**Always provide justifications and show any intermediate work for your answers. A correct but unsupported answer may not receive any marks.**

You may find the following useful:

•Bernoulli distribution

$$p(x|\mu) = \mu^x(1-\mu)^{1-x}, \quad x \in \{0,1\} \tag{1}$$

•Binomial distribution

$$p(m|N,\mu) = \binom{N}{m}\mu^m(1-\mu)^{N-m} \tag{2}$$

•Beta distribution

$$\text{Beta}(\mu|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\mu^{\alpha-1}(1-\mu)^{\beta-1} \tag{3}$$

•Gaussian distribution

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = (2\pi)^{-\frac{D}{2}}|\boldsymbol{\Sigma}|^{-\frac{1}{2}}\exp\left(-\tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right), \quad \boldsymbol{x} \in \mathbb{R}^D \tag{4}$$

•Gamma distribution

$$\text{Gamma}(\tau|a,b) = \frac{1}{\Gamma(a)}b^a\tau^{a-1}\exp(-b\tau) \tag{5}$$

•Wishart distribution

$$\mathcal{W}(\boldsymbol{\Sigma}|\boldsymbol{W},\nu) = B|\boldsymbol{\Sigma}|^{\frac{\nu-D-1}{2}}\exp\left(-\tfrac{1}{2}\text{tr}(\boldsymbol{W}^{-1}\boldsymbol{\Sigma})\right), \quad \boldsymbol{\Sigma} \in \mathbb{R}^{D\times D} \tag{6}$$

The equations for propagating probabilities in Bayesian networks are:

The $\lambda$ message from child $C$ to parents $A$ and $B$ is given by:

$$\lambda_C(a_i) = \sum_{j=1}^{m} \pi_C(b_j) \sum_{k=1}^{n} P(c_k|a_i\&b_j)\lambda(c_k)$$

In the case where we have a single parent ($A$) this reduces to:

$$\lambda_C(a_i) = \sum_{k=1}^{n} P(c_k|a_i)\lambda(c_k)$$

and for the case of the single parent we can use the simpler matrix form:

$$\lambda_{\mathbf{C}}(\mathbf{A}) = \lambda(\mathbf{C})\mathbf{P}(\mathbf{C}|\mathbf{A})$$

The matrix form for multiple parents relates to the joint states of the parents.

$$\lambda_{\mathbf{C}}(\mathbf{A}\&\mathbf{B}) = \lambda(\mathbf{C})\mathbf{P}(\mathbf{C}|\mathbf{A}\&\mathbf{B})$$

It is necessary to separate the $\lambda$ evidence for the individual parents with a scalar equation of the form:

$$\lambda_C(a_i) = \Sigma_j \pi_C(b_j)\lambda_C(a_i\&b_j)$$

The $\pi$ evidence to child node $C$ from two parents $A$ and $B$ is given by:

$$\pi(c_k) = \sum_{i=1}^{l} \sum_{j=1}^{m} P(c_k|a_i\&b_j)\pi_C(a_i)\pi_C(b_j)$$

This can be written in matrix form as follows:

$$\pi(\mathbf{C}) = \mathbf{P}(\mathbf{C}|\mathbf{A}\&\mathbf{B})\pi_{\mathbf{C}}(\mathbf{A}\&\mathbf{B})$$

where

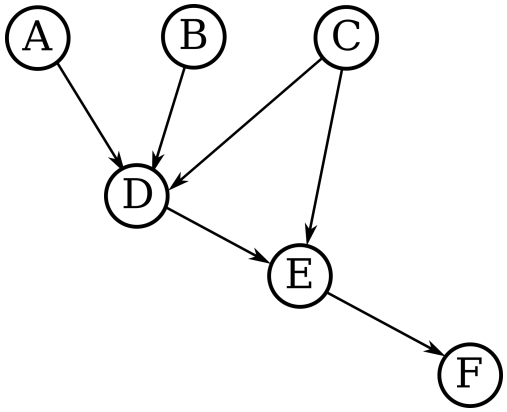
$$\pi_C(a_i\&b_j) = \pi_C(a_i)\pi_C(b_j)$$

The single parent matrix equation is:

$$\pi(\mathbf{C}) = \mathbf{P}(\mathbf{C}|\mathbf{A})\pi_{\mathbf{C}}(\mathbf{A})$$

1 **Probability Propagation**

In the Bayesian network below the variables are all binary, with, for example, variable $A$ having states $a_1$ and $a_2$.




