4. Kernel machines

Assigned reading: Chapter 6.4 and Chapter 7 of Prof. Raginsky’s notes, available at http://maxim.ece.illinois.edu/teaching/SWT/ http://maxim.ece.illinois.edu/teaching/fall15b/schedule.html and the supplementary notes provided by the instructor.

Optional supplementary reading: Shalev-Shwartz and Ben-David, Understanding Machine Learning, from Theory to Algorithms, Chapters 14-16.

Problems to be handed in:

1. [Transformation of Mercer kernels]
   Let $A \odot B$ denote Hadamard (i.e. elementwise) multiplication for two vectors or matrices of the same dimension. For example, $(A \odot B)_{ij} = A_{ij}B_{ij}$ for all $ij$.

   (a) Suppose $X$ and $Y$ are two mean zero random vectors with values in $\mathbb{R}^d$. Denote their respective covariance matrices by $\Sigma_X = \mathbb{E}[XX^T]$ and $\Sigma_Y = \mathbb{E}[YY^T]$. Suppose $X$ and $Y$ are independent of each other. Express the covariance matrix of $X \odot Y$ in terms of $\Sigma_X$ and $\Sigma_Y$.

   (b) Show that the product of two Mercer kernels for the same domain is a Mercer kernel. (Hint: A symmetric real matrix is positive semidefinite (PSD) if and only if it is the covariance matrix for some mean zero random vector.)

   (c) Show that if $(K(x,x'))_{x,x' \in X}$ is a Mercer kernel, then $(e^{K(x,x')})_{x,x' \in X}$ is also a Mercer kernel.

2. [Half-space classifiers and support vector machines (SVM)]
   Consider the concept learning problem $(X = \mathbb{R}^d, Y = \{\pm 1\}, \mathcal{P}, \mathcal{G})$ with 0-1 loss, where $\mathcal{P}$ is a set of probability distributions $P$ on $Z = X \times \{\pm 1\}$, and $\mathcal{G}$ consists of all half-space classifiers of the form $g_{w,b}(x) = \text{sgn}(\langle w, x \rangle + b)$, where $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$. The generalization loss is defined by $L_P(w, b) = P\{(Y \neq \text{sgn}(\langle w, Y \rangle + b))\}$.

   (a) Explain why this problem is PAC learnable. That is, describe a PAC learning algorithm and give a performance bound demonstrating PAC learnability. (Hint: The set of classifiers considered is a Dudley class. The bound you give must depend on $d$. Below we find a bound that does not depend on $d$ in the realizable case, under a restriction on the width of the margin.)

   (b) Given $x \in \mathbb{R}^d$ and a classifier $(w, b)$ with $w \neq 0$, let $\pi(x)$ denote the projection of $x$ onto the hyperplane defined by $\langle w, x \rangle + b = 0$. Express $\pi(x)$ and the distance, $\|x - \pi(x)\|$, between $x$ and the hyperplane in terms of $x$, $w$, and $b$. (Hint: Since $w$ is normal to the hyperplane, $\pi(x)$ is the point in the hyperplane of the form $\pi(x) = x - cw$ for some constant $c$.)

   (c) Given a data set $Z^n = ((X_1, Y_1), \ldots, (X_n, Y_n))$ and a classifier $(w, b)$ with $w \neq 0$, let the margin, $M_i$, of the $i^{th}$ sample point be defined by $M_i = Y_i((\langle w, X_i \rangle + b)/\|w\|$. Thus, $M_i$ is the signed distance of $X_i$ from the hyperplane defined by $\langle w, x \rangle + b = 0$, with the sign being positive if $Y_i = \text{sgn}(\langle w, x_i \rangle + b)$ and negative otherwise. Define the margin for the whole data set by $M = \min_{i \in [n]} M_i$. Suppose that $M > 0$ for some choice of $(w, b)$. A key idea of SVMs is to find $(w, b)$ to maximize $M$, with the hope that it will lead to a better classifier for fresh samples. Show that:

   $$\max_{(w,b)} M = \max \left\{ \frac{1}{\|w\|} : (w, b) \text{ subject to } Y_i(\langle w, X_i \rangle + b) \geq 1 \text{ for } i \in [n] \right\}$$  \hspace{1cm} (1)

   (Hint: $M$ for a given $(w, b)$ is not changed if $(w, b)$ is multiplied through by a positive scaler.)

