COMS10011 sample paper

THIS IS STILL A WORK IN PROGRESS but is close to its final form! This is a sample paper, it has the same style of question as the real paper, the layout is slightly different in trivial ways to the official exam layout.

Rubric

This paper contains *two* parts.
The first section contains *15* short questions.
Each question is worth *two marks* and all should be attempted.
The second section contains *three* long questions.
Each long question is worth *20 marks*.
The best *two* long question answers will be used for assessment.
The maximum for this paper is *70 marks*.
Calculators must have the Faculty of Engineering Seal of Approval.

Section A: short questions - answer all questions

1. What is the definition of Shannon's entropy for a discrete distribution?

   Answer:
   For sample space $\mathcal{X}$ we have

   $$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x) \tag{1}$$

2. If you have a finite set of n spike trains and calculate the entropy of the spike trains using the discretization method, in general, what limit would the entropy reach as the time step is made very small?

   Answer:
   If the time step is small enough each spike train would correspond to a different word, with small enough time step, each spike in each train would, in general, fall into a different time bins. This means there would be n different words, each with one occurance and hence probability $1/n$. Thus the entropy would be $\log n$. I recently had to reject a paper where the authors had completely misunderstood this!

3. Given a Markov chain $V \to X \to H$ what can we say about $p_{V,H|X}(v, h|x)$?

   Answer:

   $$p_{V,H|X}(v, h|x) = p_{V|X}(v|x) p_{H|X}(h|x) \tag{2}$$

4. What is the cocktail party problem?

   Answer:
   Often the environment is very noisy with a sound, such as a voice, that is being attended to, not much distinguished in amplitude from other noises.

I have read that this situation holds at 'cocktail parties', a form of social entertainment popular in the last century somewhat akin to pre-drinks but without going anywhere afterwards. The cocktail party problem is the question of how the brain separates the sound signal it is attending to from the background noise.

5. In the Eriksen flanker task sketch the accuracy versus reaction time for the consistent, **HHH**, and inconsistent, **HSH**, conditions. The overall scale of the reaction time is not what is being asked for, rather the shape.

   Answer:
   In both cases the line starts at 0.5 but in the consistent case it rises quickly to one, in the inconsistent case it falls below 0.5 before rising more slowly to one. [1 mark for starting at 0.5 and rising to 1, the other for the dip in inconsistent].

6. The n-armed bandit task is used in psychological studies of decision making. What is an n-armed bandit?

   Answer:
   In an n-armed bandit the participant has to chose between n options, typically n buttons, each with a different probability of reward.

7. If four options in a decision task have estimated reward values r_1, r_2, r_3 and r_4, what is the soft-max probability for chosing the ith option?

   Answer:

   $$p_i = \frac{e^{\beta r_i}}{\sum_j e^{\beta r_j}} \tag{3}$$

   for some β, a parameter determining the exploration to exploitation balance.

8. Define the credit assignment problem in neuroscience.

   Answer: The credit assignment is the problem of deciding how to change parameters in the brain (typically synaptic weights) to best improve behaviour (e.g. a desired motor output).

9. What are the two key features of gated recurrent neural networks?

   Answer: Gated RNNs have two defining features, the memory cell and the gating units.

10. What is the key difference between supervised and reinforcement learning in terms of their cost function?

    Answer: The existence of a teacher (or teaching signal) in supervised learning, and a reward signal in reinforcement learning.

11. What types of features are learned by sparse coding algorithms when applied to natural images?

    Answer: Features representing statistics of natural images, such as oriented bars/edges.

12. Give the classical cost function used in sparse coding.

    Answer: cost $= \|U - WV\|_2^2 + \lambda \|V\|_1$, where V is the activity of output neurons and U is the input activity of input neurons. The second term enforces sparsity.

13. Give the value update function used in Q-Learning? How does it differ from TD learning?

    Answer: $\underbrace{Q(S_t, A_t)}_{\text{value}} = Q(S_t, A_t) + (\overbrace{\underbrace{R_{t+1}}_{\text{reward}} + \lambda \underbrace{\max_a Q(S_{t+1}, a)}_{\text{future value}}}^{\text{learned value}} - Q_t(S_t, A_t))$.
    It differs from TD learning in that it takes the action with maximal value for the update, which makes Q-learning an off-policy method.