The conditional probability matrices are given consistently in the form:

$$\begin{bmatrix} P(q_1|r_1\&s_1\&t_1) & P(q_1|r_1\&s_1\&t_2) & P(q_1|r_1\&s_2\&t_1) & P(q_1|r_1\&s_2\&t_2) & \cdots \\ P(q_2|r_1\&s_1\&t_1) & P(q_2|r_1\&s_1\&t_2) & P(q_2|r_1\&s_2\&t_1) & P(q_2|r_1\&s_2\&t_2) & \cdots \end{bmatrix}$$

and are

$$P(A) = [0.2, 0.8]$$

$$P(B) = [0.4, 0.6]$$

$$P(C) = [0.6, 0.4]$$

$$P(E|C\&D) = \begin{bmatrix} 0 & 0.3 & 0 & 0.8 \\ 1 & 0.7 & 1 & 0.2 \end{bmatrix}$$

$$P(F|E) = \begin{bmatrix} 0.4 & 0.5 \\ 0.6 & 0.5 \end{bmatrix}$$

$$P(D|A\&B\&C) = \begin{bmatrix} 1 & 0 & 0.3 & 1 & 1 & 0.6 & 0 & 0 \\ 0 & 1 & 0.7 & 0 & 0 & 0.4 & 1 & 1 \end{bmatrix}$$

a Given that node a is instantiated to state $a_1$, and there are no other instantiations calculate the $\pi$ evidence for node D.

b Node A remains instantiated to state $a_1$, and node C is instantiated to state $c_2$. Calculate the $\pi$ evidence at node and E.

c Node F is now instantiated to state $f_2$, and nodes A and C remain instantiated as before. Calculate the posterior probability of node D.

d When the network was constructed from data it was found that the joint probability table P(E&C&D) was:

| | $c_1d_1$ | $c_1d_2$ | $c_2d_1$ | $c_2d_2$ |
|---|---|---|---|---|
| $e_1$ | 0 | 0.06 | 0 | 0.32 |
| $e_2$ | 0.1 | 0.14 | 0.3 | 0.08 |

Assuming that C-D was the arc with the lowest dependency find the conditional probability matrices $P(E|C)$ and $P(E|D)$ for the spanning tree.

e Given that you wanted to remove the arc C-D as in part d, but did not have the joint probability table $P(A\&B\&C\&D)$ but only the conditional and prior probability tables given above, what assumption would you need to make in order to compute an estimate of the conditional probability matrix $P(D|A\&B)$? Calculate the estimate of $P(D|A\&B)$.

*The five parts carry equal marks.*

## 2 Exact and Approximate Inference

a Find a join tree that can be used for exact probability propagation between the variables of the network defined in question 1, using the following steps:

i) Draw the moral graph and find its cliques.

ii) Using the clique containing variable A as the root node, find an ordering of the cliques that preserves the running intersection property.

iii) For each clique find the R and S variable sets and the initial value of the potential function in terms of the conditional and prior probabilities in the original Bayesian network.

b Calculate the potential tables for each clique of the join tree before any propagation occurs.

c Given that variable F is instantiated to state $f_2$ calculate:

i) The lambda message that will be sent from its clique.

ii) The change in the potential table of its parent.

d Cutset conditioning is a possible alternative algorithm for exact computation in the case of the network defined in question 1. Which nodes of the original network could be used as a cut set. Give the reasons for your choice.

e An alternative approach to ensure accurate probability propagation would be to add a hidden (or latent) node to make the network singly connected. Redraw the network with an appropriate placed hidden node and briefly explain how its associated link matrices could be found.

