EGT2
ENGINEERING TRIPOS PART IIA

Monday 11 May 2026 14.00 to 15.40

Module 3F8

INFERENCE

*Answer not more than **three** questions.*

All questions carry the same number of marks.

*The **approximate** percentage of marks allocated to each part of a question is indicated in the right margin.*

*Write your candidate number **not** your name on the cover sheet.*

STATIONERY REQUIREMENTS
Single-sided script paper

SPECIAL REQUIREMENTS TO BE SUPPLIED FOR THIS EXAM
CUED approved calculator allowed.
Engineering Data Book.

10 minutes reading time is allowed for this paper at the start of the exam.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed to do so.

You may not remove any stationery from the Examination Room.

1 A scientist uses two sensors to measure the location y of an object. The audio sensor output is equal to the object's location plus audio sensor noise, $a = y + \eta_a$ where the noise is zero-mean $\mathbb{E}[\eta_a] = 0$ and has variance $\mathbb{E}[\eta_a^2] = \sigma_a^2$. The visual sensor output is equal to the object's location plus visual sensor noise $v = y + \eta_v$ where the noise is zero-mean $\mathbb{E}[\eta_v] = 0$ and has variance $\mathbb{E}[\eta_v^2] = \sigma_v^2$. The audio noise and visual noise are independent from one another. The scientist wants to use these two pieces of information, a and v, to estimate the location of the object. They propose using an estimator $\hat{y} = \alpha a + \beta v$ where α and β are scalar valued weights.

(a) (i) By considering the expected value of the estimator $\hat{y}$, propose a sensible setting for β in terms of α . [15%]

(ii) By considering the second moment of the estimator $\hat{y}$, find the value of α which results in the lowest variance estimate for y. [20%]

(iii) What happens to your solution as $\sigma_a^2 \to \infty$ and as $\sigma_a^2 \to 0$? Explain this behaviour. [15%]

(i) $\mathbb{E}[\hat{y}] = \alpha \mathbb{E}[a] + \beta \mathbb{E}[v] = \alpha y + \beta y \Rightarrow \beta = 1 - \alpha$

(ii) $\mathbb{E}\left[(\hat{y} - y)^2\right] = \mathbb{E}\left[\left(\alpha(a - y) + (1-\alpha)(v - y)\right)^2\right]$

$= \alpha^2 \sigma_a^2 + (1-\alpha)^2 \sigma_v^2 = \text{var}(\hat{y})$

$\frac{d\,\text{var}(\hat{y})}{d\alpha} = 2\alpha\sigma_a^2 - 2(1-\alpha)\sigma_v^2 = 0$

$\Rightarrow \alpha\sigma_a^2 - \sigma_v^2 + \alpha\sigma_v^2 = 0 \quad \therefore \alpha = \frac{\sigma_v^2}{\sigma_a^2 + \sigma_v^2}$

$\hat{y} = \frac{\sigma_v^2}{\sigma_a^2 + \sigma_v^2} a + \frac{\sigma_a^2}{\sigma_a^2 + \sigma_v^2} v$

(iii) $\sigma_a^2 \to \infty \quad \hat{y} \to v$ (ignore audio) $\quad \sigma_a^2 \to 0 \quad \hat{y} \to a$ (only need audio)

(b) (i) Explain how *maximum likelihood estimation* can be used to infer an unknown quantity y from known observations x. [10%]

(ii) Relate the estimator derived in part (a) (ii) to maximum likelihood estimation in a probabilistic model. Explain any assumptions that you needed to make. [25%]

(iii) Explain how the *Bayesian* approach to inferring y differs from maximum likelihood estimation? Are there situations where Bayesian inference for y relates to the maximum-likelihood estimate? (You do not need to derive a Bayesian estimate to answer this question.) [15%]

(i) $\hat{y}_{MLE} = \arg\max_y \log p(x|y)$ i.e. select the value for the unknown which makes the observations maximally probable