14. How many different joint activity patterns could a population of N binary neurons generate? Why is it often hard to estimate the probability distribution across these patterns from neural population data?

    Answer: There are 2^N possible binary patterns (1 mark). Estimation is hard because 2^N is a very large number for any reasonable sized number of neurons, say $N \gtrapprox 20$. In contrast, a typical neural recording may only be about an hour, which would correspond to only 3.6×10^5 time bins of 10 ms each (1 mark for something similar to this reasoning).

15. What is the basic idea underlying the dichotomised Gaussian model for neural population data?

    Answer: This model assumes that our observed binary data (spike patterns across the population of neurons) was generated by thresholding an underlying latent (hidden) multivariate gaussian distribution.

Section B: long questions - answer two questions

1. This question is about the Kalman filter.

   (a) [7 marks] Consider two random variables which are conditionally independent with normal distributions:

$$p_{X|H}(x|h) = \frac{1}{\sqrt{2\pi\sigma_X^2}} e^{-(x-h)^2/2\sigma_X^2}$$

$$p_{Y|H}(y|h) = \frac{1}{\sqrt{2\pi\sigma_Y^2}} e^{-(y-h)^2/2\sigma_Y^2} \quad (4)$$

then $p_{X|H}(x|h)p_{Y|H}(y|h)$ is proportional to a normal distribution in h. What is that normal distribution?

Answer:
Well this is just an exercise in matching terms, with the obvious notation, you are told the multiple of the normal distributions is proportional to a normal distribution, hence

$$\exp[-\frac{(h-\mu)^2}{\sigma^2} = C\exp[-\frac{(x-h)^2}{\sigma_X^2}]\exp[\frac{-(y-h)^2}{\sigma_Y^2}] \quad (5)$$

where C is some constant there because the relationship is a proportional one. Then taking the log of both sides and using the fact that this relationship is true for all h to drop the constant terms we get

$$\frac{1}{\sigma^2}(h^2 - 2h\mu) = \frac{1}{\sigma_X^2}(h^2 - 2hx) + \frac{1}{\sigma_Y^2}(h^2 - 2hy) \quad (6)$$

so matching the coefficients of the h terms and of the h^2 terms we get

$$\frac{1}{\sigma^2} = \frac{1}{\sigma_X^2} + \frac{1}{\sigma_Y^2} \quad (7)$$

and

$$\mu = \frac{\sigma^2}{\sigma_X^2}x + \frac{\sigma^2}{\sigma_Y^2}y \quad (8)$$

(b) [8 marks] Consider an object moving at constant speed v so that its position x after a time δt is

$$x(t+\delta t) = x(t) + v\delta t + \xi$$

where ξ is random noise drawn from $\mathcal{N}(0, \sigma_s^2)$. A sensor estimates the position of the object with noise drawn from $\mathcal{N}(0, \sigma_s^2)$. Derive the Kalman gain for optimally estimating the position of the object.

Answer:
Let $h(t)$ be the estimated position with uncertainty $\sigma_h^2(t)$ at time t. After another δt the dead reckoning estimate is $d = h(t) + v\delta t$; that is, we have an estimate for the position at t, the dead reckoning estimate is our estimate for the position at $t+\delta t$ based on our original estimate and the motion. This has uncertainty given by variance $\sigma_d^2 = \sigma_h^2(t) + \sigma_s^2$: this is the sum of the original uncertainty in the estimate and the extra uncertainty that derives from the noise in the motion.

Then we do Bayesian fusion to calculate the new estimated position; this fuses the estimate from dead reckoning and the estimate based

on the sensor; each with its own noise.

$$h(t+\delta t) = \frac{\sigma_h^2(t+\delta t)}{\sigma_d^2} d + \frac{\sigma_h^2(t+\delta t)}{\sigma_s^2} s \tag{9}$$

where s is the sensor estimate and

$$\frac{1}{\sigma_h^2(t+\delta t)} = \frac{1}{\sigma_d^2} + \frac{1}{\sigma_s^2} \tag{10}$$

Rearranging this gives

$$h(t+\delta t) = d + k(s-d) \tag{11}$$

where

$$k = \frac{\sigma_h^2(t+\delta t)}{\sigma_s^2} = \frac{\sigma_d^2}{\sigma_d^2 + \sigma_s^2} \tag{12}$$

This is the same calculation done in the Kalman filter notes, just with a slightly different notation.

