analitical

Section A: Short Answer Questions [12 marks]

Answer each of the questions in this section as briefly as possible. Expect to answer each sub-question in a couple of lines.

Question 1: General Machine Learning [12 marks]

1. What is the difference between *marginal* and *conditional probability distributions*? [1 mark]
   Marginal: the probability of a given event occuring without any conditions on other variables.
   Conditional: the probability of an event occuring given other variables.
2. What algorithm is used for efficient *marginalisation* on *probabilistic graphical models* (PGMs)? [1 mark]
   Elimination Algorithm.
3. Name two methods (considered in the class) that can be used to find *maximum likelihood estimate* of *parameters of a probabilistic model*. [1 mark]
   1. Analytic Solution (closed form)
   2. Approximate iterative solution (gradient descent)
4. In what situation do we need to use *expectation maximisation* to train a *PGM* (as opposed to directly doing maximum likelihood)? [1 mark]
   When there are latent variables.
5. State *Bayes' rule*, as it applies to Bayesian modelling, and identify the *posterior*, *likelihood*, *prior* and *marginal likelihood (or evidence)*. Define all mathematical symbols introduced. [1 mark]
   $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$ / P(A|B): posterior, P(B|A): likelihood, P(A): prior / P(B): marginal likelihood
6. Explain the view of *frequentist supervised learning* as an *optimisation problem*. [1 mark]
   ① each model has fixed but hidden model parameters
   ② optimisation allow us to find these parameters
7. Give an example (either using a diagram or description in words) of a *dataset* that a *perceptron* cannot *classify* with perfect accuracy. [1 mark]
   example: (1, 0) -> 1, (0, 1) -> 1 / (0, 0) -> 0, (1, 1) -> 0 / in which the data is not linear separable.
8. Explain the advantage of using a *sigmoid function* rather than a *step function* as the *activation function* in a *neural network*. [1 mark]
   A step function is not continuous, it is not differentiable at x = 0; A sigmoid function is continuous and is differentiable anywhere.
9. Define a *kernel* in the context of *SVM* learning. [1 mark]
   A kernel in the context of SVM is a function that can transform the original feature space to other dimention
10. What are *support vectors*? [1 mark]
    The points on the margin boundaries.
11. Name two applications of *dimensionality reduction*. [1 mark]
    Data compression.
    Reducing computational cost.
    Data visualization (e.g. mapping 3D GPS data to 2D to record car's location)

12. Briefly describe how *active learning* differs from regular *supervised learning*. [1 mark]

Section B: Method Questions [12 marks]

In this section you are asked to demonstrate your conceptual understanding of a subset of the methods that we have studied in this subject.

Question 2: Linear Models [2 marks]

1. What is the difference between *linear regression* and *logistic regression*, in terms of how they make *predictions* (after *training*)? [1 mark]

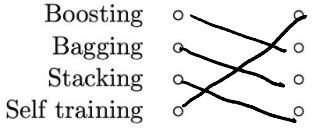
The model of linear regression is based on a Normal distribution while the logistic regression is based on logistic function

2. Why does the algorithm for *training logistic regression* involve *gradient descent*, while *linear regression* does not? [1 mark]

Linear regression has closed form solution but logistic regression doesn't have. So logistic regression has to involve approximate iterative solution (gradient descent)

Question 3: Classifier Combination [2 marks]

Please write the following in your script book, and there connect each dot on the left with one dot on the right, to create the best possible correspondence



Boosting	~~A *semi-supervised learner*~~	[0.5 marks]
Bagging	Focus *base classifiers* on hard *examples*	[0.5 marks]
Stacking	*Bootstrap aggregated ensemble*	[0.5 marks]
Self training	Layer of *learners* feed into next layer	[0.5 marks]

Question 4: Model selection [3 marks]

This question is on *model selection* for machine learning models.

1. Outline how *heldout (validation) data* is used to select between several models, under the *frequentist* paradigm. [1 mark]

randomly

① split the dataset into two part, one for training, one for testing,
② For each model, fit the model with the training set and evaluate the model with test set
③ pick the best model based on the result of the evaluation.

2. With reference to the above, explain why the *training* data cannot be used for *frequentist model selection*. [1 mark]

Because we want our model to be generalized and a model do very well in the training data could be overfitting

3. *Bayesian model selection* does not need *heldout data*. Outline how *Bayesian model selection* works, and why *heldout data* is not needed. [1 mark]

Because the Bayesian model intrinsically include noise distribution.

