Mehryar Mohri
Foundations of Machine Learning 2023
Courant Institute of Mathematical Sciences
Homework assignment 3
October 31, 2023
Due: November 14, 2023

A. Kernel PCA

Read the *Dimensionality Reduction* Chapter 15 in the course textbook Foundations of ML with a focus on PCA and Kernel PCA. Sections 15.1 and 15.2 are recommended. In this problem we will analyze a hypothesis set based on KPCA projection. Let $K(x, y)$ be a kernel function, $\Phi_K(x)$ be its corresponding feature map and $S = \{x_1, \ldots, x_m\}$ be a sample of m points. When Π is the rank-r KPCA projection, we define the (regularized) hypothesis set of linear separators in the RKHS $\mathbb{H}$ of kernel K as

$$\mathcal{H} = \{x \mapsto \langle \mathbf{w}, \Pi\Phi_K(x)\rangle_{\mathbb{H}} \colon \|\mathbf{w}\|_{\mathbb{H}} \leq 1\}. \tag{1}$$

This hypothesis set essentially means that the input data is projected onto a smaller dimensional subspace of the RKHS before fitting a separation hyperplane. This problem will show that we can use the eigenvectors and eigenvalues of the sample kernel matrix to give a closed form expression for the functions $h \in \mathcal{H}$ without a need for explicit representation of the RKHS itself.

Let $\mathbf{K}$ be the sample kernel matrix for kernel K evaluated on m points of sample S, that is $\mathbf{K}_{i,j} = K(x_i, x_j)$. Let $\lambda_1, \ldots, \lambda_r$ be the top r (nonzero) eigenvalues of $\mathbf{K}$ with the corresponding eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$. Denote the j-th element of vector $\mathbf{v}_i$ as $[\mathbf{v}_i]_j$. Follow the subproblems below to derive the explicit representation of $h \in \mathcal{H}$.

1. Assume that the feature maps $\Phi_K(x)$ are centered on sample S and recall that the sample covariance operator is $\Sigma = \sum_{i=1}^m \frac{1}{m}\Phi_K(x_i)\Phi_K(x_i)^\top$. Prove that $h(x) = \sum_{i=1}^r \alpha_i \langle \mathbf{u}_i, \Phi_K(x)\rangle_{\mathbb{H}}$ for some $\alpha_i \in \mathbb{R}$, where $\mathbf{u}_1, \ldots, \mathbf{u}_r$ are the eigenvectors of Σ corresponding to its top r eigenvalues.

   Solution: This is a direct application of the orthonormal basis $\mathbf{u}_1, \ldots, \mathbf{u}_r$.

$$\begin{aligned} h(x) &= \left\langle \mathbf{w}, \mathbf{U}_r\mathbf{U}_r^\top\Phi_K(x)\right\rangle_{\mathbb{H}} \\ &= \left\langle \mathbf{w}, \sum_{i=1}^r \mathbf{u}_i\mathbf{u}_i^\top\Phi_K(x)\right\rangle_{\mathbb{H}} \\ &= \sum_{i=1}^r \langle \mathbf{w}, \mathbf{u}_i\rangle_{\mathbb{H}}\langle \mathbf{u}_i, \Phi_K(x)\rangle_{\mathbb{H}}. \end{aligned}$$

   Denoting $\alpha_i = \langle \mathbf{w}, \mathbf{u}_i\rangle_{\mathbb{H}}$, we obtain the solution.

2. Prove that $\mathbf{u}_i = \mathbf{X}\frac{\mathbf{v}_i}{\sqrt{\lambda_i}}$, where $\mathbf{X} = [\Phi_K(x_1), \ldots, \Phi_K(x_m)]$.

   Solution: For more details see Ch15, Section 15.2 of the textbook. The eigenvalue-eigenvector equation for Σ is

$$\Sigma\mathbf{u}_i = \gamma_i\mathbf{u}_i.$$

   Substituting $\Sigma = \frac{1}{m}\mathbf{X}\mathbf{X}^\top$ and $\mathbf{u}_i = \mathbf{X}w_i$ for some $w_i \in \mathbb{R}^m$ since $\mathbf{u}_i$ belongs to the span of $\mathbf{X} = [\Phi_K(x_1), \ldots, \Phi_K(x_m)]$. Also multiplying by $\mathbf{X}^\top$ from the left, we get

$$\frac{1}{m}\big(\mathbf{X}^\top\mathbf{X}\big)\big(\mathbf{X}^\top\mathbf{X}\big)w_i = \gamma_i\big(\mathbf{X}^\top\mathbf{X}\big)w_i.$$