*The five parts carry equal marks.*

## 3 Gaussian Processes

a Fill in the gaps in the following text.
*You will get 0.5 mark for a correct answer. For a wrong or incomplete answer, we will subtract 0.5 mark. If you choose to leave the field blank, you will get 0 marks for this answer. In total, you cannot get less than 0 marks or more than 6 marks.*

We consider a regression setting $y = f(\boldsymbol{x}) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$, $\boldsymbol{x} \in \mathbb{R}^D$, $y \in \mathbb{R}$. We place a Gaussian process (GP) prior on $f$. A Gaussian process (GP) is fully specified by a mean function $m(\,\cdot\,)$ and a covariance function $k(\,\cdot\,,\,\cdot\,)$. In the following, we assume that $m \equiv 0$.

A commonly used covariance function is the squared exponential (Gaussian) covariance function

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boxed{\text{(1)}}$$

which reflects the prior belief of a $\boxed{\text{(2)}}$ latent function. This means, the closer the two inputs $\boldsymbol{x}$ and $\boldsymbol{x}'$ are, the more correlated the corresponding function values $f(\boldsymbol{x}), f(\boldsymbol{x}')$. The $\boxed{\text{(3)}}$ hyper-parameter determines the degree of correlation between to function values $f(\boldsymbol{x})$ and $f(\boldsymbol{x}')$.

The training data is given by $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]^T$ of training inputs and the vector $\boldsymbol{y} = [y_1, \ldots, y_n]^T$ of corresponding training targets (observations).

We train the hyper-parameters of the GP by maximizing the $\boxed{\text{(4)}}$, given by

$$\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = \boxed{\text{(5)}}$$

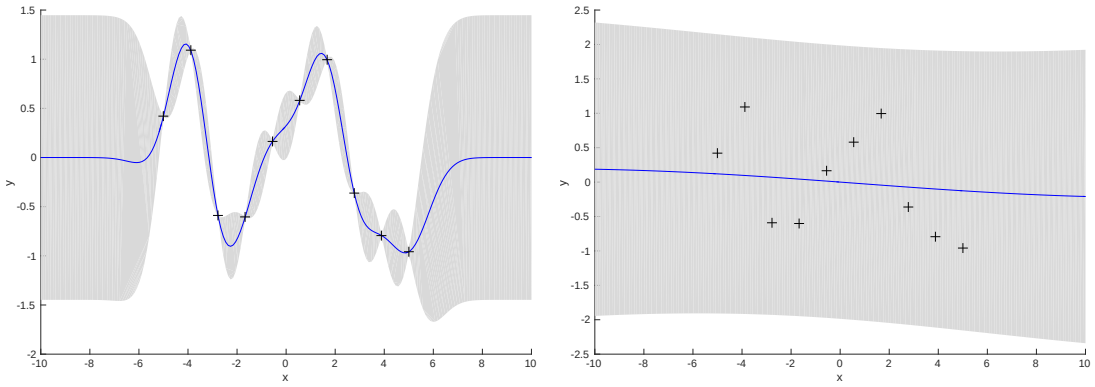
where $\boldsymbol{K}$ is the kernel matrix. This training objective automatically trades off $\boxed{\text{(6)}}$ and $\boxed{\text{(7)}}$.

The posterior predictive distribution of the function value $f_* = f(\boldsymbol{x}_*)$ at a query point $\boldsymbol{x}_*$ is $\boxed{\text{(8)}}$ with mean and variance given by

$$\mu(f_*) = \boxed{\text{(9)}}$$
$$\sigma^2(f_*) = \boxed{\text{(10)}}$$

Training a GP scales in $\mathcal{O}(\boxed{\text{(11)}})$; predictions required $\mathcal{O}(\boxed{\text{(12)}})$ computations.

b Figure 1 shows two possible fits of a GP (prior mean $m \equiv 0$, Gaussian covariance function) for a given data set (black crosses). The signal variance is




(a) GP fit 1. (b) GP Fit 2.

Fig. 1: Possible GP fits.

the same in both fits. However, the GPs differ in terms of length scale and noise variance.

i) Briefly explain the differences between the two GP fits in terms of length scales and noise variance (2 bullet points)

ii) Why can both GP fits be possible outcomes of the training procedure? (1–2 sentences)

iii) Why would it be useful to integrate out the hyper-parameters (1 bullet point) and how would you do this (1 bullet point)?

c One way to scale a GP to larger data sets is to use a distributed model.

i) Explain in 2–3 sentences the key idea behind distributed GPs with respect to training and predictions.

ii) One model that we discussed is the product-of-experts model. Give the equation for predicting the function value $f_*|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}_*$ at $\boldsymbol{x}_*$.

iii) In the product-of-experts-model, assume that every expert predicts $p(f_*|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{x}_*) = \mathcal{N}(0, 1)$

* What happens to the predictive variance in the limit of infinitely many experts

* Why does this happen?

