UNIVERSITY OF SOUTHAMPTON COMP6208W1

SEMESTER 2 EXAMINATION 2015 - 2016

ADVANCED MACHINE LEARNING

DURATION 120 MINS (2 Hours)

This paper contains 4 questions

Answer THREE out of FOUR questions (each question is worth 33 marks)

An outline marking scheme is shown in brackets to the right of each question.

This examination is worth 60%. The coursework was worth 40%.

University approved calculators MAY be used.

A foreign language dictionary is permitted ONLY IF it is a paper version of a direct Word to Word translation dictionary AND it contains no notes, additions or annotations.

12 page examination paper.

Question 1.

Assuming a linear model $f(\boldsymbol{x}|\boldsymbol{w}) = \boldsymbol{w}^\mathsf{T}\boldsymbol{x}$ and given a set of data $\mathcal{D} = \{(\boldsymbol{x}_k, y_k)|k = 1, 2, \ldots, P\}$, we can train the weights by performing a linear least square with a weight decay regularisation by minimising the "error"

$$E(\boldsymbol{w}) = \sum_{k=1}^{P} \left(\boldsymbol{w}^\mathsf{T}\boldsymbol{x}_k - y_k\right)^2 + \nu\|\boldsymbol{w}\|^2.$$

(a) Explain what the regularisation term does.

Indicative Solution for Question 1.

(Students have seen this in the lectures, but this question tests whether they understand the big picture.)

The regularisation term punishes large weights. Since large weights results in large changes in the prediction when the corresponding feature changes, the regularisation term reduces the sensitivity of the model to the data. This should reduce the variance.

[5 marks]

(b) By defining a matrix $\mathbf{X}$ with columns equal $\boldsymbol{x}_k$ and a vector $\boldsymbol{y}$ with components y_k, rewrite the error function, $E(\boldsymbol{w})$ as a matrix equation. Rearrange this to bring together the terms in powers of the weight vector.

Indicative Solution for Question 1.

(Again students have seen something very close to this, but it is a good test of their mastery of linear algebra.)

Writing the targets as a vector $\boldsymbol{y}$ with elements y_k then

$$\begin{aligned} E(\boldsymbol{w}) &= \|\boldsymbol{w}^\mathsf{T}\mathbf{X} - \boldsymbol{y}^\mathsf{T}\|^2 + \nu\|\boldsymbol{w}\|^2 \\ &= \boldsymbol{w}^\mathsf{T}\left(\mathbf{X}\mathbf{X}^\mathsf{T} + \nu\boldsymbol{I}\right)\boldsymbol{w} - 2\boldsymbol{w}^\mathsf{T}\mathbf{X}\boldsymbol{y} + \|\boldsymbol{y}\|^2 \end{aligned}$$

[5 marks]

(c) Find the set of weights, $\boldsymbol{w}^*$, that minimise the error function, $E(\boldsymbol{w})$.

Indicative Solution for Question 1.

(Basic vector algebra.)

The gradient of the error function is given by

$$\nabla E(\boldsymbol{w}) = 2\left(\mathbf{X}\mathbf{X}^{\mathsf{T}} + \nu\boldsymbol{I}\right)\boldsymbol{w} - 2\mathbf{X}\boldsymbol{y}.$$

Setting the gradient to zero we find

$$\boldsymbol{w}^* = \left(\mathbf{X}\mathbf{X}^{\mathsf{T}} + \nu\boldsymbol{I}\right)^{-1}\mathbf{X}\boldsymbol{y}.$$

Strictly to show this minimises $E(w)$, we should show that the matrix of second derivative is positive define.

[3 marks]

(d) By using the singular value decomposition, $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}}$, rewrite the equations for the optimum weights in terms of the matrices $\mathbf{U}$, $\boldsymbol{\Sigma}$ and $\mathbf{V}$.

Indicative Solution for Question 1.

(They have seen something similar in the lectures, but they really need to understand SVD to do this.)