   Divide both sides by m,

$$\left(\frac{1}{m}\mathbf{K}\right)^2 w_i = \frac{\gamma_i}{m}\mathbf{K}w_i.$$

   It can be shown that the solution to the equation above is $w_i = \frac{\mathbf{v}_i}{\sqrt{\lambda_i}}$, which directly leads to $\mathbf{u}_i = \mathbf{X}\frac{\mathbf{v}_i}{\sqrt{\lambda_i}}$.

3. Using the result above, prove that any function $h \in \mathcal{H}$ can be represented as

$$h(x) = \sum_{i=1}^{r} \sum_{j=1}^{m} \frac{\alpha_i}{\sqrt{\lambda_i}} K(x_j, x)[\mathbf{v}_i]_j,$$

for some $\alpha_i \in \mathbb{R}$.

Solution:

$$\begin{aligned}\langle \mathbf{u}_i, \Phi_K(x) \rangle_{\mathbb{H}} &= \Phi_K^\top(x) \mathbf{X} \frac{\mathbf{v}_i}{\sqrt{\lambda_i}} \\ &= \frac{1}{\sqrt{\lambda_i}} \sum_{j=1}^{m} K(x_j, x)[\mathbf{v}_i]_j.\end{aligned}$$

Substituting the above in the result from part 1 provides the final expression for $h(x)$.

4. Bonus question: derive the Rademacher complexity bound on the hypothesis set $\mathcal{H}$ defined in this problem.

Solution: Use the standard techniques for deriving generalization bounds described in this course, as well as Cauchy-Schwarz inequality and Jensen's inequality. For example, one can derive an upper bound $O\left(\sqrt{\frac{\mathrm{Tr}(\mathbf{K})}{m}}\right)$ and even tighter one $O\left(\sqrt{\frac{\sum_{i=1}^{r} \lambda_i}{m}}\right)$.

B. Boosting

1. Implement AdaBoost with boosting stumps and apply the algorithm to the `spambase` dataset

`http://archive.ics.uci.edu/ml/datasets/Spambase`.

Download a shuffled version of that dataset (will be sent by email). Scale the features of all the data. Use the first 3450 examples for training, the last 1151 for testing. The scaling parameters should be computed only on the training data and then applied to the test data.

Consider the binary classification that consists of predicting if the e-mail message is a spam using the 57 features. Randomly split the training data into ten equal-sized disjoint sets. Plot the average cross-validation error plus or minus one standard deviation as a function of the number of rounds of boosting T by selecting the value of this parameter out of $\{10, 10^2, \ldots, 10^k\}$ for a suitable value of k. Let T^* be the best value found for the parameter. Plot the error on the training and test set as a function of the number of rounds of boosting for $t \in [1, T^*]$.

Solution: For the average cross-validation error, it should first decrease and eventually level out after roughly 400 iterations.

The test error should eventually level off, while the training error continues to decrease towards zero.

2. Consider the following variant of the classification problem where, in addition to the positive and negative labels +1 and −1, points may be labeled with 0. This can correspond to cases where the true label of a point is unknown, a situation that often arises in practice, or more generally to the fact that the learning algorithm incurs no loss for predicting −1 or +1 for such a point. Let $\mathcal{X}$ be the input space and let $\mathcal{Y} = \{-1, 0, +1\}$. As in standard binary classification, the loss of $f: \mathcal{X} \to \mathbb{R}$ on a pair $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is defined by $1_{yf(x)<0}$.

Consider a sample $S = ((x_1, y_1), \ldots, (x_m, y_m)) \in (\mathcal{X} \times \mathcal{Y})^m$ and a hypothesis set $\mathcal{H}$ of base functions taking values in $\{-1, 0, +1\}$. For a base hypothesis $h_t \in \mathcal{H}$ and a distribution $\mathcal{D}_t$ over indices $i \in [1, m]$, define ϵ_t^s for $s \in \{-1, 0, +1\}$ by $\epsilon_t^s = \mathbb{E}_{i \sim \mathcal{D}_t}[1_{y_i h_t(x_i) = s}]$.

