UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFR11130 MACHINE LEARNING AND PATTERN RECOGNITION

Tuesday 12th December 2017

09:30 to 11:30

INSTRUCTIONS TO CANDIDATES

Answer QUESTION 1 and ONE other question.

Question 1 is COMPULSORY. If both QUESTION 2 and QUESTION 3 are answered, only QUESTION 2 will be marked.

All questions carry equal weight.

CALCULATORS MAY NOT BE USED IN THIS EXAMINATION

MSc Courses

Convener: G. Sanguinetti
External Examiners: W. Knottenbelt, M. Dunlop, M. Niranjan, E. Vasilaki

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

The following information might be of use in the questions on this paper:

A multivariate Gaussian random variable $\mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ in D dimensions has a probability density function

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{y};\, \boldsymbol{\mu},\, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{y}-\boldsymbol{\mu})\right).$$

1. THIS QUESTION IS COMPULSORY

(a) In a regression problem the standard deviation of the scalar labels is 1.00 on the training set, and 1.03 on the validation set.

i. A student fits a neural network to the training data. It obtains a training set mean square error of 4.63. Explain why something must have gone wrong. Briefly explain why the problem is not overfitting.

Hint: what is the mean square error of always predicting the mean label?

ii. Two more neural networks are fitted to the training data, and mean square errors are computed. The first network obtains 0.80 on the training set, and 1.10 on the validation set. The second obtains 0.10 on the training set and 0.60 on the validation set. Which neural network would you use to make predictions in future and why?

iii. You now move on to another regression problem. The standard deviation of the labels is not close to 1, but you would like to re-use your existing neural network architecture. State how you could transform the labels for this new problem so that the training labels have a standard deviation of one and a mean of zero. [*6 marks*]

(b) Sketch a 2-dimensional dataset with binary labels '×' and '∘' where a nearest neighbour classifier is likely to generalize *worse* if Principal Components Analysis (PCA) is applied to reduce the features to 1 dimension. Explain whether a different linear projection of your data to 1-dimension would give better nearest-neighbour performance than PCA. [*3 marks*]

(c) Someone accidentally includes a feature twice in the inputs to their logistic regression classifier. For some particular k and l, the input features are equal: $x_k^{(n)} = x_l^{(n)}$, for every example $n = 1 \ldots N$.

i. Explain why any settings of the weights w_k and w_l where $w_k + w_l = c$, for some constant c, give equivalent models.

ii. If we apply L2 regularization in training, show that we will set $w_k = w_l$.

Hint: you might find it helpful to assume that the optimal solution has $w_k + w_l = c$, or $w_l = c - w_k$. [*5 marks*]

QUESTION CONTINUES ON NEXT PAGE

(d) An empirical variance can be written as the difference

$$\sigma^2 = s^2 - \mu^2, \qquad \text{where } s^2 = \frac{1}{N}\sum_{n=1}^{N}(x^{(n)})^2, \qquad \text{and} \qquad \mu = \frac{1}{N}\sum_{n=1}^{N} x^{(n)}.$$

There can be numerical problems computing this difference using standard IEEE 64 bit floating point numbers. For example, if each $x^{(n)} \sim \mathcal{N}(10^9, 1)$, then s^2 and μ^2 will both be close to 10^{18}, and their difference of $\approx$1 won't be represented accurately.

State a more accurate way to compute the empirical variance of $\{x^{(n)}\}_{n=1}^{N}$, using standard arithmetic with the same floating point representation. [*2 marks*]

(e) As part of a stochastic variational inference procedure we want to estimate the derivatives of a cost function:

$$c(m, s) = \int \mathcal{N}(x;\, m, s^2)\, f(x)\, \mathrm{d}x = \mathbb{E}_{\mathcal{N}(x;\, m, s^2)}[f(x)],$$

where $f(x)$ is a function we can evaluate, and m, s and x are all scalars.

i. Use the 'reparameterization trick' to rewrite this cost as an expectation under $\mathcal{N}(\nu; 0, 1)$.

ii. Hence write down expressions to estimate $\frac{\partial c}{\partial m}$ and $\frac{\partial c}{\partial s}$ given N samples $\nu^{(n)} \sim \mathcal{N}(0, 1)$, $n = 1 \ldots N$.

