Mehryar Mohri
Foundations of Machine Learning 2020
Courant Institute of Mathematical Sciences
Homework assignment 3
Nov 14, 2020
Due: Nov 24, 2020 [before class starts].

A. Kernel methodss

1. Graph kernel. Let $G = (\mathcal{V}, \mathcal{E})$ be an undirected graph with vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. $\mathcal{V}$ could represent a set of documents or biosequences and E the set of connections between them. Let $w[e] \in \mathbb{R}$ denote the weight assigned to edge $e \in \mathcal{E}$. The weight of a path is the product of the weights of its constituent edges. Show that the kernel K over $\mathcal{V} \times \mathcal{V}$ where $K(p, q)$ is the sum of the weights of all paths of length two between p and q is PDS (*Hint*: you could introduce the matrix $W = (W_{pq})$, where $W_{pq} = 0$ when there is no edge between p and q, W_{pq} equal to the weight of the edge between p and q otherwise).

   Solution:

   Graph kernel For any $i, j \in \mathcal{V}$, let $\mathcal{E}[i, j]$ denote the set of edges between i and j which is either reduced to one edge or is empty. For convenience, let $\mathcal{V} = \{1, \ldots, n\}$ and define the matrix $\mathbf{W} = (W_{ij}) \in \mathbb{R}^{n \times n}$ by $W_{ij} = 0$ if $\mathcal{E}[i, j] = \emptyset$, $W_{ij} = w[e]$ if $e \in \mathcal{E}[i, j]$. Then, we can write

   $$K(p, q) = \sum_{e \in \mathcal{E}[p,r], e' \in \mathcal{E}[r,q]} w[e]w[e'] = \sum_r W_{pr} W_{rq} = W^2_{pq}.$$

   Let $\mathbf{K} = (K_{pq})$ denote the kernel matrix. Since $\mathbf{W}$ is symmetric For any vector $\mathbf{X} \in \mathbb{R}^n$, $\mathbf{X}^\top \mathbf{K} \mathbf{X} = \mathbf{X}^\top \mathbf{W}^2 \mathbf{X} = \mathbf{X}^\top \mathbf{W}^\top \mathbf{W} \mathbf{X} = \|\mathbf{W}\mathbf{X}\|^2 \geq 0$. Thus, the eigenvalues of $\mathbf{K}$ are non-negative. The same holds similarly for the kernel matrix restricted to any subset of $\mathcal{V}$, thus K is PDS. □

2. Pixel kernel. This problem consists of showing that a kernel useful in applications is PDS.

   (a) Show that $S \colon (z, z') \mapsto \int_0^{+\infty} 1_{t \in [0,z]} 1_{t \in [0,z']} \, dt$ is a PDS kernel defined over $\mathbb{R} \times \mathbb{R}$.

   Solution: $(f, g) \mapsto \int_{t=0}^{+\infty} f(t)g(t)dt$ is a positive semi-definite inner product over the set $L_2(\mathbb{R})$. Let $\langle \cdot, \cdot \rangle$ denote that inner product. Then, for any $z, z' \in \mathbb{R}$, $S(z, z') = \langle t \mapsto 1_{t \in [0,z]}, t \mapsto 1_{t \in [0,z']} \rangle = \langle \Psi(z), \Psi(z') \rangle$, where Ψ is the feature mapping $\Psi \colon z \mapsto t \mapsto 1_{t \in [0,z]}$. Thus, S can be expressed as an inner product and is therefore PDS. □

(b) Use the previous question to prove that the kernel $K_\mu \colon (\mathbf{x}, \mathbf{x}') \mapsto \prod_{k=1}^N e^{\min(|x_k|^\mu, |x'_k|^\mu)}$, where x_k is the kth coordinate of $\mathbf{x}$, is a PDS kernel defined over $\mathbb{R}^N \times \mathbb{R}^N$, for any $\mu \geq 0$.