(a) Derive a boosting-style algorithm for this setting in terms of ϵ_t^ss, using the same objective function as that of AdaBoost. You should carefully justify the definition of the algorithm.

Solution: Say a 'boosting-style algorithm' is just AdaBoost with a possibly different step size α_t. Recall these definitions from the description of AdaBoost: The final hypothesis is $f(x) = \sum_t \alpha_t h_t(x)$ and the normalization constant in round t is $Z_t = \sum_i \mathcal{D}_t(i)\exp(-\alpha_t y_i h_t(x_i))$. We proved in class that

$$\frac{1}{m}\sum_i 1_{y_i f(x_i)<0} \le \frac{1}{m}\sum_i \exp(-y_i f(x_i)) = \prod_t Z_t$$

and that AdaBoost's step size can be derived by minimizing this objective in each round t. Taking that same approach, observe that

$$Z_t = \sum_i \mathcal{D}_t(i)\exp(-\alpha_t y_i h_t(x_i)) = \epsilon_t^0 + \epsilon_t^- \exp(\alpha_t) + \epsilon_t^+ \exp(-\alpha_t).$$

Differentiating the right-hand side with respect to α_t and setting equal to zero shows that Z_t is minimized by letting $\alpha_t = \frac{1}{2}\log\left(\frac{\epsilon_t^+}{\epsilon_t^-}\right)$.

(b) What is the weak learning condition in this setting?

Solution: One possible assumption is $\frac{\epsilon_t^+ - \epsilon_t^-}{\sqrt{1-\epsilon_t^0}} \ge \gamma > 0$. Informally, this assumption says that the difference between the accuracy and error of each weak hypothesis is non-negligible relative to the fraction of examples on which the hypothesis makes any prediction at all. In part (d) we will prove that this assumption suffices to drive the training error to zero.

(c) Write the full pseudocode of the algorithm.

Solution:

1. Given: Training examples $((x_1, y_1), \ldots, (x_m, y_m))$.
2. Initialize $\mathcal{D}_1$ to the uniform distribution on training examples.
3. for $t = 1, \ldots, T$:
   a. $h_t \leftarrow$ base classifier in $\mathcal{H}$ with small error $\epsilon_t^- - \epsilon_t^+$.
   b. $\alpha_t \leftarrow \frac{1}{2}\log\left(\frac{\epsilon_t^+}{\epsilon_t^-}\right)$.
   c. For each $i = 1, \ldots, m$: $\mathcal{D}_{t+1}(i) \leftarrow \frac{\mathcal{D}_t(i)\exp(-\alpha_t y_i h_t(x_i))}{Z_t}$, where $Z_t \leftarrow \sum_i \mathcal{D}_t(i)\exp(-\alpha_t y_i h_t(x_i))$ is the normalization constant.

   i. $f \leftarrow \sum_{t=1}^T \alpha_t h_t$.
4. Return: $\text{sign}(f)$.

(d) Give an upper bound on the training error of the algorithm as a function of the number of rounds of boosting and ϵ_t^ss.

Solution: Plug in the value of α_t from part (a) into $Z_t = \epsilon_t^0 + \epsilon_t^- \exp(\alpha_t) + \epsilon_t^+ \exp(-\alpha_t)$ to obtain $Z_t = \epsilon_t^0 + 2\sqrt{\epsilon_t^- \epsilon_t^+}$. Therefore

$$\frac{1}{m}\sum_i 1_{y_i f(x_i)<0} \le \prod_t Z_t = \prod_t \left(\epsilon_t^0 + 2\sqrt{\epsilon_t^- \epsilon_t^+}\right).$$

Morever, if the weak learning condition from part (b) is satisfied then

$$\begin{aligned}
\epsilon_t^0 + 2\sqrt{\epsilon_t^- \epsilon_t^+} &= \epsilon_t^0 + \sqrt{(1-\epsilon_t^0)^2 - (\epsilon_t^+ - \epsilon_t^-)^2} \\
&= \epsilon_t^0 + (1-\epsilon_t^0)\sqrt{1 - \frac{(\epsilon_t^+ - \epsilon_t^-)^2}{(1-\epsilon_t^0)^2}} \\
&\le \sqrt{1 - \frac{(\epsilon_t^+ - \epsilon_t^-)^2}{1-\epsilon_t^0}} \\
&\le \sqrt{1-\gamma^2}.
\end{aligned}$$

The first equality follows from $(\epsilon_t^+ + \epsilon_t^-)^2 - (\epsilon_t^+ - \epsilon_t^-)^2 = 4\epsilon_t^+\epsilon_t^-$ (just multiply and gather terms) and $\epsilon_t^+ + \epsilon_t^- = 1 - \epsilon_t^0$. The first inequality follows from the fact that square root is concave on $[0, \infty)$, and thus $\lambda\sqrt{x} + (1-\lambda)\sqrt{y} \le \sqrt{\lambda x + (1-\lambda)y}$ for $\lambda \in [0,1]$. The last inequality follows from the weak learning condition.