(c) [5 marks] Explain what is meant by a forward model for motor control.

Answer:
In a forward model the brain predicts the effect of a motor command, this is used to calculate the next motor command is a feedback control loop; sensor feedback is used to correct the prediction. For movement the prediction will be a dead reckoning estimate.

2. There are two parts to this question, the first is about information theory, the second is about statistical models.

(a) Information theory

i. [3 marks] Define $I(X;Y)$ and give a sufficient condition for $I(X;Y) = 0$ for non-trivial random variables X and Y? **Answer**:

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{13}$$

and this is zero if the argument of the log is zero, that is $p(x,y) = p(x)p(y)$ for all x and y, that is when X and Y are independent.

ii. [4 marks] Calculate the mutual information between random variables X and Y with sample spaces $\mathcal{X} = \{\mathrm{a}, \mathrm{b}, \mathrm{c}\}$ and $\mathcal{Y} = \{\alpha, \beta\}$.

	a	b	c
α	0.5	0.125	0
β	0	0.125	0.25

You can write the answer in terms of $\log 3$ and $\log 5$ if you would prefer.

Answer:

So to marginalize we have, using the obvious abuse of notation $p(X) = \{0.5, 0.25, 0.25\}$ and $p(Y) = \{0.625, 0.375\}$ so the table for $p_X(x)p_Y(y)$ is

	a	b	c
α	5/16	5/32	5/32
β	3/16	3/32	3/32

now

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{14}$$

so

$$\begin{aligned} I(X;Y) &= \frac{1}{2}\log\frac{1/8}{5/16} + \frac{1}{8}\log\frac{1/8}{5/32} + \frac{1}{4}\log\frac{1/8}{3/32} + \frac{1}{4}\log\frac{1/4}{3/32} \\ &= \frac{1}{2}\log\frac{8}{5} + \frac{1}{8}\log\frac{4}{5} + \frac{1}{8}\log\frac{4}{3} + \frac{1}{4}\log\frac{8}{3} \\ &= \frac{11}{4} - \frac{5}{8}\log 5 - \frac{3}{8}\log 3 \approx 0.704 \end{aligned} \tag{15}$$

iii. [3 marks] Show $I(X;Y) = H(X) - H(X|Y)$.

Answer:

So we start with the definition of the mutual information

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{16}$$

and then write $p(x,y) = p(x|y)p(y)$ giving

$$\begin{aligned} I(X;Y) &= \sum_{x,y} p(x,y) \log \frac{p(x|y)}{p(x)} \\ &= \sum_{x,y} p(x,y) \log p(x|y) - \sum_{x,y} p(x,y) \log p(x) \end{aligned} \tag{17}$$

In the first term write $p(x,y) = p(x|y)p(y)$ again to get the definition of $H(X|Y)$ and in the second term sum over y to marginalize $p(x,y)$ to $p(x)$.

(b) Statistical models.

The pairwise and K-pairwise maximum entropy statistical models are often used for neural population data. The K-pairwise model gives the probability for a neural population activity pattern as

$$p(x_1, x_2, \ldots, x_N) = \frac{1}{Z} \exp[\sum_i h_i x_i + \sum_{i \neq j} J_{ij} x_i x_j + V(\sum_i x_i)]$$

i. [3 marks] The equation for the standard pairwise maximum entropy model differs from the above by one term. Which term does it omit?
Answer: The third term on the right hand side $V(\sum_i x_i)$ is omitted from the pairwise model.

ii. [6 marks] How many unique parameters do each of the pairwise and K-pairwise maximum entropy models have?
Answer: The pairwise model has $N + \frac{N(N-1)}{2}$ parameters, the K-pairwise model has $N + \frac{N(N-1)}{2} + N + 1$ parameters. In each case the first N parameters correspond to each of the individual neuron biases, the h_i's. The second set of parameters are the pairwise couplings, J_{ij}'s. There is one of these for every possible pair of neurons. The division by two is because the couplings are symmetric, so $J_{ij} = J_{ji}$. And the K-pairwise model has a further $N+1$ parameters, one for each of the possible numbers of neurons simultaneously active, $k = 0, k = 1, k = 2, \ldots, k = N$. [3 marks for each case. In both cases partial marks will be awarded for an approximately right answer that provides some reasoning.]

iii. [1 mark] Which of the two models more accurately matches neural population data? [1 mark]
Answer: The K-pairwise model more accurately matches the data: it captures some higher-order correlations. Also, since the pairwise model is a special case of the K-pairwise (where all the V terms equal zero), then the K-pairwise model must only do at least as well as or better than the pairwise model.