Question 5: Gradient Descent [3 marks]

1. Write down the *gradient descent* algorithm for minimising *training error* (also known as *total loss*). You can leave the function in a generic form $L(\theta)$. Explain your notation and any parameters you introduce. [2 marks]

Choose $\theta^{(1)}$ and some T

for $i = 1$ to T:

$\theta^{(i+1)} = \theta^{(i)} - \eta \nabla L(\theta^{(i)})$

return $\hat{\theta} \leftarrow \theta^{(i)}$

η: learning rate

$\nabla L(\theta)$: differentiate training Loss.

θ: unknown parameter

T: number of iterations.

2. Explain the difference between *gradient descent* and *stochastic gradient descent* algorithms. [1 mark]

For stochastic gradient descent, we first shuffle all the training example in B batches, then use gradient discent to update the parameter using from one batch at a time.

Question 6: Semi-Supervised Learning [2 marks]

Outline the steps involved in *self training*. [2 marks]

Not cover.

Section C: Numeric Questions [16 marks]

In this section you are asked to demonstrate your understanding of a subset of the methods that we have studied in this subject, in being able to perform calculations.

Question 7: Artificial Neural Networks [2 marks]

Consider a 2-class classification problem and *perceptron* training algorithm. Assume you have a 3-feature prediction function

$$f(x) = w_1 x_1 + w_2 x_2 + w_3 x_3$$

and current weights $\mathbf{w} = [2, 3, 4]'$. For simplicity assume that there is no bias term. Given an *example* $\mathbf{x} = [-2, 3, 1]'$ and *label* $y = -1$, what are the updated *weights*? Introduce additional *parameter(s)* if required, and specify which values were used for these parameter(s).

$s = w' \cdot x = -4 + 9 + 4 = 9$

$\because y = -1, \ s > 0$

$\therefore w^{(i)} = w^{(i)} - \eta \cdot x = [2 - \eta(-2), 3 - \eta(3), 4 - \eta(1)]$

assume that $\eta = 0.1$

$w^{(i)} = [2.2, 2.7, 3.9]$

Question 8: Properties of Kernels [3 marks]

1. What is the *kernel trick* and what is the benefit of using it? [1.5 marks]

2. Suppose $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^m$, $k : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ is a kernel, $c \in \mathbb{R}$, and $c > 0$. Prove that $ck(\boldsymbol{u}, \boldsymbol{v})$ is a kernel. [1.5 marks]

1. Fast computation of feature space dot product. For some $\phi(x)'$, it is much faster to compute the kernel directly than first mapping to feature space than take dot product. Benefit is reducing computational cost.

2. proof: $k' = c\,k(u, v), \quad c \in R, c > 0$

$$k' = \begin{bmatrix} c\,k(u_1, v_1) & \cdots & c\,k(u_1, v_m) \\ \vdots & \ddots & \\ c\,k(u_1, v_m) & \cdots & c\,k(u_m, v_m) \end{bmatrix}$$

$$A^T K' A = \sum_{i,j}^{m} c A_i \cdot A_j \cdot k(u_i, v_j)$$

$$= c \sum_{i,j}^{m} A_i \cdot A_j \cdot k(u_i, v_j) \geq 0$$

Question 9: Probabilistic Inference [3 marks]

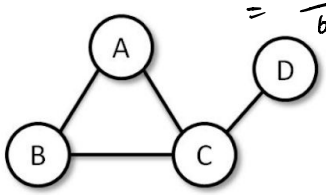
$\frac{1}{Z}\sum_B f(\bar{A},B,\bar{C})\cdot\sum_D g(\bar{C},D)$

$= \frac{1}{Z} \quad 1 \quad \cdot 4$

$= \frac{4}{68} = \frac{1}{17}$

Consider the following *undirected PGM*

A	B	C	f(A,B,C)
T	T	T	5
T	T	F	3
T	F	T	3
T	F	F	1
F	T	T	3
F	T	F	1
F	F	T	1
F	F	F	0



C	D	g(C,D)
T	T	0
T	F	4
F	T	4
F	F	0

where the truth tables associated with cliques are the model's *potentials* $f(A,B,C)$ and $g(C,D)$ respectively, and as such are not normalised conditional probability tables (as you get for *directed PGMs*).