Therefore we have $\frac{1}{m}\sum_i 1_{y_i f(x_i)<0} \le \left(\sqrt{1-\gamma^2}\right)^T \le \exp\left(-\frac{\gamma^2 T}{2}\right)$, where we used $1 + x \le \exp(x)$.

3. Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{-1, +1\}$ the binary label space. Consider the exponential loss used in AdaBoost: $\ell(h, x, y) = \exp(-yh(x))$. Given a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, the Bayes error for the exponential loss ℓ is defined as the infimum of the errors achieved by measurable functions $h: \mathcal{X} \to \mathbb{R}$:

$$R_\ell^* = \inf_{h:\mathcal{X}\to\mathbb{R} \text{ measurable}} R_\ell(h),$$

where $R_\ell(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(h, x, y)]$. A hypothesis $h_{\exp}$ with $R_\ell(h_{\exp}) = R_\ell^*$ is called a Bayes optimal solution. Define $\eta(x) = \mathbb{P}[y = +1|x]$.

(a) Give the expression of the Bayes optimal solution $h_{\exp}$ for the exponential loss in terms of $\eta(x)$.

Solution: By the definition, $R_\ell(h)$ can be expressed as follows:

$$\begin{aligned}
R_\ell(h) &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\exp(-yh(x))] \\
&= \mathop{\mathbb{E}}_{x}\mathop{\mathbb{E}}_{y|x}[\exp(-yh(x))] \\
&= \mathop{\mathbb{E}}_{x}[\eta(x)\exp(-h(x)) + (1-\eta(x))\exp(h(x))] \\
&\ge \mathop{\mathbb{E}}_{x}\left[2\sqrt{\eta(x)(1-\eta(x))}\right],
\end{aligned}$$

where the equality holds if and only if for any $x \in \mathcal{X}$, we have $h(x) = \frac{1}{2}\log\left(\frac{\eta(x)}{1-\eta(x)}\right)$. Therefore, $R_\ell^* = \mathbb{E}_x\left[2\sqrt{\eta(x)(1-\eta(x))}\right]$ is the Bayes error for the exponential loss and $h_{\exp}: x \mapsto \frac{1}{2}\log\left(\frac{\eta(x)}{1-\eta(x)}\right)$ is the Bayes optimal solution.

(b) Define the generalization error and the Bayes error for the binary classification loss as follows:

$$R(h) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[1_{\text{sign}(h(x))\neq y}\right], \quad R^* = \inf_{h:\mathcal{X}\to\mathbb{R} \text{ measurable}} R(h),$$

where $\text{sign}(t) = 1_{t\ge 0} - 1_{t<0}$. Show that $R(h_{\exp}) = R^*$.

Solution: By the definition, $R(h)$ can be expressed as follows:

$$\begin{aligned}
R(h) &= \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[1_{\text{sign}(h(x))\neq y}\right] \\
&= \mathop{\mathbb{E}}_{x}\mathop{\mathbb{E}}_{y|x}\left[1_{\text{sign}(h(x))\neq y}\right] \\
&= \mathop{\mathbb{E}}_{x}\left[\eta(x)1_{h(x)<0} + (1-\eta(x))1_{h(x)\ge 0}\right] \\
&\ge \mathop{\mathbb{E}}_{x}[\min\{\eta(x), 1-\eta(x)\}],
\end{aligned}$$

where the equality holds if and only if for any $x \in \mathcal{X}$, $\text{sign}(h(x)) = \text{sign}(\eta(x) - 1/2)$. Since for any $x \in \mathcal{X}$, $\text{sign}(h_{\exp}(x)) = \text{sign}(\eta(x) - 1/2)$, we prove that $R(h_{\exp}) = R^*$.