   Remark: The right-hand side of (1) represents an optimization problem that is equivalent to the quadratic optimization problem (2) below.

   (d) (Bound not depending on $d$, realizable case with lower bound on relative margin) Suppose $C_K > 0$ and $\lambda > 0$. Let $\mathcal{P}$ denote the set of all probability distributions $P$ on $Z = X \times \{\pm 1\}$ such
that: \( P\{\sqrt{1+\|X\|^2} \leq C_K \} = 1 \), and there exists a classifier \((w, b)\) (depending on \(P\)) such that \(\|w\|^2 + b^2 \leq \lambda^2\) and \(P\{Y(\langle w, X \rangle + b) \geq 1\} = 1\). These assumptions ensure that iid samples generated by \(P\) satisfy the following with probability one: \(\|X_i\| \leq C_K\) for each \(i\), and there exists \((w, b)\) for the data points with margin \(M\) at least \(1/\lambda\). Thus, the ratio of the margin to \(\max_i\|X_i\|\) is greater than or equal to \(1/\lambda C_K\). Of course, just because the data samples can be separated by a particular hyperplane doesn’t necessarily mean that the hyperplane will classify fresh sample points well. Show that if \((\widehat{w}_n, \widehat{b}_n)\) is the particular ERM classifier given by

\[
(\widehat{w}_n, \widehat{b}_n) = \arg \min \left\{ \|w\|^2 : (w, b) \text{ subject to } Y_i(\langle w, X_i \rangle + b) \geq 1 \text{ for } i \in [n] \right\},
\]

then with probability at least \(1 - \delta\),

\[
L_P((\widehat{w}_n, \widehat{b}_n)) \leq \frac{4\lambda C_K}{\sqrt{n}} + \sqrt{\frac{\log(\frac{1}{\delta})}{2n}}.
\]

The bound (3) does not depend on the dimension, \(d\), of the feature space. (Hint: Bring in a Mercer kernel \(K\), and use the ramp penalty function with unit scale parameter: \(\varphi(x) = \min\{1, (1+x)^+\}\).)

3. **Stochastic gradient descent for solving soft SVM and the kernel trick**

Consider the regularized ERM problem for half-space classifiers and surrogate loss using the hinge penalty function, \(\varphi(x) = (1+x)^+\):

\[
\min_w \left( \frac{\tau \|w\|^2}{2} + \frac{1}{n} \sum_{i=1}^{n} (1 - y_i(\langle w, x_i \rangle)^+) \right)
\]

with respect to \(w \in \mathbb{R}^d\)

where the labeled data is \(((x_1, y_1), \ldots, (x_n, y_n)) \in (\mathbb{R}^d \times \{\pm 1\})^n\). (We could have used \(\langle w, x_i \rangle + b\) instead of \(\langle w, x_i \rangle\) and \(\|w\|^2 + b^2\) instead of \(\|w\|^2\), but by adding a first coordinate equal to 1 to each \(x_i\), the \(b\) can be thought of as the first coordinate of \(w\).) Let \(F(w)\) denote the objective function, defined to be the quantity in parentheses in (4).

(a) Show that a subgradient of the objective function \(F\) at a point \(w \in \mathbb{R}^d\) is give by

\[
\tau w - \frac{1}{n} \sum_{i=1}^{n} y_i x_i 1\{y_i(\langle w, x_i \rangle) < 1\}.
\]

