Tutorial Revision Week

1. This question reviews the rules of propositional calculus and uses the model checking algorithm to test entailment in a simple logical knowledge base.
   (a) Explain what a proposition is and how we can combine them.
   (b) Describe the most fundamental logical connectives of propositional calculus and provide self-made examples related to a single context for each connective.
   (c) Let P, Q, and R be valid propositions. Construct the truth table for $X = (P \wedge Q) \vee (\neg Q \wedge R)$.
   (d) Explain what a knowledge base is. Assume a knowledge base consisting of the following propositions:
      1. It is raining.
      2. Cloud coverage is extensive.
      3. Weather warning is issued.
      4. If it is raining and cloud coverage is extensive, then a weather warning is issued (rule).

      Does this knowledge base entails that if a weather warning is not issued, then it is not raining or cloud coverage is not extensive?

Solution:

(a) A **proposition** p is a statement that can only be either true (T, or 1) or false (F, or 0). We can use **logical connectives** to connect two or more propositions; therefore, logical connectives allow us to form complex statements from atomic (or single) propositions.

(b) The five most fundamental logic connectives are:

- the logical disjunction ($\vee$), which has the meaning of "or", i.e., it is true if either one or the other proposition is true;
- the logical conjunction ($\wedge$), which has the meaning of "and" when combining propositions and it is only true if both propositions are true;
- the conditional (or implication) statement ($\implies$), that can be read as "if P then Q", and only returns false if the condition (P) is true but the conclusion (Q) is false;
- the biconditional proposition (or equality) ($\iff$), that can be read as "if and only if", and will return true only if both propositions are either true or false together;
- the logical negation ($\neg$), which inverts the truth value of a proposition. (Note, this is the only logical connective that doesn't combine two propositions.)

For the second part of the question, you can use any proposition you wish to illustrate the use of the connectives above. One specific example to illustrate what the problem asked would be: Let D = "I like AI", C = "I will pass this module", and S = "I will do my assignments." Then,

- I like AI and I will pass this module can be represented as: $D \wedge C$
- I will do my assignments or I will not pass this module would be: $S \vee \neg C$
- If I won't do my assignments, then I won't pass this module: $\neg S \implies \neg C$
- I will pass this module if and only if I will do my assignments: $C \iff S$

(c) The truth table for the proposed composed proposition is below. As the composed proposition depends on 3 atomic propositions, there are $2^3 = 8$ possible cases to be analyzed:

P	Q	R	$P \wedge Q$	$\neg Q$	$\neg Q \wedge R$	X
F	F	F	F	T	F	F
F	F	T	F	T	T	T
F	T	F	F	F	F	F
F	T	T	F	F	F	F
T	F	F	F	T	F	F
T	F	T	F	T	T	T
T	T	F	T	F	F	T
T	T	T	T	F	F	T

(d) We can think of a **knowledge base** as a set of propositions expressing facts and logical rules, and whose truth value expresses the conjunction of all propositions it contains.
Assuming the KB given, and denoting propositions 1, 2, and 3 as P, Q, and S, we can express the rule (proposition 4) as $(P \wedge Q) \implies S$. Similarly, the test proposition can be expressed as $T = \neg S \implies (\neg P \vee \neg Q)$.
To test whether the KB $\models T$, we have to compute the models using exhaustive search (Algorithm 8.1 of the lecture notes). As the KB contains 3 independent propositions, there are $2^3 = 8$ possible models for these binary propositions, which are given in the first 3 columns of the table below.

P	Q	S	$(P \wedge Q)$	$(P \wedge Q) \implies S$	$\neg S$	$\neg P$	$\neg Q$	$(\neg P \vee \neg Q)$	$T = \neg S \implies (\neg P \vee \neg Q)$
F	F	F	F	T	T	T	T	T	T
F	F	T	F	T	F	T	T	T	T
F	T	F	F	T	T	T	F	T	T
F	T	T	F	T	F	T	F	T	T
T	F	F	F	T	T	F	T	T	T
T	F	T	F	T	F	F	T	T	T
T	T	F	T	F	T	F	F	F	F
T	T	T	T	T	F	F	F	F	T

From the table above, we can see that the set of models where the KB holds is explicitly given by

$$M(KB) = \{[F,F,F],[F,F,T],[F,T,F],[F,T,T],[T,F,F],[T,F,T],[T,T,T]\},$$

which happens to be the same set of models where T holds (last column of the table). Since $M(KB) = M(T)$, $M(KB) \subseteq M(T)$ and, therefore, $KB \models T$ is true, i.e., if a weather warning is not issued, it can be concluded that either it is not raining or cloud coverage is not extensive, according to this weather forecasting logical system.

2. This question reviews clustering as an unsupervised machine learning problem and applies the K-means clustering algorithm to a small data clustering problem.

   (a) Briefly explain what an unsupervised ML problem is and how it differs from a supervised ML problem.

   (b) Describe the steps of the K-means clustering algorithm to generate two clusters from 5 data samples with two features each.

   Consider the following two-dimensional data which has been divided into two clusters as shown:

   - Cluster 1: $[2,-3]^T, [2,1]^T, [3,2]^T$
   - Cluster 2: $[-3,1]^T, [-3,2]^T$

   (c) Given the above information, determine which cluster a new point $(0,0)$ belongs to by using the K-means clustering algorithm.

(d) What would be the updated centroid of the clusters after including the new point?

Solution:

(a) In an **unsupervised** ML problem, the data we are given do not have any associated labels, which is exactly the main difference from a supervised ML problem. In the latter, the data is accompanied by labels that one can use to estimate the best model parameters in an ML problem by optimizing an error function, $F(w)$.

