COMP307/AIML420 — Fundamentals of AI
Assignment 3: Reasoning under uncertainty

10% of Final Mark COMP307, 8% of Final Mark AIML420

Due date: 11:59 PM - May 15, 2024 (Wednesday)

Objectives

The goal of this assignment is to help you understand the basic concepts and algorithms related to reasoning under uncertainty. **IMPORTANT.** You most not use external libraries (like PyTorch, sklearn, or any others) or any AI Tool (like ChatGPT, CoPilot) to complete this assignment.

Part 1: Naive Bayes Method [60 marks]

This part is to implement the Naive Bayes algorithm, and evaluate the program on the breast cancer dataset to be described below. The program should build a Naive Bayes classifier from the training dataset and apply it to the test set.

Dataset Description

The *breast cancer* dataset is obtained from the UCI machine learning library (`https://archive-beta.ics.uci.edu/ml/datasets/breast+cancer`).

The original dataset consists of 286 instances that belong to two classes: *no-recurrence-events* and *recurrence-events*.

Each instance is described by 9 categorical attributes (features). The name and domain of each attribute is described as follows:

1. **age** (9 values): 10-19, 20-29, 30-39, 40-49, 50-59, 60-69, 70-79, 80-89, 90-99
2. **menopause** (3 values): lt40, ge40, premeno
3. **tumor-size** (12 values): 0-4, 5-9, 10-14, 15-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59
4. **inv-nodes** (13 values): 0-2, 3-5, 6-8, 9-11, 12-14, 15-17, 18-20, 21-23, 24-26, 27-29, 30-32, 33-35, 36-39
5. **node-caps** (2 values): yes, no
6. **deg-malig** (3 values): 1, 2, 3
7. **breast** (2 values): left, right
8. **breast-quad** (5 values): left_up, left_low, right_up, right_low, central
9. **irradiat** (2 values): yes, no

The original dataset had some missing values. After removing instances with missing values 277 instances remained. These instances were split into the following training and test datasets as follows.:

- **267 training instances**: 189 *no-recurrence-events* + 78 *recurrence-events*.
- **10 test instances**: 7 *no-recurrence-events* + 3 *recurrence-events*.

The datasets are available in the `breast-cancer-training.csv` and `breast-cancer-test.csv` files.

Requirements

Your job is to implement and use the Naive Bayes classifier to classify the test instances in the `breast-cancer-test.csv` file.

The pseudo code of the training is given as follows to obtain the (conditional) probabilities of each feature given the class, and the probabilities of each class.

Algorithm 1: Training of the Naive Bayes Classifier

```
Input: The training set.
Output: A probability table.
// Initialise the count numbers to 1.
for each class label $y$ do
    $count(y) = 1$;
    for each feature $X_i$ do
        for each possible value $x_i$ of feature $X_i$ do
            $count(X_i, x_i, y) = 1$;
// Count the numbers of each class and feature value based on the training
   instances.
for each training instance $[X_1 = x_i, \dots, X_n = x_n, Y = y]$ do
    $count(y) = count(y) + 1$;
    for each feature $X_i$ do
        $count(X_i, x_i, y) = count(X_i, x_i, y) + 1$;
// Calculate the total/denominators.
$class\_total = 0$;
for each class label $y$ do
    $class\_total = class\_total + count(y)$;
    for each feature $X_i$ do
        $total(X_i, y) = 0$;
        for each possible value $x_i$ of feature $X_i$ do
            $total(X_i, y) = total(X_i, y) + count(X_i, x_i, y)$;
// Calculate the probabilities from the counting numbers.
for each class label $y$ do
    $prob(y) = count(y)/class\_total$;
    for each feature $X_i$ do
        for each possible value $x_i$ of feature $X_i$ do
            $prob(X_i, x_i, y) = count(X_i, x_i, y)/total(X_i, y)$;
return $prob$;
```

For the prediction of each test instance, you need to calculate the score of the test instance for each class, and predict the class with the largest score. The score of a class is calculated as follows.

Algorithm 2: Calculation of the class score.

```
Input: A test instance $[X_1 = x_1, \dots, X_n = x_n]$, a class label $y$, the probability table $prob$.
Output: The score.
$score = prob(y)$;
for each feature $X_i$ do
    $score = score * prob(X_i, x_i, y)$;
return $score$;
```

You should implement the Naive Bayes method from scratch (not call it from any machine learning library). Your program should take two file names as command line arguments, construct a classifier from the data in the first file, and then apply the classifier to the data in the second file.

You may write the program code preferrably in `Java, Python`, but other programming languages are accepted as well.

You should submit the following files electronically and also a report.