1. For arbitrary truth values a,b,c,d, write an expression for $\Pr(A=a,B=b,C=c,D=d)$ in terms of $f(a,b,c), g(c,d)$, and normalising constant $Z=\sum_{A,B,C,D} f(A,B,C)g(C,D)$ only (just using these three expressions, without using the truth tables). [1 mark]

$\Pr(A=a,B=b,C=c,D=d) = \frac{1}{Z} f(a,b,c)\cdot g(c,d)$

2. Now using the tables, calculate the *normalising constant* $Z=\sum_{A,B,C,D} f(A,B,C)g(C,D)$. [1 mark]

$Z = 4 \times 17 = 68$

3. Calculate $\Pr(A=F,C=F)$. You may leave your answer as a fraction. (If you were unable to compute the normalising constant in the previous part, leave it as Z in your workings here.) [1 mark]

$\Pr(A=F,C=F) = \sum_{B,D} \Pr(A=F,B,C=F,D) = \frac{1}{Z}\cdot\sum_{B,D} f(A,B,C)\cdot g(C,D)$

$= \frac{1}{17}$

Question 10: Bayesian Posterior Updating [4 marks]

Consider a random variable over $X \in 0,1,2,\ldots$ governed by a $Poisson(\lambda)$ distribution. That is, X has probability mass function $p(x|\lambda)=\frac{\lambda^x \exp(-\lambda)}{x!}$. We set as the prior distribution the $Gamma(\alpha,\beta)$ distribution with probability density function $\frac{\lambda^{\alpha-1}\exp(-\lambda\beta)}{Z(\alpha,\beta)}$ where $Z(\alpha,\beta)$ is a normalising constant and $\alpha,\beta>0$ are constants. The *mode* of the $Gamma(\alpha,\beta)$ is $\frac{\alpha-1}{\beta}$ for $\alpha\geq 1$, and 0 otherwise.

1. Prove that this is a conjugate prior and likelihood, i.e., the posterior $p(\lambda|x,\alpha,\beta)$ has a *Gamma* distribution. [2 marks]

$p(x|\lambda)\cdot p(\lambda) = \frac{\lambda^x \exp(-\lambda)}{x!}\cdot\frac{\lambda^{\alpha-1}\exp(-\lambda\beta)}{Z(\alpha,\beta)} = \frac{\lambda^{x+\alpha-1}\exp(-\lambda-\lambda\beta)}{x!\cdot Z(\alpha,\beta)} \propto \lambda^{(x+\alpha)-1}\exp(-\lambda(1+\beta))$

prior × likelihood ∼ prior

$\propto Gamma(x+\alpha, 1+\beta)$

hence ...

2. After a single observation $X=x$, what is the resulting posterior $p(\lambda|x,\alpha,\beta)$? I.e., what are the new parameters, α',β', of the posterior $Gamma(\alpha',\beta')$ in terms of α,β,x? [1 mark]

Don't know.

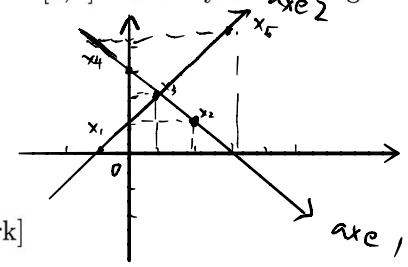
3. Explain whether the posterior distribution is more or less informative than the *posterior mode*; consider in your answer the effect of the number of observed data points. [1 mark]

?

posterior distribution is more informative because uncertainty about weight can be encoded in the posterior distribution.

Question 11: Principal Component Analysis [4 marks]

In this question, you will be performing *Principal Component Analysis (PCA)* over the following 2D dataset: $\mathbf{x}_1 = [-1, 0]'$, $\mathbf{x}_2 = [2, 1]'$, $\mathbf{x}_3 = [1, 2]'$, $\mathbf{x}_4 = [0, 3]'$, $\mathbf{x}_5 = [3, 4]'$. Show your working when answering parts 2 to 4.



1. Plot the dataset and draw the two PCA axes. [1 mark]

2. Compute the *variance* along the first PCA dimension. [1 mark]

$\frac{8+8}{5-1} = 4$

3. Compute the *variance* along the second PCA dimension. [1 mark]

$\frac{4}{4} = 1$

4. Compute the *covariance* between the two dimensions in the transformed space. [1 mark]

$\begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$

Hints: Recall that sample variance/covariance are normalised by $\frac{1}{n-1}$ where n is the number of points. You should be able to answer this question without using the notion of eigenvalues/eigenvectors. However, if you wish to use eigenvectors note that before PCA eigenvalues of the covariance matrix of the centered data are 1 and 4.