We first note due to the fact that V is an orthogonal matrix

$$\mathbf{X}\mathbf{X}^{\mathsf{T}} = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathsf{T}}\mathbf{U}^{\mathsf{T}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathsf{T}}$$

where Λ is a diagonal matrix with components $\Lambda_{ii} = \Sigma_i^2$. Since U is also an orthogonal matrix $\mathbf{I} = \mathbf{U}\mathbf{I}\mathbf{U}^{\mathsf{T}}$ so that

$$\mathbf{X}\mathbf{X}^{\mathsf{T}} + \nu\boldsymbol{I} = \mathbf{U}\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathsf{T}} + \nu\mathbf{I}\right)\mathbf{U}^{\mathsf{T}}$$

But because U is an orthogonal matrix

$$\left(\mathbf{X}\mathbf{X}^{\mathsf{T}} + \nu\boldsymbol{I}\right)^{-1} = \mathbf{U}\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathsf{T}} + \nu\mathbf{I}\right)^{-1}\mathbf{U}^{\mathsf{T}}.$$

Thus

$$\begin{aligned}\boldsymbol{w}^* &= \mathbf{U}\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathsf{T}} + \nu\mathbf{I}\right)^{-1}\mathbf{U}^{\mathsf{T}}\left(\mathbf{U}\boldsymbol{\Sigma}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\right)\boldsymbol{y}\\ &= \mathbf{U}\left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{\mathsf{T}} + \nu\mathbf{I}\right)^{-1}\boldsymbol{\Sigma}^{\mathsf{T}}\mathbf{V}^{\mathsf{T}}\boldsymbol{y}.\end{aligned}$$

[5 marks]

(e) Thus show that $\boldsymbol{w} = \mathbf{U}\mathbf{M}\mathbf{V}^{\mathsf{T}}\boldsymbol{y}$ where $\mathbf{M}$ is a diagonal matrix. What are the components of $\mathbf{M}$?

Indicative Solution for Question 1.

(Simple algebra provided students understand the structure of Σ)

TURN OVER

Since $\mathbf{\Sigma}$ is a diagonal matrix of singular values (note that it may not be square, but all off-diagonal elements are zero) then

$$\mathbf{M} = \left(\mathbf{\Sigma}\mathbf{\Sigma}^{\mathsf{T}} + \nu\mathbf{I}\right)^{-1}\mathbf{\Sigma}^{\mathsf{T}}$$

which has diagonal elements

$$M_{ii} = \frac{\Sigma_{ii}}{\Sigma_{ii}^2 + \nu}$$

[5 marks]

(f) Explain why using $\nu > 0$ leads to a better conditioned problem than if $\nu = 0$.

Indicative Solution for Question 1.

(Test students understand the big picture of how regularisation is working in this problem.)

Typically, the matrix X will be poorly conditions in that some of the singular values Σ_{ii} are much smaller than others. In computing the optimal weights we multiply by M_{ii}. If $\nu = 0$ this is proportional to $1/\Sigma_{ii}$ which can make the weights very sensitive to the data y. In some case Σ_{ii} might even be zero (for example, if the are more features and data points) and the problem would be unsolvable when $\nu = 0$. In contrast by make $\nu > 0$ we ensure that we can always compute w and furthermore the weights become less sensitive to small changes in the data.

[5 marks]

(g) Explain the Bias-Variance dilemma, and explain how introducing a regularisation term improves the generalisation performance.

Indicative Solution for Question 1.

(Test knowledge of bias-variance dilemma and whether students can make connections between a the theory and the example shown.)

The bias-variance dilemma shows that in expectation the generalisation perform (measure in terms of the squared error), can be seen as a sum of a bias term giving the generalisation error of the average machine and a variance, which encodes how sensitive the machine is to the data. The dilemma arises because a simple machine will typically have a large bias, but small variance, while a complicated machine will have a small bias but large variance.

Adding a regularisation term will reduce the sensitivity of the learning machine to the data and hence reduce the variance. It may increase the bias as the learning machine will not fit the data so well, but this is usually more than compensated for by reducing the variance. Adding a regularisation term allows us to use much more complicated machines with small biases, which otherwise would suffer from over-fitting (high variance).

[5 marks]

Question 2.

(a) Explain in words how SVMs control over-fitting.

Indicative Solution for Question 2.

(This question tests students high-level understanding of machine learning without overtaxing their mathematical abilities.)

SVM's control over-fitting by finding the maximal margin hyper-plane. This would correctly classify any data point from the training set which suffers an error equal to the margin in the direction perpendicular to the hyper-plane. This effectively works like a regularisation term, finding the "simplest" rule amongst all possible rules.