3. This question is about supervised learning in the brain.

(a) [5 marks] How may the brain implement supervised learning?
Answer: The classical example of supervised learning in the brain is in the cerebellum, where specific error signals appears to be computed [2.5 marks]. However, the cortex may also use some forms of supervised learning by relying on internally generated supervised signals, which may in turn use algorithms akin to the backpropagation algorithm to efficiently update synaptic weights [2.5 marks].

(b) [11 marks] Give one feature that has been suggested to make the backpropagation algorithm used in supervised learning biologically implausible? You should use a simple two layer neural network (with one hidden neuron h, one output neuron v, one input weights, and no biases, see Figure 1) to derive the weight update. Assume the cost function (or error) to be $E = (v - y)^2$, where y is the desired target.
Answer: To illustrate the backprop algorithm we provide the update rule for weight w_{hu_1} (similar for w_{hu_2}) is given by $\frac{\partial E}{\partial w_{hu_1}} =$

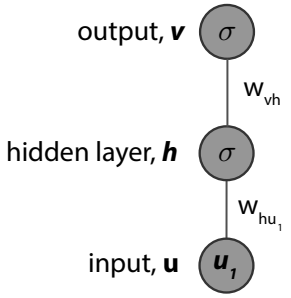

Figure 1: **Schematic of simple feedfoward neural network, with quadratic units $\sigma(x) = x^2$.**

$\frac{\partial E}{\partial v}\frac{\partial v}{\partial h}\frac{\partial h}{\partial w_{hu_1}}$ [1 mark]. For the network considered here, these components are [1 mark for each]:

$$\frac{\partial E}{\partial v} = 2(v - y) \tag{18}$$

$$\frac{\partial v}{\partial h} = \sigma'_v w_{vh} \tag{19}$$

$$\frac{\partial h}{\partial w_{hu_1}} = \sigma'_h u_1 \tag{20}$$

which together yields $\frac{\partial E}{\partial w_{hu_1}} = 2(v-y)\sigma'_v w_{vh}\sigma'_h u_1$, where $\sigma' = 2x$ denotes the derivative of the linear activation function [2 marks]. Note: A good exercise is to represent the different components of the gradient in the schematic of the network.

Any of the following features can be indicated [5 marks]

- *Weight transport problem*: the existence os symmetric weights (i.e. feedback weights derived from backprop are equal to feedforward weights, e.g. w_{vh} (as in Eq. 18). Synaptic weights are unidirectional in the brain, which makes symmetric feedforward and feedback weights implausible.

- *Derivative of activation (input-output) function*: Backprop needs the derivative of the neuronal activation function (as in Eqs. 19 and 20), which is unclear how it is computed biologically.

- *Target or teaching signal*, unclear whether the brain could provide such a teaching signal needed for the cost/error function. However, in principle, different brain areas encoding different aspects of the environment can act as teachers of other brain areas,

or such teaching signal could be provided by an external teacher.

- *Weight learning rule is non-local*: synaptic plasticity (i.e. modification of synaptic weights) typically depends only on locally available information (e.g. pre and postsynaptic activity). However, classical backprop relies on non-local information (error signal, as in $\frac{dE}{dw_{hu_1}}$).

- *Separate learning phase*: Learning needs its only separate phase. Backprop relies on first having a forward phase to update activity and then a backward phase to calculate the error and update the neural network parameters. There is no evidence of such a clear separation between activity and learning in neuroscience.

- *Separate error network*: The use of a separate learning phase with errors, suggests the need for a separate error network. There is no evidence for such separate error networks.

(c) [4 marks] What features do artificial neuronal networks trained with backprop learn (e.g. when trained to discriminate objects in images)? Why do they provide a good match to the activity of neurons in the brain?

Answer: Artificial neural networks (ANNs) learn to detect features similar to the ones observed in the brain [1 mark], such as oriented bars or Gabor-like receptive fields, but also face-like features [1 mark]. Given that both ANNs and the brain need to be optimised to discriminate images [1 mark] and that there is recent evidence for learning principles similar to backprop in the brain [1 mark] this might explain why these networks are good computational models of neuronal data.