(ii) Assume errors in sensors are independent

$$p([n_a, n_v]) = N(n_a; 0, \sigma_a^2)\, N(n_v; 0, \sigma_v^2)$$

$$\Rightarrow \log p(a, v|y) = \log N(a; y, \sigma_a^2) + \log N(v; y, \sigma_v^2) \quad \text{const. does not depend on } y$$

$$= -\frac{1}{2\sigma_a^2}(a-y)^2 - \frac{1}{2\sigma_v^2}(v-y)^2 + c$$

$$\left.\frac{d \log p(a,v|y)}{dy}\right|_{\hat{y}_{MLE}} = \frac{1}{\sigma_a^2}(a-\hat{y}_{MLE}) + \frac{1}{\sigma_v^2}(v-\hat{y}_{MLE}) = 0$$

$$\Rightarrow \hat{y}_{MLE} = \frac{\sigma_v^2}{\sigma_a^2+\sigma_v^2}a + \frac{\sigma_a^2}{\sigma_a^2+\sigma_v^2}v \Rightarrow \text{same as Q1b}$$

ie. MLE = min variance unbiased estimator for y

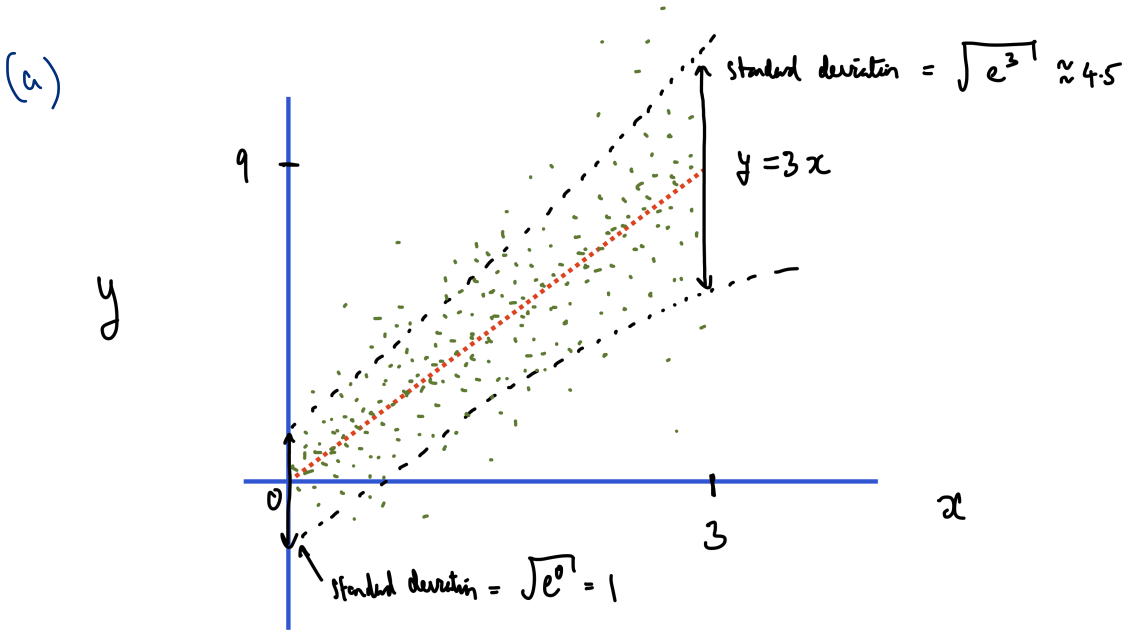
need a prior on y

(iii) Bayesian inference computes posterior $p(y|x) = p(x|y)\,p(y)/p(x)$

MAP finds maximum of posterior $\hat{y}_{MAP} = \arg\max_y p(y|x) = \hat{y}_{MLE}$

2 A *regression problem* comprises scalar inputs x_n and scalar outputs y_n which are linearly related $y_n = mx_n + \epsilon_n$. The inputs x_n take non-negative values. The observation noise ϵ_n is Gaussian, with mean 0, but it has a variance that depends on the input $p(\epsilon_n) = \mathcal{N}(\epsilon_n; 0, \sigma^2(x_n))$ where $\sigma^2(x_n) = \exp(x_n)$. A Gaussian prior with mean μ_m and variance σ_m^2 is placed on the slope parameter so $p(m) = \mathcal{N}(m; \mu_m, \sigma_m^2)$.