[5 marks]

(b) Explain the kernel trick.

Indicative Solution for Question 2.

(Test for broad understanding of kernel trick.)

The kernel trick allows us to map data into a high dimensional feature space $x \to \phi(x)$. This can be carried out for sufficiently simple machines where parameter optimisation involve the dot product $\phi^{\mathsf{T}}(x)\phi(y)$. This defines a kernel function $K(x, y) = \phi^{\mathsf{T}}(x)\phi(y)$. Provided the kernel function is positive semi-definite, this decomposition is always possible. In fact, we never need to explicitly calculate the extended features $\phi(x)$. This often makes working in the extended feature space very efficient as $K(x, y)$ may be quick to calculate.

[5 marks]

(c) Define what it means for a kernel to be positive semi-definite (giving different properties a positive semi-definite kernel satisfies). Explain why it is necessary for the kernel function of an SVM to be positive semi-definite.

Indicative Solution for Question 2.

(This tests mathematical knowledge at the level of definitions.)

A positive semi-definite kernel satisfies a number of different conditions

(i) All its eigen-functions have non-negative eigenvalues

(ii) The quadratic form is non-negative. That is for all functions $f(x)$

$$\int\int f(\boldsymbol{x})\, K(\boldsymbol{x}, \boldsymbol{y})\, f(\boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \geq 0.$$

TURN OVER

(iii) Any positive semi-definite kernel can be written as a dot (inner) product between two vector functions

$$K(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\phi}^{\mathsf{T}}(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{y}).$$

It is important for a kernel to be positive semi-definite in order that the condition of the margin being positive is meaningful. That is, if the eigenvalues were negative, then the "distance" would not necessarily be non-negative.

[5 marks]

(d) Briefly explain how to train a deep belief network.

Indicative Solution for Question 2.

(Again test a broad understanding of machine learning.)

Deep belief networks are trained in two stages. In the first stage we use unsupervised learning to train the weights layer by layer. This could use restrictive Boltzmann machines or auto-encoders. In RBMs the weights are updated to maximise the likelihood of the input patterns from the previous layer. In auto-encoders the weights are updated so that the input patterns can be reconstructed from the activity of the next layer—the backward weights used to reconstruct the inputs are discarded. RBMs can be seen as a special case of auto-encoders where the backward weights are identical to the forward weights. The new layer forms a representation of the data based on the training data. Ideally higher layers contain high-level abstractions of the data.

Having built several layers of networks an MLP is put on the front of the network. This is trained by back-propagation. Typically errors are also back-propagated to the layers that were initial trained using unsupervised learning.

[8 marks]

(e) Explain why back-propagation is not an effective way to train a deep multi-layer perceptron.

Indicative Solution for Question 2.

(Test understanding of why deep networks are non-trivial to get to work.)

There are two main problems. The first is that there is little gradient information in sigmoidal units when far away from the threshold. Since learning is usually proportional to the gradient, this can slow down learning considerably. The second problem is that there is a huge permutation symmetry and sign (parity) symmetry meaning that every optimum has a huge number of equivalent optima in a completely different direction in search space. As a consequence gradient search is being pulled in very many different directions. In networks with more the two layers the earlier layers are found not to learn.

[5 marks]

(f) Explain the advantages and disadvantages of using an SVM as opposed to a deep belief network.

Indicative Solution for Question 2.

(Test practical understanding of which learning machines are likely to be the most useful.)

SVMs have a unique optima so that it is usually straightforward to know when the training is complete. Although they can over-fit, this is usually well controlled. They tend to work well where there is a limited amount of training data. However, they scale poorly with the number of training examples (typically as P^3). In contrast DBMs typically need a large amount of training data to prevent over-fitting. Even then a lot of care is required to prevent over-fitting. They typically slow down linearly with the number of training examples so can be used with much larger training sets. Where there is a lot of training data, they tend to have better performance than SVMs.

[5 marks]

TURN OVER

Question 3.

(a) You are given a sequence $\mathcal{X} := \{x^{(1)}, \ldots, x^{(12)}\}$ of heads ($x^{(i)} = H$) and tails ($x^{(i)} = T$) which are the outcomes of 12 tosses of a (potentially biased) coin. Describe how you would fit the data to a binomial distribution $B(N, \theta)$ for $N = 12$ and $\theta = Pr(H)$, using maximum likelihood estimation.