Solution: Observe that, for any $\mu \geq 0$, $\min(|u|^\mu, |u'|^\mu) = S(|u|^\mu, |u'|^\mu)$. Thus, for any $\mu \geq 0$ and $k \in [N]$, $(\mathbf{x}, \mathbf{x}') \mapsto \min(|x_k|^\mu, |x'_k|^\mu)$ is a PDS kernel. The sum of these PDS kernels (over all $k \in [N]$) is therefore also PDS and the composition with the power series exp whose coefficients are non-negative and whose radius of convergence is infinite is also PDS. □

B. Boosting

1. Logistic loss boosting.

(a) Show that $\Phi \colon u \mapsto \log_2(1 + e^{-u})$ defines a convex and decreasing function upper-bounding $u \mapsto 1_{u \leq 0}$.

Solution: This is straightforward. For any $u \in \mathbb{R}$,

$$\Phi'(u) = -\frac{1}{\log(2)} \frac{e^{-u}}{1 + e^{-u}} = -\frac{1}{\log(2)} \frac{1}{1 + e^{u}} < 0.$$

Thus, Φ is decreasing. For any $u \in \mathbb{R}$,

$$\Phi''(u) = \frac{1}{\log(2)} \frac{e^{u}}{(1 + e^{u})^2} > 0.$$

Thus, Φ is convex. For $u \leq 0$, we have $e^{-u} \geq 1$, thus $\log_2(1 + e^{-u}) \geq \log_2(1 + 1) = 1$. For $u > 0$, since log is increasing, $\log_2(1 + e^{-u}) \geq \log_2(1) = 0$.

(b) Let $\mathcal{H} = \{h_1, \ldots, h_N\}$ be a finite hypothesis set serving as a family of base predictors taking values in $\{-1, +1\}$. Define the objective $F(\boldsymbol{\alpha})$ for a boosting-type algorithm using Φ, instead of the exponential function used in AdaBoost, for a given sample of size m and show that it is a convex function of $\boldsymbol{\alpha}$.

Solution: The objective function can be written for any $\boldsymbol{\alpha} \in \mathbb{R}^N$, $\boldsymbol{\alpha} \geq 0$ as follows:

$$F(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \Phi\left(y_i \sum_{j=1}^{N} \alpha_j h_j(x_i)\right).$$

This a convex function of $\boldsymbol{\alpha}$ since Φ is convex and composition with an affine function of $\boldsymbol{\alpha}$ preserves convexity. □

(c) Determine the best direction at iteration t if you apply coordinate descent to F. You should adopt a notation similar to the one used in class and define D_t, Z_t, and $\epsilon_{t,k}$ for $t \in [T]$ and $k \in [N]$. What should be the boosting condition on $\epsilon_{t,k}$ for the best direction k?

Solution: For any $\eta > 0$ and direction $k \in [N]$, we can write:

$$F(\alpha_{t-1} + \eta \mathbf{e}_k) = \frac{1}{m} \sum_{i=1}^{m} \Phi\left(y_i \sum_{j=1}^{N} \alpha_{t-1,j} h_j(x_i) + y_i \eta h_k(x_i)\right).$$

Thus, the directional derivative of F along $\mathbf{e}_k$ is given by:

$$\begin{aligned} F'(\alpha_{t-1}, \mathbf{e}_k) &= \frac{1}{m} \sum_{i=1}^{m} y_i h_k(x_i) \Phi'\left(y_i \sum_{j=1}^{N} \alpha_{t-1,j} h_j(x_i)\right) \\ &= -\frac{1}{m} \sum_{i=1}^{m} \frac{1}{\log(2)} \frac{y_i h_k(x_i)}{1 + e^{y_i \sum_{j=1}^{N} \alpha_{t-1,j} h_j(x_i)}} \\ &= -\frac{1}{m} \sum_{i=1}^{m} y_i h_k(x_i) D_t(i) Z_t \\ &= -\frac{Z_t}{m} [(1 - \epsilon_{t,k}) - \epsilon_{t,k}] \\ &= \frac{Z_t}{m} (2\epsilon_{t,k} - 1), \end{aligned}$$

where $D_t(i) Z_t = \frac{1}{\log(2)} \frac{1}{1 + e^{y_i \sum_{j=1}^{N} \alpha_{t-1,j} h_j(x_i)}}$. In view of that, the direction maximizing $|F(\alpha_{t-1} + \eta \mathbf{e}_k)|$ is the one with the minimal weighted error $\epsilon_{t,k}$. For coordinate descent to succeed, we need $\epsilon_{t,k} < \frac{1}{2}$ for that best direction.

