MATHEMATICS OF MACHINE LEARNING Part II
Example Sheet 3 (of 3) RDS/Lent 2022

1. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a strictly convex function and suppose $C \subseteq \mathbb{R}^d$ is a convex set. Suppose $x_1, x_2 \in C$ satisfy $f(x_1) = f(x_2) = \inf_{x \in C} f(x)$. Show that $x_1 = x_2$.

2. Let $f : \mathbb{R}^d \to \mathbb{R}$ be convex.

   (a) Show that $\partial(\alpha f)(x) = \{\alpha g : g \in \partial f(x)\}$ for $\alpha > 0$.

   (b) Show that if $h(x) = f(Ax+b)$ for $A \in \mathbb{R}^{d \times m}$ and $b \in \mathbb{R}^d$, then $\partial h(x) = A^T \partial f(Ax+b)$ in the special case where $d = 1$. [*Hint: Show that any $v \in \partial h(x)$ must lie in the row space of A by considering the orthogonal projection P onto the row space of A i.e. such that $AP = A$ and $P = P^T = P^2$.*]

3. Show that $\partial\|\beta\|_1 = \{b : \text{ for each } j, b_j \in [-1,1] \text{ and } b_j = \operatorname{sgn}(\beta_j) \text{ if } \beta_j \neq 0\}$. [*Hint: Use 2 (b).*]

4. Show that $x \in \mathbb{R}^d$ minimises $f : \mathbb{R}^d \to \mathbb{R}$ if and only if $0 \in \partial f(x)$.

5. This question derives the form of the projection onto an ℓ_1-norm constraint set.

   (a) Fix $x \in \mathbb{R}^p$ and $\gamma > 0$, and let $g(\beta) := \|\beta - x\|_2^2/2 + \gamma\|\beta\|_1$. Show that g is minimised over $\beta \in \mathbb{R}^p$ by
   $$\beta_j^* = \max(|x_j| - \gamma, 0)\operatorname{sgn}(x_j).$$
   [*Hint: Use 3 and 4.*]

   (b) Argue that if β^* above has $\|\beta^*\|_1 = \lambda$, then β^* is the projection of x onto the set $C = \{z : \|z\|_1 \leq \lambda\}$ i.e. $\beta^* = \pi_C(x)$.

6. Consider a version of stochastic gradient descent for minimising
   $$f(\beta) = \frac{1}{n}\sum_{i=1}^{n} \ell(h_\beta(x_i), y_i)$$
   (assumed here to be differentiable) over $\beta \in C \subseteq \mathbb{R}^p$ where C is closed and convex. We take $U_1, \ldots, U_{k-1}$ uniformly distributed on $\{1, \ldots, p\}$ and writing $\beta^{(s)} \in \mathbb{R}^p$ for the sth iterate we take
   $$\tilde{g}_s = e_{U_s} \frac{\partial f}{\partial \beta_{U_s}}\bigg|_{\beta^{(s)}}.$$
   Show that under the setup of Theorem 26 (on the convergence of gradient descent), the output $\bar{\beta}$ of the algorithm set out above, with a suitable step size $\eta > 0$ you should specify, satisfies
   $$\mathbb{E}f(\bar{\beta}) - f(\hat{\beta}) \leq 2LR\sqrt{\frac{p}{k}}.$$

7. The following result shows that the theory for stochastic gradient descent can be used to obtain some forms of generalisation error bounds. Let $(X_1, Y_1), \ldots, (X_n, Y_n)$ be i.i.d. input–output pairs. Consider empirical risk minimisation with logistic loss ϕ where $\mathcal{X} = \{x \in \mathbb{R}^p : \|x\|_2 \leq C\}$ and $\mathcal{H} = \{x \mapsto x^T\beta : \|\beta\|_2 \leq \lambda\}$. Let π denote projection onto $\{\beta : \|\beta\|_2 \leq \lambda\}$. Let $\beta_1 \in \mathbb{R}^p$ be the 0 vector and define iteratively for $i = 1, \ldots, n-1$,
   $$g_i = Y_i X_i \phi'(Y_i X_i^T \beta_i),$$
   $$\beta_{i+1} = \pi(\beta_i - \eta g_i).$$

[Note the β_i above are vectors.] Let $\bar{\beta} = \frac{1}{n}\sum_{i=1}^n \beta_i$ and set $\bar{h}(x) = x^T\bar{\beta}$. Show that for some step size $\eta > 0$ you should specify,

$$\mathbb{E}R_\phi(\bar{h}) - R_\phi(h^*) \leq \frac{2C\lambda}{\log(2)\sqrt{n}}.$$

[*Hint: Write the risk itself in the form* $\mathbb{E}\tilde{f}(\beta; U)$ *for some* U.]

8. Consider the Adaboost algorithm and assume that at no iteration does any $h \in \mathcal{B}$ perfectly classify the data.

   (a) Show that

   $$\frac{\sum_{i=1}^n w_i^{(m+1)}}{\sum_{i=1}^n w_i^{(m)}} = 2\sqrt{\widehat{\mathrm{err}}_m(1-\widehat{\mathrm{err}}_m)}$$

   where $\widehat{\mathrm{err}}_m := \mathrm{err}_m(\hat{h}_m)$.

   (b) Assume that for each m, $\widehat{\mathrm{err}}_m \leq 1/2 - \gamma$ for some $\gamma > 0$. Show that the empirical risk of the Adaboost output decreases exponentially fast with M:

   $$\begin{aligned}\frac{1}{n}\sum_{i=1}^n \exp(-Y_i\hat{f}_M(X_i)) &= \prod_{m=1}^M 2\sqrt{\widehat{\mathrm{err}}_m(1-\widehat{\mathrm{err}}_m)} \\ &\leq \exp(-2\gamma^2 M).\end{aligned} \tag{1}$$

9. (Continuation of 8.)