(a) Roughly sketch what a typical sample from this model might look like when $\mu_m = 3$, $\sigma_m^2 = 0.01$ and the inputs are drawn i.i.d. from a uniform distribution between (0,+3). A sample includes a single draw of m and a set of input-output pairs. Label your sketch. [20%]



 (cont.

(b) (i) Compute the expected value of the output given an input $\mathbb{E}_{p(y_n|x_n,\mu_m,\sigma_m^2)}[y_n]$ by marginalising over m. [10%]

(ii) Similarly, compute the expected value of the product of two outputs, y_n and $y_{n'}$, given the corresponding inputs $\mathbb{E}_{p(y_n,y_{n'}|x_n,x_{n'},\mu_m,\sigma_m^2)}[y_n\ y_{n'}]$ by marginalising over m. [20%]

(iii) Explain how the results in parts (b)(i) and (b)(ii) lead to an alternative way of viewing the regression model. [10%]

(i) $\mathbb{E}_{p(y_n|x_n,\mu_m,\sigma_m^2)}[y_n] = \mathbb{E}_{p(m,\varepsilon_n)}[mx_n+\varepsilon_n]$

$= \mu_m x_n$

(ii) $\mathbb{E}_{p(y_n,y_{n'}|x_n,x_{n'},\mu_m,\sigma_m^2)}[y_n y_{n'}] = \mathbb{E}_{p(m,\varepsilon_n,\varepsilon_{n'})}[(mx_n+\varepsilon_n)(mx_{n'}+\varepsilon_{n'})]$

$= x_n x_{n'}\mathbb{E}_m(m^2) + x_n \mathbb{E}_m(m)\cancel{\mathbb{E}_\varepsilon(\varepsilon_{n'})}^{0} + x_{n'}\cancel{\mathbb{E}_m(m)}^{0}\mathbb{E}_{\varepsilon_n}(\varepsilon_n) + \mathbb{E}_{\varepsilon_n,\varepsilon_{n'}}(\varepsilon_n\varepsilon_{n'})$

$= x_n x_{n'}(\mu_m^2+\sigma_m^2) + \delta_{n,n'}e^{x_n}$

(iii) Alternative view of model:

$\underline{y}\,|\,\underline{x}, \theta \sim N\left(\mu_m \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}, (\sigma_m^2+\mu_m^2)\begin{bmatrix} x_1^2 & x_1x_2 & \cdots \\ x_1x_2 & x_2^2 & \ddots \\ \vdots & & x_N^2 \end{bmatrix} + \begin{bmatrix} e^{x_1} & 0 & \cdots \\ 0 & e^{x_2} & \ddots \\ \vdots & \ddots & e^{x_N} \end{bmatrix}\right)$

($\underline{y}$: $\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$; $\underline{x}$: $\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$; μ_m & σ_m^2)

i.e. a big Gaussian over the ys given the xs & parameters

(c) (i) The slope m must now be learned from a training dataset $\{x_n, y_n\}_{n=1}^N$ in a Bayesian way. Compute the posterior distribution over m after seeing N data points, that is $p(m|\{x_n, y_n\}_{n=1}^N, \mu_m, \sigma_m^2)$. [20%]

(ii) You are allowed to select an input location x at which you will be provided with an output y. Which location is most informative about the parameter m? Explain your reasoning. [20%]

(i) $p(m|\{x_n,y_n\}_{n=1}^N,\mu_m,\sigma_m^2) \propto p(m|\mu_m,\sigma_m^2)\prod_{n=1} p(y_n|x_n,m)$