(b) Given an initial guess for each cluster centroid (*initialization* step), μ_1 and μ_2, we can use these guesses to compute the configuration matrix, i.e., find the best cluster to assign each data point to. We can accomplish this by computing a distance (e.g., the Euclidean distance) of the data point to each cluster centroid and assigning the data point to the centroid that minimizes this distance (*update configuration* step). This new configuration can then be used to get a better guess of the centroids (*update centroids* step). We can compare the new configuration with the previous configuration, and exit if both configurations match (*convergence* check), or go back to the *update configuration* step for a new iteration.

(c) As described in the algorithm above, the assignment is performed by minimizing the distance between the data point and each centroid. Considering the two clusters, the centroid for each cluster is given by:

$$\mu_1 = \left(\frac{2+2+3}{3}, \frac{-3+1+2}{3}\right) = (2.3, 0)$$
$$\mu_2 = \left(\frac{-3-3}{2}, \frac{1+2}{2}\right) = (-3, 1.5)$$

The Euclidean distance for the new data point, $x_n = (0, 0)$, to each centroid is:

$$||x_n - \mu_1||^2 = (0 - 2.3)^2 + (0 - 0)^2 = 5.44$$
$$||x_n - \mu_2||^2 = (0 - (-3))^2 + (0 - 1.5)^2 = 9 + 2.25 = 11.25.$$

We can see that the new point is closer to the centroid of cluster 1 than cluster 2. Therefore, the new data point will be assigned to cluster 1.

(d) Given that cluster 2 was not modified, its centroid will be the same. For cluster 1, we would have to consider the new data point to compute the new centroid. As the point added is $(0, 0)$, the centroid will be:

$$\mu_1 = \left(\frac{2+2+3+0}{4}, \frac{-3+1+2+0}{4}\right) = (1.75, 0).$$

Therefore, $\mu_1 = (1.75, 0)$ and $\mu_2 = (-3, 1.5)$.

3. This question reviews probabilistic classification using Bayes' theorem.

(a) Describe the difference between a naïve Bayes' classifier versus a standard Bayes' classifier.

(b) Consider the following example data available about what kind of pizza your friends would find delicious:

Pizza ID	Toppings	Spicy	Temperature	Crust	Delicious
1	Yes	Yes	Mild	Thick	Yes
2	Yes	Yes	Hot	Thin	Yes
3	No	No	Hot	Thin	Yes
4	No	Yes	Mild	Thick	No
5	Yes	No	Hot	Thick	Yes
6	No	No	Hot	Thick	No

By finding the model parameters, and by computing the maximum a-posteriori (MAP) estimate with Naïve Bayes assumption, predict the outcome for pizza 7:

Pizza ID	Toppings	Spicy	Temperature	Crust	Delicious
7	No	No	Mild	Thick	?

Solution:

(a) A standard Bayes' classifier uses Bayes' theorem to select the value (outcome) of the random variable Y which maximizes the posterior probability of Y after seeing the input feature data, X, i.e., $P(X|Y)$. Therefore, probabilistic Bayesian classification requires knowing the likelihood distribution, $P(X|Y)$, and estimating these values can be challenging if different features depend on one another.
The naïve Bayes' classifier makes the assumption that each feature is conditionally independent of the others, given the class. This makes the Markov factorization for a naïve Bayes' classifier simpler, as

$$P(X|Y) = P(X^1|Y) \times P(X^2|Y) \times ... \times P(X^D|Y),$$

where X^d refers to the dimension of the input feature data.

(b) The MAP decision rule consists of selecting the value of Y which maximizes the posterior probability $P(Y = y|X = x)$, which is equivalent to:

$$y^\star = \arg\max_{y \in \Omega_Y} P(X = x|Y = y)P(Y = y).$$

We can use the example data given to estimate the prior and likelihood probabilities for each class we want to predict ($Y = y, n$ for the "delicious" outcome. Let's denote the features "toppings", "spicy", "temperature", and "crust", respectively, as RVs $T = t$ for $t \in \{y, n\}$, $S = s$ for $s \in \{y, n\}$, $P = p$ for $p \in \{m, h\}$, and $C = c$ for $c \in \{tk, th\}$. Then,

$$P(Y = y) = \frac{4}{6} = \frac{2}{3} = 0.67, P(Y = n) = \frac{2}{6} = \frac{1}{3} = 0.33$$
$$P(T = n|Y = y) = \frac{1}{4} = 0.25, P(T = n|Y = n) = \frac{2}{2} = 1$$
$$P(S = n|Y = y) = \frac{2}{4} = 0.5, P(S = n|Y = n) = \frac{1}{2} = 0.5$$
$$P(P = m|Y = y) = \frac{1}{4} = 0.25, P(P = m|Y = n) = \frac{1}{2} = 0.5$$
$$P(C = th|Y = y) = \frac{2}{4} = 0.5, P(C = th|Y = n) = \frac{2}{2} = 1$$

Therefore, the two possibilities outcomes for $y \in \Omega_Y = \{y, n\}$ are:

$$\begin{aligned} y = y : P(Y = y|X = x) =& P(T = n|Y = y)P(S = n|Y = y)P(P = m|Y = y)P(C = th|Y = y)P(Y = y) \\ =& 0.25 \times 0.5 \times 0.25 \times 0.5 \times 0.67 = 0.0105 \\ y = n : P(Y = n|X = x) =& P(T = n|Y = n)P(S = n|Y = n)P(P = m|Y = n)P(C = th|Y = n)P(Y = n) \\ =& 1 \times 0.5 \times 0.5 \times 1 \times 0.33 = 0.0825 \end{aligned}$$

Therefore, the predicted outcome for pizza 7 would be "No" for "Delicious".