable if there is some tree that perfectly classifies the data-set.

If the number of attributes A is sufficiently large, under what condition would a dataset not be realizable for decision trees of no fixed depth? Prove that the dataset is unrealizable if and only if that condition is true.

2 Application of Decision Tree on Real-Word Data-set [25 pts]

In this task, you will build a decision tree classifier using a real-world data set called Census-Income Data Set, available publicly for downloading at Dataset. This data set contains weighted census data extracted from the 1994 and 1995 Current Population Surveys conducted by the U.S. Census Bureau. The data contains 41 demographic and employment-related variables.

Basic statistics for this data set are provided below.

- Number of instances data = 199523
- Duplicate or conflicting instances: 46716
- Number of instances in test = 99762
- Duplicate or conflicting instances: 20936
- Class probabilities for income-projected.test file
- Probability for the label '- 50000': 93.80
- Probability for the label '50000+': 6.20
- Majority accuracy: 93.80% on value - 50000
- Number of attributes = 40 (continuous: 7 nominal: 33)
- Information about .data file :
  - 91 distinct values for attribute #0 (age) continuous
  - 9 distinct values for attribute #1 (class of worker) nominal
  - 52 distinct values for attribute #2 (detailed industry recode) nominal
  - 47 distinct values for attribute #3 (detailed occupation recode) nominal
  - 17 distinct values for attribute #4 (education) nominal
  - 1240 distinct values for attribute #5 (wage per hour) continuous
  - 3 distinct values for attribute #6 (enroll in edu inst last wk) nominal
  - 7 distinct values for attribute #7 (marital stat) nominal
  - 24 distinct values for attribute #8 (major industry code) nominal
  - 15 distinct values for attribute #9 (major occupation code) nominal
  - 5 distinct values for attribute #10 (race) nominal
  - 10 distinct values for attribute #11 (Hispanic origin) nominal
  - 2 distinct values for attribute #12 (sex) nominal
  - 3 distinct values for attribute #13 (member of a labor union) nominal
  - 6 distinct values for attribute #14 (reason for unemployment) nominal

- 8 distinct values for attribute #15 (full or part-time employment stat) nominal
- 132 distinct values for attribute #16 (capital gains) continuous
- 113 distinct values for attribute #17 (capital losses) continuous
- 1478 distinct values for attribute #18 (dividends from stocks) continuous
- 6 distinct values for attribute #19 (tax filer stat) nominal
- 6 distinct values for attribute #20 (region of previous residence) nominal
- 51 distinct values for attribute #21 (state of previous residence) nominal
- 38 distinct values for attribute #22 (detailed household and family stat) nominal
- 8 distinct values for attribute #23 (detailed household summary in the household) nominal
- 10 distinct values for attribute #24 (migration code-change in MSA) nominal
- 9 distinct values for attribute #25 (migration code-change in reg) nominal
- 10 distinct values for attribute #26 (migration code-move within reg) nominal
- 3 distinct values for attribute #27 (live in this house one year ago) nominal
- 4 distinct values for attribute #28 (migration prev res in sunbelt) nominal
- 7 distinct values for attribute #29 (num persons worked for the employer) continuous
- 5 distinct values for attribute #30 (family members under 18) nominal
- 43 distinct values for attribute #31 (country of birth father) nominal
- 43 distinct values for attribute #32 (country of birth mother) nominal
- 43 distinct values for attribute #33 (country of birth self) nominal
- 5 distinct values for attribute #34 (citizenship) nominal
- 3 distinct values for attribute #35 (own business or self-employed) nominal
- 3 distinct values for attribute #36 (fill inc questionnaire for veteran's admin) nominal
- 3 distinct values for attribute #37 (veterans benefits) nominal
- 53 distinct values for attribute #38 (weeks worked in year) continuous
- 2 distinct values for attribute #39 (year) nominal

- Classes: - 50000, 50000+.

One instance per line with comma-delimited fields. There are 199,523 instances in the data file and 99,762 in the test file.

The data was split into train/test in approximately $\frac{2}{3}$, $\frac{1}{3}$ proportions using MineSet's MIndUtil mineset-to-mlc. Below are your tasks:

(a) [10 pts] Train a decision tree classifier using the data file. Vary the cut-off depth from 2 to 10 and report the training accuracy for each cut-off depth k. Based on your results, select an optimal k.

(b) [8 pts] Using the trained classifier with optimal cut-off depth k, classify the 99,762 instances from the test file and report the testing accuracy (the portion of testing instances classified correctly).

(c) [7 pts] Do you see any over-fitting issues for this experiment? Report your observations.

Naive Bayes Classifiers

3 Independent Events and Bayes Theorem [20 pts]

(a) [5 Points] For events A, B prove:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|\neg A)P(\neg A)}$$

($\neg A$ denotes the event that A does not occur.)

(b) Let X, Y , and Z be random variables taking values in {0, 1}. The following table lists the probability of each possible assignment of 0 and 1 to the variables X, Y , and Z:

	Z = 0		Z = 1	
	X = 0	X = 1	X = 0	X = 1
Y = 0	0.1	0.05	0.1	0.1
Y = 1	0.2	0.1	0.175	0.175

(a) [5 Points] Is X independent of Y ? Why or why not?

(b) [5 Points] Is X conditionally independent of Y given Z? Why or why not?

(c) [5 Points] Calculate $P(X \neq Y | Z = 0)$.

4 Implementing Naive Bayes [25 pts]

You will now learn how to use Naive Bayes Algorithm to solve a real-world problem: text categorization. Text categorization (also referred to as text classification) is the task of assigning documents to one or more topics. For our homework, we will use a benchmark dataset that is frequently used in text categorization problems. This dataset, Reuters-21578, consists of documents that appeared in Reuters newswire in 1987. Each document was then manually categorized into a topic among over 100 topics. In this homework, we are only interested in earn and acquisition (acq) topics, so we will use a shortened version of the dataset (documents assigned to topics other than “earn” or “acq” are not in the dataset provided for the homework). As features, we will use the frequency (counts) of each word that occurred in the document. This model is known as the bag-of-words model and it is frequently used in text categorization. You can download Assignment 2 data from the Canvas. In this folder, you will find:

- **train.csv:** Training data. Each row represents a document, and each column separated by commas represents features (word counts). There are 4527 documents and 5180 words.
- **train labels.txt:** labels for the training data
- **test.csv:** Test data, 1806 documents and 5180 words

- **test labels.txt:** labels for the test data
- **word indices:** words corresponding to the feature indices.

Implement Naive Bayes Algorithm. Train your classifier on the training set that is given and report training accuracy, testing accuracy, and the amount of time spent training the classifier.

Disclaimers: This assignment re-uses some materials from the publicly available website: CMU Introduction to Machine Learning Course, 10-315, Spring 2019. I personally thank Prof. Maria-Florina Balcan for sharing her teaching materials publicly. This assignment is exclusively used for instructional purposes.