(d) Show that for any $(u, v) \in \mathbb{R}^2$, the following inequality holds:

$$\Phi(u + v) - \Phi(u) \leq -\Phi'(u)(e^{-v} - 1).$$

Solution:

$$\begin{aligned}
\Phi(u+v) - \Phi(u) &= \log_2 \left[\frac{1+e^{-(u+v)}}{1+e^{-u}} \right] \\
&= \log_2 \left[\frac{1+e^{-u}+e^{-(u+v)}-e^{-u}}{1+e^{-u}} \right] \\
&= \log_2 \left[1 + \frac{e^{-v}-1}{1+e^{u}} \right] \\
&\leq \frac{1}{\log(2)} \frac{e^{-v}-1}{1+e^{u}} && (\log(1+z) \leq z) \\
&= -\Phi'(u)(e^{-v}-1).
\end{aligned}$$

(e) Use that to show the following:

$$F(\alpha_{t-1} + \eta \mathbf{e}_k) - F(\alpha_{t-1}) \leq \frac{1}{m} \sum_{i=1}^{m} D_t(i) Z_t [e^{-\eta y_i h_k(x_i)} - 1].$$

Solution: Observe that $D_t(i)Z_t = -\Phi'(y_i \sum_{j=1}^N \alpha_{t-1,j} h_j(x_i))$. Then, using the result of the previous question,

$$\begin{aligned}
F(\alpha_{t-1} + \eta \mathbf{e}_k) - F(\alpha_{t-1}) &\leq \frac{1}{m} \sum_{i=1}^{m} -\Phi'(y_i \sum_{j=1}^{N} \alpha_{t-1,j} h_j(x_i)) [e^{-\eta y_i h_k(x_i)} - 1] \\
&= \frac{1}{m} \sum_{i=1}^{m} D_t(i) Z_t [e^{-\eta y_i h_k(x_i)} - 1].
\end{aligned}$$

(f) To determine the best step η for a given direction k, we can minimize the upper bound of the previous question. Show that this is syntactically the same minimization as the one for finding the best step in AdaBoost. Give the expression of the step at iteration t.

Solution: Minimizing that upper bound is equivalent to minimizing $\frac{1}{m} \sum_{i=1}^m D_t(i) e^{-\eta y_i h_k(x_i)}$, which is the quantity minimized by AdaBoost to determine the step. The expression of D_t is different here but we can immediately use the result known for AdaBoost to determine the step at iteration t: $\eta = \frac{1}{2} \log \left[\frac{1-\epsilon_{t,k}}{\epsilon_{t,k}} \right]$.

(g) Give the full pseudocode of the algorithm.

Solution: This is straightforward based on the previous results.

(h) Give a margin-based generalization bound for this algorithm.

Solution: Straightforward using the ensemble Rademacher complexity bound presented in class: fix $\rho > 0$; for any $\delta > 0$, with probability at least $1 - \rho$,

$$R\left(\frac{\sum_{t=1}^{T} \alpha_t h_t}{\|\boldsymbol{\alpha}\|_1}\right) \leq \widehat{R}_{S,\rho}\left(\frac{\sum_{t=1}^{T} \alpha_t h_t}{\|\boldsymbol{\alpha}\|_1}\right) + \frac{2}{\rho}\mathfrak{R}_m(H) + \sqrt{\frac{\log \frac{1}{\delta}}{2m}}.$$

(i) Compare empirically AdaBoost and this logistic loss boosting algorithm on the same dataset as the one used in the previous homework by choosing T via cross-validation as multiples of 100.