$\propto e^{-\frac{1}{2\sigma_m^2}(m-\mu_m)^2 - \frac{1}{2}\sum_n \frac{(y_n - mx_n)^2}{\sigma_n^2}}$ where $\sigma_n^2 = e^{x_n}$

$\propto e^{-\frac{1}{2\sigma_m^2}m^2 + \frac{\mu_m}{\sigma_m^2}\cdot m - \frac{1}{2}m^2\sum_n x_n^2/e^{x_n} + m\cdot\sum_n x_n y_n/e^{x_n}}$

$p(m|\{x_n,y_n\}_{n=1}^N,\mu_m,\sigma_m^2) = N(m;\mu_{m|D},\sigma_{m|D}^2)$

$= e^{-\frac{1}{2}m^2/\sigma_{m|D}^2 + \mu_{m|D}/\sigma_{m|D}^2 m}$

$\therefore \quad \sigma_{m|D}^2 = \frac{1}{1/\sigma_m^2 + \sum_n x_n^2/e^{x_n}} \qquad \mu_{m|D} = \sigma_{m|D}^2\left(\mu_m/\sigma_m^2 + \sum_n x_n y_n/e^{x_n}\right)$

$\sigma_{m|D}^2 = \frac{\sigma_m^2}{1+\sigma_m^2\sum_n x_n e^{-x_n}} \qquad \mu_{m|D} = \sigma_{m|D}^2\left(\mu_m/\sigma_m^2 + \sum_n x_n y_n e^{-x_n}\right)$

(ii) Select point which minimises posterior variance (maximally certain)

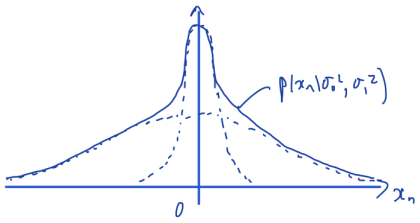
$\underset{x}{\arg\min}\ \sigma_{m|D}^2 = \underset{x}{\arg\max}\left(\frac{1}{\sigma_m^2} + x^2e^{-x}\right) = \underset{x}{\arg\max}\left(x^2e^{-x}\right)$

$\frac{d}{dx}(x^2e^{-x}) = 2xe^{-x} - x^2e^{-x} = 0 \Rightarrow \boxed{x=2}$ is the min of the posterior variance

3 A machine learner uses a *latent variable model* for a dataset comprising N real-valued scalar variables $\{x_n\}_{n=1}^N$. The model comprises binary latent variables $s_n \in \{0,1\}$. The prior distribution over the latent variable is uniform $p(s_n = 1) = \frac{1}{2}$. The observed data are generated from a zero-mean Gaussian distribution whose variance depends on the latent variable $p(x_n|s_n,\sigma_0^2,\sigma_1^2) = \mathcal{N}(x_n;0,(1-s_n)\sigma_0^2 + s_n\sigma_1^2)$. The machine learner would like to use the *EM algorithm* to fit the model to the dataset.

(a) Sketch the marginal distribution over the observed data $p(x_n|\sigma_0^2,\sigma_1^2)$. [10%]

$p(x_n|\sigma_0^2,\sigma_1^2) = \sum_{s_n=0}^{1} p(x_n|s_n,\sigma_0^2,\sigma_1^2)\,p(s_n=1) = \frac{1}{2}\mathcal{N}(x_n;0,\sigma_0^2) + \frac{1}{2}\mathcal{N}(x_n;0,\sigma_1^2)$



(b) Define the *E-step* of the EM algorithm. [10%]

(c) Calculate the *E-step* update for the model above, leaving your answer in a form which is suitable for implementation. [30%]

(d) What happens to your solution when $\sigma_0^2 = \sigma_1^2$? Explain this behaviour. [10%]

(b) E-Step $q_n^{(new)}(s_n) = \underset{q_n(s_n)}{\operatorname{argmax}}\; \mathcal{Q}\left(\theta, \{q_n(s_n)\}_{n=1}^N\right)$