You can assume we can evaluate the function $f'(z) = \left.\frac{\partial f(x)}{\partial x}\right|_{x=z}$.

iii. We usually optimize an unconstrained reparameterization of the standard deviation parameter, $a = \log(s)$. How can we estimate $\frac{\partial c}{\partial a}$? [*9 marks*]

2. ANSWER EITHER THIS QUESTION OR QUESTION 3

A gray-scale image is represented by a vector $\mathbf{x}$ containing D pixel intensities. Given two non-overlapping regions $\mathcal{R}_1$ and $\mathcal{R}_2$, the average pixel values for these regions are:

$$a_1 = \frac{1}{|\mathcal{R}_1|} \sum_{d \in \mathcal{R}_1} x_d \,, \qquad a_2 = \frac{1}{|\mathcal{R}_2|} \sum_{d \in \mathcal{R}_2} x_d \,.$$

(a) Define a weight vector $\mathbf{w}$, such that $\mathbf{w}^\top \mathbf{x} = a_1 - a_2$, which measures the change in the average pixel values between the two regions. *[2 marks]*

(b) K different pairs of regions $\{(\mathcal{R}_1^{(k)}, \mathcal{R}_2^{(k)})\}_{k=1}^K$ are selected to construct a set of weights $\{\mathbf{w}^{(k)}\}_{k=1}^K$ as above. These weights define a feature vector:

$$\boldsymbol{\phi}(\mathbf{x}) = [\mathbf{w}^{(1)\top}\mathbf{x} \quad \mathbf{w}^{(2)\top}\mathbf{x} \quad \cdots \quad \mathbf{w}^{(K)\top}\mathbf{x}]^\top.$$

A colleague is keen to use an established logistic regression implementation for a specialized image classification task, but finds using the raw pixels doesn't work. They then used the hand-crafted features $\boldsymbol{\phi}(\mathbf{x})$ to fit logistic regression parameters $(\mathbf{v}, b)$ for the predictor:

$$P(y\!=\!1 \,|\, \mathbf{x}, \mathbf{v}, b) = \sigma(\mathbf{v}^\top \boldsymbol{\phi}(\mathbf{x}) + b), \quad \text{where} \;\; \sigma(a) = 1/(1 + \exp(-a)).$$

i. Explain why using the change-detection features in this way didn't help.

ii. Suggest an incremental improvement, that your colleague could easily adopt, that might immediately improve classification performance. *[4 marks]*

(c) Another colleague tries a convolutional neural network. They don't have much data, so they initialize their parameters with a model trained on another image recognition task.

i. Write down a regularizer that, if added to a cost function, encourages a vector of weights $\mathbf{v}$ to stay close to some pre-trained weights $\mathbf{v}^{(0)}$.

ii. State clearly how stochastic gradient descent optimization could be modified to include your regularization.

iii. How would you control and set the trade-off between fitting the new data, and moving far from the pre-trained weights? *[7 marks]*

(d) Another way to stop the model from moving too far from an initialization is early stopping. Your colleague plans to compute a cost function on a large validation set after the stochastic gradient descent update for each training example. They will stop fitting when the cost on the validation set is worse than after the previous update.

State two problems with their proposed stopping procedure, and two improvements, one to address each problem. *[4 marks]*

QUESTION CONTINUES ON NEXT PAGE

(e) Gaussian processes can be used to model the performance of a neural network as a function of the neural network's 'hyperparameters'. The hyperparameters are any choices that we need to set, except the main weights that we set by gradient-based methods.

The Gaussian process itself also has hyperparameters. For example, ℓ in the covariance function $k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \exp(-\frac{1}{2\ell^2}(\mathbf{x}^{(i)} - \mathbf{x}^{(j)})^\top(\mathbf{x}^{(i)} - \mathbf{x}^{(j)}))$.

i. State three examples of neural network hyperparameters.

ii. What specific feature do Gaussian processes (GPs) have that conventional feedfoward neural networks do not, which makes GPs useful for optimizing the hyperparameters of neural networks? Briefly explain why this feature is useful.

iii. Now consider fitting the Gaussian process. Explain why we should not select the lengthscale ℓ that minimizes mean square error on the Gaussian process's training data. State a more sensible cost function that we could use to optimize ℓ based on the training data.

[*8 marks*]

3. ANSWER EITHER THIS QUESTION OR QUESTION 2

You wish to build a classifier that will predict which product customers will buy. You have a dataset of N purchases $\{\mathbf{x}^{(n)}, y^{(n)}\}_{n=1}^{N}$, where $\mathbf{x}^{(n)}$ is a vector of D features describing the customer who made the nth purchase, and $y^{(n)} \in \{1, \ldots, K\}$ identifies which product was purchased.