[7 marks]

(b) Show that maximising the log-likelihood is equivalent to minimising the Kullback-Leibler divergence between $B(N, \theta)$ and the empirical distribution $\tilde{p}(\mathbf{x})$.

[7 marks]

(c) Discuss how a conjugate Beta prior

$$\frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1}d\theta} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}$$

introduces "psuedo-counts" to affect the estimation of parameters of the binomial distribution. *Hint:* $\Gamma(n) = (n-1)!$ for integer arguments n.

[7 marks]

(d) You are given a data set containing observed values $v^{(n)}$ of variable V. You construct a probability model $p(V = v^{(n)}, H = h^{(n)}|\theta)$ that introduces model hidden variables H that take values $h^{(n)} \in H$ that are associated with observations $v^{(n)}$. To infer the parameters θ you will need to maximise the log likelihood of the observed data

$$\sum_{n=1}^{N} \log p(V = v^{(n)}|\theta).$$

Describe how you would use the EM algorithm to perform this task. Describe how you can impute missing data using the EM algorithm.

[12 marks]

Indicative Solution for Question 3.

(a) (7 marks) The binomial distribution with parameter θ for the probability of heads captures the probability of obtaining n_H heads and $n_T = N - n_H$ tails thus:

$$p(n_H, n_T) = \frac{N!}{n_H! n_T!} \theta^{n_H} (1-\theta)^{n_T}.$$

The log likelihood $\mathcal{L}(\theta; \mathcal{X})$

$$\log\left(\frac{N!}{n_H! n_T!}\right) + n_H \log\theta + (N - n_H)\log(1-\theta)$$

is maximised for $\theta^* = n_H/N$. This follows from taking partial derivatives w.r.t. θ and using a Lagrange multiplier to enforce normalisation. The empirical probability $\tilde{p} = \theta^* = n_H/N$, which is the MLE.

(b) (7 marks) The KL divergence $KL(\tilde{p}\|q)$ is minimised for $p = q$. Hence, for $q(\theta) = B(\theta; N)$ a binomial distribution $q(\theta^*) = \tilde{p}$ in this case. The log likelihood of the empirical distribution is thus maximal.

(c) (7 marks) For prior distribution $p(\theta|\alpha, \beta)$,

$$p(n_H, n_T) = \frac{N!}{n_H! n_T!} \theta^{n_H}(1-\theta)^{n_T} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta^{\alpha-1}(1-\theta)^{\beta-1} = \frac{\Gamma(N+\alpha+\beta)}{\Gamma(n_H+\alpha)\Gamma(n_T+\beta)} \theta^{n_H+\alpha-1}(1-\theta)^{n_T+\beta-1}$$

which is equivalent to adding pseudo-counts $(\tilde{n}_H, \tilde{n}_T) = (\alpha - 1, \beta - 1)$ to the observations of heads and tails.

(d) (12 marks)

Initialise $q_t(h), \theta^t$ at random for $t = 0$, t being the index of the algorithmic iteration step. For parameters θ^t_α corresponding to variables x_α (including both hidden and visible variables) calculate $\langle x_\alpha \rangle_{q_t(h)}$. This is the result of maximising the likelihood of the complete data $x = (v, h)$, the M-step. E-step: These new values θ^{t+1} are used to update the conditionals $q_{t+1}(h|v, \theta^{t+1})$.

For missing data, construct hidden variable distributions for probability of unseen data $q(h)$. Then construct likelihood of total (v, h), the latter weighted by a probability $q(h)$ (E). Maximising this likelihood is equivalent to update of counts for joint distributions using direct observation and averages using $q(h)$ (M). These are the new probabilities for joint distributions. Use this to find posteriors for $q(h|v)$. Set these to be the new $q(h)$. Iterate.

TURN OVER

Question 4.

(a) In a graphical model a node is shaded if it represents an observed variable. By writing down the joint probabilities of the following graphical models determine when A and B are (conditionally or otherwise) independent.