(c) $\Rightarrow$ $q_n(s_n=1) = p(s_n=1|x_n,\sigma_0^2,\sigma_1^2) = \dfrac{p(s_n=1)\,p(x_n|s_n=1,\sigma_1^2)}{p(s_n=1)\,p(x_n|s_n=1,\sigma_1^2) + p(s_n=0)\,p(x_n|s_n=0,\sigma_0^2)}$

$= \dfrac{\mathcal{N}(x_n;0,\sigma_1^2)}{\mathcal{N}(x_n;0,\sigma_1^2) + \mathcal{N}(x_n;0,\sigma_0^2)}$

$= \dfrac{1}{1 + e^{-\frac{1}{2}\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right)x_n^2 - \frac{1}{2}\log \sigma_1^2/\sigma_0^2}}$

logistic regression with basis functions $(1, x_n^2)$

weights $\left(\frac{1}{2}\log \sigma_1^2/\sigma_0^2,\ \frac{1}{2}\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right)\right)$

(d) $\sigma_0^2 = \sigma_1^2 \Rightarrow q_n(s_n=1) = 1/2 \quad \forall n$

(e) Define the *M-step* of the EM algorithm. [10%]

(f) Calculate the *M-step* update for the model leaving your answer in a form which is suitable for implementation. [30%]

(e) M-step $\theta^{(new)} = \underset{\theta}{\operatorname{argmax}} \; q\left(\theta, \{q_n(s_n)\}_{n=1}^{N}\right)$

(f)

$$\{\sigma_{0,new}^2, \sigma_{1,new}^2\} = \underset{\sigma_0^2, \sigma_1^2}{\operatorname{argmax}} \; q\left(\sigma_0^2, \sigma_1^2, \{q_n(s_n)\}_{n=1}^{N}\right)$$

$$= \underset{\sigma_0^2, \sigma_1^2}{\operatorname{argmax}} \sum_{n=1}^{N} \sum_{s_n=0}^{1} q_n(s_n) \log p(x_n \mid 0, \sigma_{s_n}^2)$$

$$\frac{dq}{d\sigma_1^2} = \frac{d}{d\sigma_1^2} \sum_{n=1}^{N} q_n(s_n=1) \left[-\frac{1}{2}\log 2\pi\sigma_1^2 - \frac{1}{2\sigma_1^2} x_n^2\right]\Bigg|_{\sigma_1^2 = \sigma_{1,new}^2} = 0$$

$$= \sum_{n=1}^{N} q_n(s_n=1) \left[-\frac{1}{2}\frac{1}{\sigma_{1,new}^2} + \frac{1}{2}\frac{1}{\sigma_{1,new}^4} x_n^2\right]$$

$$\Rightarrow \sigma_{1,new}^2 = \frac{\sum_{n=1}^{N} q_n(s_n=1) x_n^2}{\sum_{n=1}^{N} q_n(s_n=1)}$$

By symmetry

$$\sigma_{0,new}^2 = \sum_{n=1}^{N} q_n(s_n=0) x_n^2 \Big/ \sum_{n=1}^{N} q_n(s_n=0)$$

4 (a) (i) Define a *Hidden Markov Model* (HMM) for real-valued observed data $\{y_t\}_{t=1}^T$ and discrete latent variables $\{x_t\}_{t=1}^T$. Your solution should define the *initial state probabilities*, *transition probabilities*, and the *emission distribution* of the HMM. [15%]

(ii) A machine learner has developed a non-standard HMM comprising two hidden state variables at each time step, $x_t^{(1)}$ and $x_t^{(2)}$. Each of these variables can take one of two states and they are generated by the following *bigram* models,

$$p(x_1^{(1)} = 1) = 1/2, \;\; p(x_t^{(1)} = 1 | x_{t-1}^{(1)} = 1) = p(x_t^{(1)} = 2 | x_{t-1}^{(1)} = 2) = 0.9$$
$$p(x_1^{(2)} = 1) = 3/4, \;\; p(x_t^{(2)} = 1 | x_{t-1}^{(2)} = 1) = p(x_t^{(2)} = 2 | x_{t-1}^{(2)} = 2) = 0.8$$