(b) Hence, an iteration of the gradient descent algorithm takes the form

\[
w^{(t+1)} = w^{(t)} + \eta_t \left( -\tau w^{(t)} + \frac{1}{n} \sum_{i=1}^{n} y_i x_i 1\{y_i(\langle w^{(t)}, x_i \rangle) < 1\} \right),
\]

where \(\eta_t\) is a step size multiplier with \(\eta_t > 0\). The idea of stochastic gradient descent (SGD) is to replace the subgradient by a random vector \(v_t\) such that \(E[v_t]\) is a subgradient. A popular way to select such a \(v_t\) is to replace the average over \(n\) terms appearing in (5) by one term selected uniformly at random. Hence, an iteration of the SGD algorithm takes the form

\[
w^{(t+1)} = w^{(t)} + \eta_t \left( -\tau w^{(t)} + y_{i_t} x_{i_t} 1\{y_{i_t}(\langle w^{(t)}, x_{i_t} \rangle) < 1\} \right),
\]

where \(i_t\) is selected uniformly at random from \([n]\). The initial vector could be taken to be \(w^{(0)} = 0\). An intuitive interpretation of (6) is that unless the sample \((x_{i_t}, y_{i_t})\) is correctly labeled and far enough away from the decision boundary, \(w\) is moved a bit in the direction \(y_{i_t} x_{i_t}\) to increase \(y_{i_t} \langle w, x_{i_t} \rangle\). More elaborate versions of SGD have been devised with performance guarantees. They would use a specific choice of stepsizes multipliers (\(\eta_t\) and may return an average of computed values instead of a final value. However, for now, we stick to the basic algorithm here.
The idea of the kernel trick in machine learning is that if there is an algorithm that uses feature vectors only through their inner products, then the algorithm can be implemented in \( \mathbb{R}^n \), where \( n \) is the number of sample points, independently of the dimension of the feature space \( X \) (i.e. \( d \) in this case). To illustrate that, show how the iteration (6), which deals with vectors of length \( d \), can be implemented using vectors of length \( n \).

(Hint: Note that, by induction, \( w^{(t)} \) is always in the span of the \( n \) vectors \( (x_i)_{i \in [n]} \). For the particular kernel \( K(x, x') = \langle x, x' \rangle \), the function \( K_x(\cdot) = \langle \cdot, x \rangle \), so the association between \( K_x \) and \( x \) is particularly simple.)

4. [Convergence properties of Mercer representations]
Suppose \( K = (K(x, x'))_{x, x' \in X} \) is a Mercer kernel for some closed subset \( X \) of \( \mathbb{R}^d \). Suppose \( K \) has a representation of the form \( K(x, x') = \sum_{i=1}^{\infty} \psi_i(x)\psi_i(x') \), where \( \psi_1, \psi_2, \ldots \) are continuous functions on \( X \).

(a) Suppose \( c \in \ell^2 \), so that \( \sum_{i=1}^{\infty} c_i^2 < \infty \). Show that the series \( \sum_{i=1}^{\infty} c_i \psi_i(x) \) converges uniformly on compact (i.e. bounded, closed) subsets of \( X \) (Hint: Dini’s theorem implies that if a sequence of continuous functions converges monotonically to a continuous limit function, then the convergence is uniform on compact subsets. Hence, the series for \( K(x, x) \), \( \sum_{i=1}^{\infty} \psi_i(x)^2 \), converges uniformly on compact subsets of \( X \).)

(b) For convenience, let’s switch to the case of a finite set of basis functions, so \( K(x, x') = \sum_{i=1}^{n} \psi_i(x)\psi_i(x') \), where, as before, the functions \( \psi \) are continuous. Are the functions \( \psi_1, \ldots, \psi_n \) uniquely determined by \( K \)?

5. [Violation of strong linear independence condition]
Find a set of continuous functions \( (\psi_n : n \geq 1) \) on the interval \([0, 1]\) that is linearly independent (i.e. all finite subsets of the functions are linearly independent) but for which there exits a nonzero \( c \in \ell^2 \) such that \( \sum_{i=1}^{\infty} c_i \psi_i(t) = 0 \) for \( t \in [0, 1] \). (Hint: You can use the following fact. The following set of functions is a complete orthonormal basis for the space \( L^2[0, 1] \) of functions \( f \) on \([0, 1]\) such that \( \int_{0}^{1} (f(t))^2 dt < \infty \):

\[
\phi_1(t) = 1; \quad \phi_{2k}(t) = \sqrt{2} \cos (2\pi kt), \quad \phi_{2k+1}(t) = \sqrt{2} \sin (2\pi kt) \quad \text{for } k \geq 1.
\]