   (a) Let $\mathcal{B}$ be a class of base classifiers $h : \mathcal{X} \to \mathbb{R}$ such that $h \in \mathcal{B} \Rightarrow -h \in \mathcal{B}$. Suppose $\mathcal{B}$ has finite VC dimension and let

   $$\mathcal{H} = \left\{ \sum_{m=1}^M \beta_m h_m : \|\beta\|_1 \leq 1,\ h_m \in \mathcal{B} \text{ for } m = 1, \ldots, M \right\}.$$

   Explain why for $x_{1:n} \in \mathcal{X}^n$,

   $$\hat{\mathcal{R}}(\mathcal{H}(x_{1:n})) \leq \sqrt{\frac{2\mathrm{VC}(\mathcal{B})\log(n+1)}{n}}.$$

   (b) Given input–output pairs $(X_i, Y_i)_{i=1}^n$ taking values in $\mathcal{X}\times\{-1,1\}$, let $\hat{f}_M = \sum_{m=1}^M \hat{\beta}_m\hat{h}_m$ be the output of the Adaboost algorithm with base classifier class $\mathcal{B}$. With the assumption of 8 (b) that $0 < \widehat{\mathrm{err}}_m \leq 1/2 - \gamma$ for some $1/2 > \gamma > \rho > 0$, show that

   $$\frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{\hat{f}_M(X_i)Y_i \leq \rho\|\hat{\beta}\|_1\}} \leq \exp\{-2M(\gamma^2 - c\rho)\}, \quad \text{where} \quad c = \frac{1}{4}\log\left(\frac{1+2\gamma}{1-2\gamma}\right),$$

   and $\hat{\beta} := (\hat{\beta}_m)_{m=1}^M$. [*Hint: Use* $\mathbb{1}_{\{u \leq b\}} \leq \exp(b-u)$ *and* (1). *You may further use the fact that* $u \mapsto u^{1-\rho}(1-u)^{1+\rho}$ *is increasing for* $0 < u < 1/2 - \rho$.]

   (c) Show, using Qu. 12 of Ex. sheet 2, that writing $\hat{h} := \mathrm{sgn} \circ \hat{f}_M$, we have that with probability at least $1-\delta$, the misclassification risk $R(\hat{h})$ satisfies

   $$R(\hat{h}) \leq \exp\{-2M(\gamma^2 - c\rho)\} + \frac{2}{\rho}\sqrt{\frac{2\mathrm{VC}(\mathcal{B})\log(n+1)}{n}} + \sqrt{\frac{\log(1/\delta)}{2n}}.$$

10. Consider the optimisation problem of finding the optimal split point s when fitting a regression tree:

$$\min_{s\in\mathcal{S}}\left\{\min_{c_1\in\mathbb{R}}\sum_{i:X_{ij}\le s}(Y_i-c_1)^2+\min_{c_2\in\mathbb{R}}\sum_{i:X_{ij}>s}(Y_i-c_2)^2\right\},$$

where $\mathcal{S}$ is the set of midpoints among $X_{1j}<\cdots<X_{nj}$. Describe how this optimisation over s can be performed using $O(n)$ computational operations (i.e. operations involving addition, subtraction, multiplication and division of pairs of real numbers).

11. In this question, we will study the Rademacher complexity of a simple neural network with a single hidden layer of m nodes, reLU activation function ψ, and additional ℓ_2-norm constraints on the parameters. Specifically, consider the set $\mathcal{H}$ of hypotheses of the form

$$h(x)=\sum_{j=1}^{m}\alpha_j\psi(\beta_j^Tx)$$

where $\alpha_j\in\mathbb{R}$ and $\beta_j\in\mathbb{R}^p$ for $j=1,\dots,m$, with the constraints $\|\alpha\|_2\le\lambda_\alpha$ and $\max_{j=1,\dots,m}\|\beta_j\|_2\le\lambda_\beta$. Let $\mathcal{X}=\{x\in\mathbb{R}^p:\|x\|_2\le C\}$ and take $x_{1:n}\in\mathcal{X}^n$.

(a) By considering $\mathcal{B}:=\{x\mapsto\psi(b^Tx):\|b\|_2\le\lambda_\beta\}$ and $\mathcal{B}'=\{x\mapsto b^Tx:\|b\|_2\le\lambda_\beta\}$, show that

$$\mathbb{E}\left(\sup_{b:\|b\|_2\le\lambda_\beta}\left|\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i\psi(b^Tx_i)\right|\right)\le\frac{2C\lambda_\beta}{\sqrt{n}},$$

where $\varepsilon_{1:n}$ are i.i.d. Rademacher random variables. [*Hint: Apply the contraction lemma.*]

(b) Let us introduce the set of vector-valued functions $\mathcal{G}:=\{g:=(g_1,\dots,g_m):g_j\in\mathcal{B}\text{ for }j=1,\dots,m\}$. Show that

$$\hat{\mathcal{R}}(\mathcal{H}(x_{1:n}))\le\lambda_\alpha\mathbb{E}\left(\sup_{g\in\mathcal{G}}\left\|\frac{1}{n}\sum_{i=1}^{n}\varepsilon_ig(x_i)\right\|_2\right).$$

(c) Finally show that

$$\hat{\mathcal{R}}(\mathcal{H}(x_{1:n}))\le 2C\lambda_\alpha\lambda_\beta\sqrt{\frac{m}{n}}.$$