The observed state variable y_t is a continuous scalar and is modelled using a Gaussian distribution whose mean and variance depend on the two hidden states

$$p(y_t | x_t^{(1)}, x_t^{(2)}) = \mathcal{N}\left(y_t; \mu(x_t^{(1)}, x_t^{(2)}), \sigma^2(x_t^{(1)}, x_t^{(2)})\right).$$

In this way the mean and the variances are stored in 2×2 matrices and if, say, $x_t^{(1)} = 2$ and $x_t^{(2)} = 1$ then the mean of the Gaussian is $\mu(2,1)$ and the variance is $\sigma^2(2,1)$.

Write the model above in terms of an equivalent standard HMM. Explain your reasoning. [35%]

a(i) book work

(ii) merge the two binary chains into a single 4-state chain

$$\begin{array}{lcccccc} (x_t^{(1)}, x_t^{(2)}) & t & (1,1) & (1,2) & (2,1) & (2,2) \\ & & \downarrow & \downarrow & \downarrow & \downarrow \\ z_t & t & 1, & 2, & 3, & 4 \end{array}$$

initial state distribution

$$\underline{\pi} = \left(\frac{1}{2}\cdot\frac{3}{4}, \frac{1}{2}\cdot\frac{1}{4}, \frac{1}{2}\cdot\frac{3}{4}, \frac{1}{2}\cdot\frac{1}{4} \right) = \left(\frac{3}{8}, \frac{1}{8}, \frac{3}{8}, \frac{1}{8} \right)$$

transition matrix

$$\underline{\underline{T}}^{(1)} = \begin{pmatrix} 0.9 & 0.1 \\ 0.1 & 0.9 \end{pmatrix} \qquad \underline{\underline{T}}^{(2)} = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}$$

$$\underline{\underline{T}} = \begin{bmatrix} 0.9 \times \underline{\underline{T}}^{(2)} & 0.1 \times \underline{\underline{T}}^{(2)} \\ 0.1 \times \underline{\underline{T}}^{(2)} & 0.9 \times \underline{\underline{T}}^{(2)} \end{bmatrix}$$

Aside: (known as a kronecker product)

$$= \begin{bmatrix} 0.72 & 0.18 & 0.08 & 0.02 \\ 0.18 & 0.72 & 0.02 & 0.08 \\ 0.08 & 0.02 & 0.72 & 0.18 \\ 0.02 & 0.08 & 0.18 & 0.72 \end{bmatrix}$$

Emission prob: $p(y_t | z_t = (i,j)) = N(y_t; \mu_{(i,j)}, \sigma^2_{(i,j)})$ — shorthand for this mapping

(b) Consider a joint distribution over seven variables $x_1, x_2, \ldots, x_7$ each of which is discrete and can take one of D different values.

(i) Write an expression for computing the marginal distribution $p(x_3)$. How costly is it to compute $p(x_3)$ in the general case? [15%]

(ii) Now consider a joint distribution which factorises according to

$$p(x_{1:7}) = p(x_1)p(x_2|x_1)p(x_3|x_1,x_4,x_6)p(x_4|x_7)p(x_5|x_3)p(x_6|x_7)p(x_7).$$

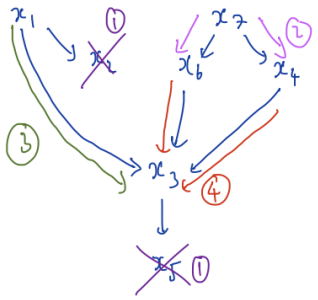
Write down an efficient way of computing the marginal $p(x_3)$ in this case. [25%]

(iii) How costly is it to compute $p(x_3)$ using your solution in part (b)? How does it relate to *dynamic programming*? [10%]

