10 Machine Learning and Bayesian Inference (SBH)

A *linear maximum-margin classifier* computes a function

$$f_{\mathbf{w},w_0}(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

and assigns a class as $\text{sgn}(f_{\mathbf{w},w_0}(\mathbf{x}))$ where $\text{sgn}(x) = 1$ if $x \geq 0$ and $\text{sgn}(x) = -1$ otherwise. It is trained using a training sequence $((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m))$ and the aim in training is to solve the problem

$$\text{argmax}_{\mathbf{w},w_0} \left(\min_i \frac{y_i f_{\mathbf{w},w_0}(\mathbf{x}_i)}{||\mathbf{w}||} \right). \tag{1}$$

(*a*) Give a brief explanation of how Equation 1 is derived. You may assume that the distance from $\mathbf{x}'$ to the line $f_{\mathbf{w},w_0}(\mathbf{x}) = 0$ is $|f_{\mathbf{w},w_0}(\mathbf{x}')|/||\mathbf{w}||$. [2 marks]

(*b*) Why is Equation 1 not used in practice? Explain how an alternative optimization problem is derived that can form the basis of a practical learning algorithm. You need only derive a statement of the primal optimization problem. [3 marks]

(*c*) Explain how the linear maximum-margin classifier can be modified to be nonlinear and to allow misclassification of the training examples. Give a derivation of the modified optimization problem needed for training. You need only derive a statement of the primal optimization problem. [4 marks]

(*d*) As part of the derivation of the full learning algorithm we find that the function f might be expressible in terms of m new parameters α_i as

$$f_{\alpha_1,\ldots,\alpha_m,w_0}(\mathbf{x}) = \sum_{i=1}^{m} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + w_0. \tag{2}$$

Explain the purpose of K in Equation 2 and explain why its use might be beneficial. [3 marks]

(*e*) Your boss can not afford to provide you with a solver capable of training your system using the algorithm in Parts (*c*) and (*d*). Your boss does however provide you with a solver for *linear programs*. For a matrix $\mathbf{A}$ and vectors $\mathbf{b}$ and $\mathbf{c}$, this solves problems of the form

Find $\mathbf{x}$ minimizing $\mathbf{b}^T\mathbf{x}$ with constraints $\mathbf{A}\mathbf{x} \geq \mathbf{c}$ and $\mathbf{x} \geq \mathbf{0}$.

Suggest a way in which you could use this optimizer to (approximately) train your system. [8 marks]