(a) A softmax classifier computes activations $a_k = \mathbf{x}^\top \mathbf{w}^{(k)}$, for $k \in \{1, \ldots, K\}$, using weight vectors $\{\mathbf{w}^{(k)}\}$. Write down how these activations are combined to form a prediction $P(y = c \,|\, \mathbf{x}, \{\mathbf{w}^{(k)}\})$ for input features $\mathbf{x}$. *[2 marks]*

(b) The first feature, x_1, is an integer taking values $\{1, 2, 3\}$ indicating whether the customer is from England, Wales, or Scotland. Explain the problem with using this feature in a softmax classifier. How would you fix this problem? *[3 marks]*

(c) For this part, assume it is possible to set the weights so that every purchase is predicted correctly: $y^{(n)} = \arg\max_c P(y = c \,|\, \mathbf{x}, \{\mathbf{w}^{(k)}\})$. Explain a problem with fitting the weights by maximum likelihood in this situation. Name one possible solution to this problem. *[3 marks]*

(d) We set random weights, $\tilde{\mathbf{w}}^{(k)} \sim \mathcal{N}(\mathbf{0}, \mathbb{I}_D)$, independently for each k. We then consider scaling all the weights at once:

$$\mathbf{w}^{(k)} = m\tilde{\mathbf{w}}^{(k)}, \qquad k = 1, \ldots, K,$$

for a scalar m. What happens to the predictions $P(y \,|\, \mathbf{x}, \{\mathbf{w}^{(k)}\})$ as $m \rightarrow 0$, and what happens as $m \rightarrow \infty$? *[3 marks]*

(e) Instead we now add a multiple of the weight vector for class 1 to the weight vector of every class:

$$\mathbf{w}^{(k)} = \tilde{\mathbf{w}}^{(k)} + m\tilde{\mathbf{w}}^{(1)}, \qquad k = 1, \ldots, K.$$

Explain what happens to the predictions $P(y \,|\, \mathbf{x}, \{\mathbf{w}^{(k)}\})$ now as $m \rightarrow 0$ and $m \rightarrow \infty$? *[3 marks]*

The remainder of the question relates to Bayesian inference with linear regression models. D regression weights are given a prior $\mathcal{N}(\mathbf{w};\, \mathbf{0}, \mathbb{I}_D)$ and a likelihood given observations $\mathbf{y}$

$$p(\mathbf{y} \,|\, \mathbf{w}) = \mathcal{N}(\mathbf{y};\, \Phi\mathbf{w},\, \sigma^2\mathbb{I}_N),$$

where Φ and σ^2 are assumed to take known values. The posterior is Gaussian: $p(\mathbf{w} \,|\, \mathbf{y}) = \mathcal{N}(\mathbf{w};\, \mathbf{m}, V)$.

(f) Show that the posterior covariance V doesn't depend on the observed values $\mathbf{y}$. *[4 marks]*

QUESTION CONTINUES ON NEXT PAGE

(g) Write down the posterior mean $\mathbf{m}$ and covariance V in the limit $\sigma^2 \rightarrow \infty$. *[2 marks]*

(h) The value of the function f at test location $\mathbf{x}^{(*)}$, which for known weights is $f = \mathbf{w}^\top \mathbf{x}^{(*)}$, also has a Gaussian posterior:

$$p(f \,|\, \mathbf{y}) = \mathcal{N}(f;\, \bar{f}, s^2).$$

What are the mean $\bar{f}$ and variance s^2 as expressions of $\mathbf{x}^{(*)}$, $\mathbf{m}$ and V? *[3 marks]*

(i) If we knew the weights, $\mathbf{w}$, the model's distribution over a test observation would be:

$$p(y^{(*)} \,|\, \mathbf{x}^{(*)}, \mathbf{w}) = \mathcal{N}(y^{(*)}; \mathbf{w}^\top \mathbf{x}^{(*)}, \sigma^2) = \mathcal{N}(y^{(*)}; f, \sigma^2).$$

What is the predictive distribution for unknown weights, $p(y^{(*)} \,|\, \mathbf{x}^{(*)}, \mathbf{y})$, in terms of $\bar{f}$ and s^2 from the previous part? *[2 marks]*