Department of Electrical Engineering, IIT Delhi

ELL784 Introduction to Machine Learning: Major Examination

(Open book/Open Notes) Time: 1 hour Maximum Marks: 38

"Thou shalt not covet thy neighbour's answers"

Name: Entry No.

Assume all symbols to have their 'usual' meanings, as done in the class. No queries allowed: make suitable assumptions. Appropriate marks will be assigned according to their 'reasonability'.

1. **No Support Slack:** Consider the soft-margin SVM formulation as done in the class. Consider the non-negative parameter C. If C is close to zero, should the formulation approach the hard-margin SVM formulation? Or should this happen when C is very large? Give an intuitive explanation. *(2 marks)*

2. **High Perceptron... Hyper Ceptron:** A Perceptron has $y(\mathbf{x}) = h(\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}))$, where $h(a)$ is the signum function, $h(a) = \left\{ \begin{array}{l} +1,\ a \geq 0, t = +1; \\ -1,\ a < 0, t = -1; \end{array} \right\}$, where $\mathbf{x}$ indicates a data point, and t, its target value. We are given N training points, $\{(\mathbf{x}_{(n)}, t_{(n)})\}$, $n \in \{1, N\}$. Consider the following weight update rule, for every mis-classified point $\mathbf{x}_{(n)}$ at iteration $\tau + 1$: $\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \eta\ t_{(n)}\boldsymbol{\phi}(\mathbf{x}_{(n)})$. For this question, assume the learning rate $\eta = 1$. *((3+3+3) marks)*

   (a) Show that for a mis-classified example, $t_{(n)}\mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_{(n)}) < 0$.

   (b) Show that for a mis-classified example, $t_{(n)}\mathbf{w}^{(\tau+1)^T}\boldsymbol{\phi}(\mathbf{x}_{(n)}) < t_{(n)}\mathbf{w}^{(\tau)^T}\boldsymbol{\phi}(\mathbf{x}_{(n)})$

   (c) Write the Perceptron $y(\mathbf{x})$ in terms of a kernel function of point $\mathbf{x}_{(n)}$ and a new point $\mathbf{x}$.

3. **Lightweight Question... it weighs only a Gram:** Consider the kernel example for regularised linear regression, as done in class, where we computed Gram matrix $\mathbf{K}$ from the design matrix $\boldsymbol{\Phi}_{N\times M}$. *((2+4) marks)*

   (a) Show that $\mathbf{K}$ is PSD (Positive Semi-Definite).

   (b) Assume that an eigenvalue-eigenvector decomposition of $\mathbf{K}$ exists. Use this to compute $\boldsymbol{\phi}(\mathbf{x})$ in terms of the eigenvalues and eigenvectors.

4. **Error-prone?** The backpropagation example done in the class had a squared error function. Consider a similar squared error d^2. Define $\rho(d, \alpha) = \frac{d^2}{d^2+\alpha^2}$. Give the physical significance of this function when the squared error is small, and when it is very large. At what value of d does the behaviour of this function change, in terms of α? *((2+2+4) marks)*

5. **Break the chain... not the chain rule:** Consider a neural network as done in class, with D input units $\mathbf{x}$ (indexed $i = 0$ to D), M hidden layer units $\mathbf{z}$ (indexed $j = 0$ to M) and K output units $\mathbf{y}$ (indexed $k = 1$ to K). ($i, j = 0$ handle the constant term case, as done in class). The weights in the two layers of connections are indexed as done in class. Unlike what was done in class, the activation function at both the hidden layer and the output layer is a sigmoid i.e., $h(a_j) = \frac{1}{1+\exp(-a_j)}$ and $\sigma(a_k) = \frac{1}{1+\exp(-a_k)}$. Unlike what was done in class, consider the error function at the kth output neuron to be the cross-entropy function, $E = -\sum_{k=1}^{K} t_k \log y_k + (1 - t_k)\log(1 - y_k)$. *((2+5+6) marks)*

   (a) Write the partial derivative of $h(a)$ completely in terms of $h(a)$ itself.

   (b) Write the chain rule in terms of three 'chain' steps with a possible summation (if required), and show that $\partial E/\partial w_{kj}^{(2)} = (y_k - t_k)z_j$.

   (c) Write the chain rule in terms of three 'chain' steps with a possible summation (if required), and show that $\partial E/\partial w_{ji}^{(1)} = \sum_{k=1}^{K}(y_k - t_k)w_{kj}^{(2)} z_j(1 - z_j)x_i$.