- (40 marks) `Program code` for your Naive Bayes Classifier (both the source code and the executable program running on ECS School machines),
- (4 marks) `sampleoutput.txt` containing the output of your program on the test dataset, and
- (16 marks) `A report` in PDF, text or DOC format. The report should include:
  1. The conditional probabilities $P(X_i = x_i|Y = y)$ for each feature X_i (e.g., age), its possible value x_i (e.g., 10-19), and each class label $Y = y$ (y can be *no-recurrence-events* or *recurrence-events*).
  2. The class probabilities $P(Y = y)$ for each class label $Y = y$.
  3. For each test instance, given the input vector $\mathbf{X} = [X_1 = x_1, \ldots, X_9 = x_9]$, give the calculated
     – $score(Y = \textit{no-recurrence-events}, \mathbf{X})$,
     – $score(Y = \textit{recurrence-events}, \mathbf{X})$,
     – predicted class of the input vector.

Part 2: Building Bayesian Network [40 marks]

This part is to build a Bayesian Network for the problem described below.

Problem Description

Dr. Eve is a Lecturer, who lives far away from her university. So, she prefer to work from home and only comes to her office if she has research meetings with her postgraduate students, or teaching lectures for undergraduate students, or she has both meetings and teaching:

- The probability for Eve to have meetings is 70%, and the probability of Eve having lectures is 60%.
- If Eve has both meetings and lectures, the probability of Eve coming to her office is 95%.
- When Eve only has meetings (without lectures), the probability of her coming to the office is 75% as she can connect with her students via Zoom.
- When Eve has only lectures (without meetings), there is an 80% chance that she will come to her office.
- If Eve has neither meetings nor lectures, there is only a 6% chance that she will come to the office.
- When Eve is in her office, half the time her light is off as she tries to hide from others to get work done quickly.
- When she is not in her office, she leaves her light on only 2% of the time.
- 80% of the time when Eve is in her office, she is logged onto the computer.
- Since she occasionally works from home, she is logged onto the computer 20% of the time when she is not in her office.

Note regarding the calculation, you should show your *working process* of the calculation to demonstrate *your understanding.*

Requirements

1. (10 marks) Construct a Bayesian network to represent the above scenario. (*Hint: First decide what your domain variables are; these will be your network nodes. Then decide what the causal relationships are between the domain variables and add directed arcs in the network from cause to effect. Finally, you have to add the prior probabilities for nodes without parents, and the conditional probabilities for nodes that have parents.*)

2. (2 marks) Calculate the number of free parameters in your Bayesian network.

3. (8 marks) What is the *joint* probability that Eve has lectures, has no meetings, she is in her office and logged on her computer but with lights off.

4. (8 marks) Calculate the probability that Eve is in the office.

5. (12 marks) If we know that Eve is in her office, what is the *conditional* probability that she is logged on, but her light is off.

Part 3: Bayesian Network: Applications [For AIML420 ONLY, 20 marks]

Identify a real-world application (**different from the examples given in this assignment and the lectures**) that can be described using Bayesian network. There should be at least 5 random variables in this Bayesian network.

In your report, you should:

1. Clearly define the random variables and their domains.

2. Clearly describe their relationships (using plain language).

3. Draw the Bayesian network that can reflect the described relationship.

4. Write the factorisation of the Bayesian network.

1 Relevant Data Files

The relevant data files, information files about the data sets, and some utility program files can be found as a .zip file on the course homepage (See Assignment 3: `https://ecs.wgtn.ac.nz/Courses/COMP307_2024T1/Assignments`).

1.1 Submission Method

The programs and the report should be submitted through the web submission system from the COMP307 or AIML420 course web site **by the due time**. Please make sure you submit to the course you are enrolled in.

Notice that you can submit a .zip file to preserve the subdirectory structure you might have created to organise your submission.

Please check **again** that your programs can be run on the ECS machines easily according to your readme. If the tutors can't run your code, you may **lose marks!** Each tutor has a limited amount of time (¡ 5 minutes) to get your code running, so please don't ask them to use Pycharm, IntelliJ IDEA, Visual Studio, etc to run your code. All these IDEs support exporting runnable code.

1.2 Late Submissions

The assignment must be submitted on time unless you have made a prior arrangement with the course **co-ordinator** or have a valid medical excuse. This year, we are using the ECS extension system for all extension requests. Please make a request there if you think you have a valid reason.

1.3 Plagiarism

Plagiarism in programming (copying someone else's code) is just as serious as written plagiarism, and is treated accordingly. Make sure you explicitly write down where you got code from (and how much of it) if you use any other resources asides from the course material. Using excessive amounts of others' code may result in the loss of marks, but plagiarism should result in zero marks!