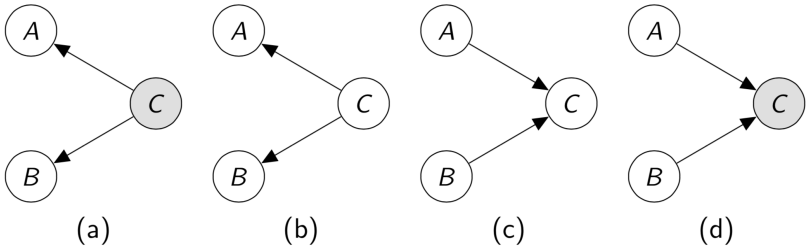

FIGURE 1: A shaded node indicates that a value taken by the corresponding variable is observed.

[8 marks]

(b) What does it mean for a classifier to have high variance? Explain what the concepts of bootstrapping and bagging are, and explain how the variance of a predictive classifier may be reduced if they are used appropriately. [10 marks]

(c) Describe the AdaBoost algorithm. The classification rule $F : X \to Y$ in AdaBoost takes $x \in X$ an input vector produces $Y = \{1, -1\}$ a binary output, where $F(x) = \text{sign}(\sum_m c_m f_m(x))$. What are $f_m(x)$ and c_m and how are they used? Prove that the expectation, taken with respect to the conditional distribution $p(Y = y|X = x)$ of the loss function $L(y, f_m(x)) = \exp(-y f_m(x))$:

$$\mathbb{E}_{Y|X=x}(e^{-y f_m(x)})$$

is minimised at

$$f_m(x) = \frac{1}{2} \log \left(\frac{p(y = 1|x)}{p(y = -1|x)} \right).$$

[7 marks]

(d) Describe the Metropolis-Hastings sampling method and show how the detailed balance condition explains the emergence of a stable probability distribution for the sample.

[8 marks]

TURN OVER

Indicative Solution for Question 4.

(a) (8 marks)

a $A \perp\!\!\!\perp B|C$

b $A \not\perp\!\!\!\perp B$

c $A \perp\!\!\!\perp B$

d $A \not\perp\!\!\!\perp B|C$

(b) (10 marks) The variability of the performance of the classifier on random partitions of the training set is a signal of high variance. Bootstrapping is a technique of obtaining multiple samples from the same data set, with standard error on the bootstrapped sample approximating that from the original population from with the sample was obtained. The averaging of classification results on random bootstrapped samples of the data is called bagging. The reduction in variance can come about only f the bootstrapped samples are sufficiently uncorrelated, from the Central Limit Theorem. Credit given to considerations of the number of attributes p and number of data points n, etc.

(c) (7 marks) The derivative of the expectation value of the loss function $\mathcal{L}(y, f_m(x)) = \exp(-yf_m(x))$ is:

$$\frac{\partial}{\partial f_m(x)}\mathbb{E}_{Y|x}\exp(-yf_m(x)) = \frac{\partial}{\partial f_m(x)}\left[p(Y=1|x)e^{-f_m(x)} + p(Y=-1|x)e^{f_m(x)}\right].$$

The result follows upon setting the derivative to 0.

(d) (8 marks)

Introduce ratio r:

$$r(\theta'|\theta) = \frac{P(\theta')T_t(\theta|\theta')}{P(\theta)T_t(\theta'|\theta)}$$

Set new state θ^{t+1} by the rule $A(\theta^t \to \theta^{t+1})$:

$$\theta^{t+1} = \begin{cases} \theta' & \textbf{with probability } \min(r(\theta'|\theta^t), 1) \quad \textbf{accept new} \\ \theta^t & \textbf{otherwise } (1 - r(\theta'|\theta^t)) \quad\quad\quad\quad\quad\quad\quad\quad \textbf{keep old} \end{cases}$$

The unconditional (joint) distribution $P(\theta_a, \theta_b)$:

$$\begin{aligned} P(\theta^t = \theta_a, \theta^{t-1} = \theta_b) &= P(\theta_b)T_t(\theta_a|\theta_b)A(\theta_b \to \theta_a) \\ &= P(\theta_b)T_t(\theta_a|\theta_b)\tfrac{P(\theta_a)}{P(\theta_b)} = P(\theta_a)T_t(\theta_a|\theta_b). \end{aligned}$$

Detailed balance $T(\theta_b|\theta_a)P(\theta_a) = T(\theta_a|\theta_b)P(\theta_b)$ guarantees equilibrium distribution.

END OF PAPER