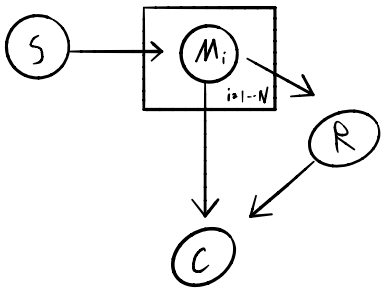
Section D: Design and Application Questions [10 marks]

In this section you are asked to demonstrate that you have gained a high-level understanding of the methods and algorithms covered in this subject, and can apply that understanding. Expect your answer to each question to be from one third of a page to one full page in length. These questions may require significantly more thought than those in Sections A–C and should be attempted only after having completed the earlier sections.

Question 12: Probabilistic Graphical Models [5 marks]

Your task is to design a probabilistic graphical model (PGM) to capture a real-world scenario. Consider the problem of a student graduating from University and looking for a graduate job, by writing a CV and having a lecturer write them a reference letter. The CV is based on the student's marks for the N subjects they have taken, and each mark is a reflection of the student's general skill. The reference letter is written by a lazy lecturer, basing their assessment purely on the mark obtained by the student in their subject.

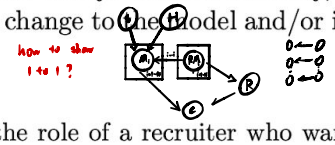
1. Design a PGM to capture the above scenario, using as random variables: M_i: mark on subject i, with $i \in [1, N]$; C: quality of their CV; S: skill level; R: recommendation of reference letter. [1 mark]



2. Assuming each variable is binary (i.e., each value is *high* or *low*; or *true* or *false*), state the number of conditional probability tables and the number of free parameters in each table needed to specify the model. [1 mark]

Node S : 1 CPT, 1 free parameter each CPT
Node M : N CPT, 2 free parameter each CPT
Node C : 1 CPT, 2^{M+1} free parameter each CPT
Node R : 1 CPT, 2^N free parameter

3. CVs are not always written honestly, in that the marks may not always be reported correctly. Propose a change to the model and/or its parameterisation to best capture this situation. [1 mark]



4. Consider the role of a recruiter who wants to employ honest and skilful workers. Their task is to find the skill of the student and whether or not they are honest. What method might they use to determine these values, and what inputs would be required? [1 mark]

Use probabilistic inference and use Elimination algorithm to determine Pr(Honesty = True) and Pr(Skill = True), require input: CV quality, Reference, Mark, and Real mark.

5. Finally, how might the conditional probability tables in the model be learned from data? What kind of data might be required, and what technique might be used to fit the parameters? [1 mark]

For a company, the latent data of Real mark may not be observed, they can use EM algorithm to fit the parameter.

Question 13: Multi-target tracking [5 marks]

Consider a *surveillance camera* overlooking a central area in a busy shopping center, and producing a video stream in real time. Hundreds of people are crossing this area throughout the day. Automated *detection and tracking software* is capable of identifying locations of pedestrians in each video frame, as well as of following customers' *trajectories*. Each person is being followed by the software from the moment this customer appears in the *field of view* of the *camera* until disappearance. In each *video frame* t, location for each identified person is represented as a 2D Euclidean point $(x_i(t), y_i(t))$, where i is the unique index for each person. For each person i, a *trajectory* is a sequence $\{(x_i(t_{0,i}), y_i(t_{0,i})), \ldots, (x_i(t_{n,i}), y_i(t_{n,i}))\}$, where $t_{0,i}$ and $t_{n,i}$ denote, respectively, frames of appearance and disappearance for this person. Customers that re-enter the field of view later are treated as new individuals.
One of the major questions that can be addressed with these data is identification of common routes. For example, it may turn out that the majority of customers are passing from south-east to north-west section of the shopping center, while a relatively small group heads towards west instead of north-west. Therefore, your task is to identify groups of similar trajectories.

Answer the following questions and justify your answers. Some of these questions will not necessarily have a single correct answer.

1. What is the main challenge for using *Gaussian Mixture models* to *cluster* the *trajectories*? [1 mark]

2. Can *Euclidean distance* be used for comparing *trajectories*? [1 mark]

3. Design a *function* that measures *similarity* between *trajectories*. [1 mark]

5. What other information apart from locations $(x_i(t), y_i(t))$ could be used to provide a deeper understanding of movement patterns? [1 mark]