* How could you address this issue?

*The three parts carry, respectively, 30%, 30%, and 40% of the marks.*

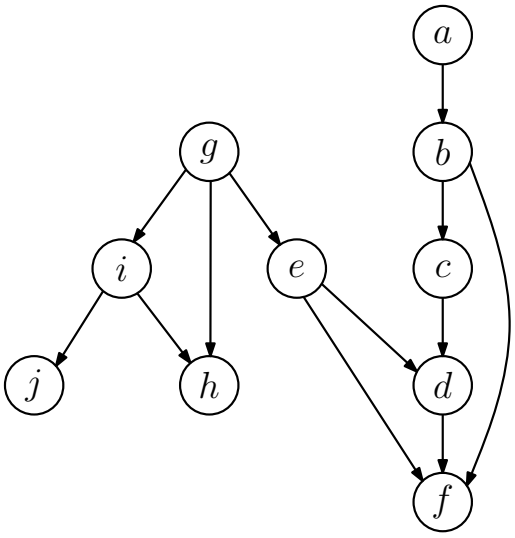



Fig. 2: Directed graphical model

## 4 Graphical Models, Bayesian Optimization, Sampling

### a Graphical Models

Consider the graphical model in Figure 2. State whether the following conditional independence statements are correct?
*You will get 1 mark for a correct answer. For a wrong answer, we will subtract 1 mark. You cannot get less than 0 marks.*

i) $a \perp\!\!\!\perp f$

ii) $a \perp\!\!\!\perp g$

iii) $b \perp\!\!\!\perp i|f$

iv) $d \perp\!\!\!\perp j|g, h$

v) $i \perp\!\!\!\perp b|h$

vi) $j \perp\!\!\!\perp d$

vii) $i \perp\!\!\!\perp c|h, f$

### b Bayesian Optimization

We want to globally minimize an unknown objective function $f$ with respect to its parameters $\boldsymbol{x}$. We only have access to noisy function evaluations $y = f(\boldsymbol{x}) + \epsilon, \epsilon \sim \mathcal{N}(0, \sigma_n^2)$.

i) Describe the key steps of Bayesian optimization given an initial set of experiments $\mathcal{D}^{(0)} = \{\boldsymbol{X}^{(0)}, \boldsymbol{y}^{(0)}\}$

ii) What is the role of the acquisition function in Bayesian optimization?

iii) Gaussian processes (without approximations) scale up to about 10,000 data points. Why is Bayesian optimization with 10,000 not practical from a computational perspective?
*Hint: Where does Bayesian optimization spend most time?*

iv) Briefly describe how the PI (probability of improvement) acquisition function works (when looking at sampled functions) and explain why exploration is very slow in its standard form. How could this be fixed? (3–4 sentences).
*Hint: What does the typical PI acquisition function look like?*

c **Sampling**

i) Imagine that a model consists of three random variables $A, B, C$. What three distributions would a standard Gibbs sampler sample from in order to generate samples from $p(A, B, C)$?

ii) In the context of rejection sampling, consider a true distribution $p(\boldsymbol{x}) = \mathcal{N}(\mathbf{0}, \sigma_p^2 \boldsymbol{I})$ and a proposal distribution $q(\boldsymbol{x}) = \mathcal{N}(\mathbf{0}, \sigma_q^2 \boldsymbol{I})$ where $\sigma_q = 1.01\sigma_p$.

A) What is the value of $k$ such that $kq \geq p$, if $\boldsymbol{x} \in \mathbb{R}^{1000}$?

B) What is the acceptance rate of rejection sampling with this proposal distribution?

*Hint: It may be helpful to draw a diagram.*

*The three parts carry, respectively, 35%, 45%, and 20% of the marks.*