(i) $p(x_3) = \sum_{x_1,x_2,x_4,x_5,x_6,x_7} p(x_1,x_2,x_3,x_4,x_5,x_6,x_7)$

cost = D^7 operations

(ii)



message passing interpretation / dynamic programming

$$p(x_3) = \sum_{x_1,x_2,x_4,x_5,x_6,x_7} p(x_1)\,p(x_2|x_1)\,p(x_3|x_1,x_4,x_6)\,p(x_5|x_3)\,p(x_6|x_7)\,p(x_4|x_7)\,p(x_7)$$

$$= \sum_{x_1,x_4,x_6,x_7} p(x_1)\,p(x_3|x_1,x_4,x_6)\,p(x_6|x_7)\,p(x_4|x_7)\,p(x_7) \left(\sum_{x_2} p(x_2|x_1)\right)\left(\sum_{x_5} p(x_5|x_3)\right)$$

$$= \sum_{x_4,x_6} \left(\sum_{x_1} p(x_1)p(x_3|x_1,x_4,x_6)\right) \times \left(\sum_{x_7} p(x_6|x_7)\,p(x_4|x_7)\,p(x_7)\right)$$

(iii) (2) requires $D \propto D^2$ operations $= D^3$

(3) requires $D \cdot D^3$ operations $= D^4$

(4) requires $D^2 \cdot D$ operations $= D^3$

$\Rightarrow$ cost is $O(D^4) << O(D^7)$

dynamic programming leads to efficient computations – has an interpretation as message passing

ASSESSOR'S REPORT
ENGINEERING TRIPOS PART IIA 2025/26
ASSESSOR'S REPORT
MODULE 3F8: INFERENCE

The examination was taken by 153 candidates in total. The raw marks (from those who had taken IB) had an average of 70% and standard deviation 11% with the top at 95% and bottom at 42%.

Q1 Fundamental Inference Concepts

127 attempts, Ave. raw mark 14.5/20, St.Dev. 2.9.

The first question tested knowledge about basic inference problems and concepts such as expectations, bias, variance, maximum likelihood, Bayesian and MAP estimation. Part (a) was very well answered. Second part (b)(ii) was not as well answered. Only a few candidates could derive the maximum likelihood estimate for y and show that it was equal to the minimum variance estimate derived in part (b)(ii).

Q2 Regression and multivariate Gaussian distributions

99 attempts, Ave. raw mark 13.3/20, St.Dev. 2.9.

This question tested understanding of Bayesian regression models and multivariate Gaussians. Many candidates found the derivation of the covariance of the data challenging — many people used integrals for this rather than rules around expectations which are much simpler to use. Many missed the diagonal contribution to the covariance arising from the variance of the noise. Very few candidates saw that an alternative way of viewing the model is as a Gaussian distribution over the outputs y given the inputs x with a mean and covariance as derived in b(i) and (b)(ii).

Q3 The EM Algorithm

126 attempts, Ave. raw mark 14.6/20, St.Dev. 2.8.

This question covered latent variable models and the expectation maximisation algorithm. Generally this question was very well done, especially in the context of previous questions on this topic where candidates have generally struggled. The sketch required in part (a) of a scale mixture of Gaussians was the weakest answer with many candidates sketching a distribution that was indistinguishable from a single Gaussian. Some candidates left part (c) in a form that is not ready for implementation.

Q4 Discrete Hidden Markov Models

44 attempts, Ave. raw mark 12.6/20, Stan. Dev. 3.3.

This question covered discrete Hidden Markov Models and dynamic programming. Candidates struggled with this question, especially part (b) where they did not correctly apply the sum and the product rule to efficiently compute the marginalisation. A large number of candidates applied Bayes' rule here which was not necessary and does not lead to computational efficiencies. Many others did not realise that the summations could be distributed in an efficient way after using the product rule to split up the joint distribution.

R. E. Turner (Principal Assessor)
20/5/2026