COMS 4771: Machine Learning
Homework 4 solutions

Extra credit 1. By the definition of $\boldsymbol{\alpha} = \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}}_\lambda$, since $n\lambda\hat{\boldsymbol{w}}_\lambda = \boldsymbol{X}^\top(\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}}_\lambda)$, we have $n\lambda\hat{\boldsymbol{w}}_\lambda = \boldsymbol{X}^\top\alpha$. Thus, $n\lambda\boldsymbol{X}\hat{\boldsymbol{w}}_\lambda = \boldsymbol{X}\boldsymbol{X}^\top\alpha$, and so $\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top\alpha = \boldsymbol{X}\hat{\boldsymbol{w}}_\lambda$, and hence $\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top\alpha + \alpha = \boldsymbol{X}\hat{\boldsymbol{w}}_\lambda + \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}}_\lambda = \boldsymbol{y}$.

Extra credit 2. We can rewrite the equation obtained in extra credit 1 as $(\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top + \boldsymbol{I})\boldsymbol{\alpha} = \boldsymbol{y}$, and hence $\boldsymbol{\alpha} = (\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top + \boldsymbol{I})^{-1}\boldsymbol{y}$. Then, since $n\lambda\hat{\boldsymbol{w}}_\lambda = \boldsymbol{X}^\top\alpha$, we conclude that

$$\hat{\boldsymbol{w}}_\lambda \;=\; \frac{1}{n\lambda}\boldsymbol{X}^\top\left(\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top + \boldsymbol{I}_n\right)^{-1}\boldsymbol{y} \;=\; \boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}\boldsymbol{y}.$$

In the last equation we use the fact that $\left(\frac{1}{n\lambda}\boldsymbol{X}\boldsymbol{X}^\top + \boldsymbol{I}_n\right)^{-1} = n\lambda(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}$.

Part 1. We need to compute

$$\langle\hat{\boldsymbol{w}}_\lambda, \boldsymbol{\phi}(\boldsymbol{x})\rangle = \hat{\boldsymbol{w}}_\lambda^\top\boldsymbol{\phi}(\boldsymbol{x}) = (\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}\boldsymbol{y})^\top\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{y}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x}).$$

Here, we used the facts that $(\boldsymbol{A}\boldsymbol{B})^\top = \boldsymbol{B}^\top\boldsymbol{A}^\top$, and $(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}$ is symmetric and so it equals its transpose. We now show that we can compute the matrix $(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}$ and the vector $\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$ using the kernel function k, and thus we can compute the prediction $\langle\hat{\boldsymbol{w}}_\lambda, \boldsymbol{\phi}(\boldsymbol{x})\rangle$.

First, we observe that $\boldsymbol{X}\boldsymbol{X}^\top$ is an $n \times n$ matrix whose i,j-th is $\langle\boldsymbol{\phi}(\boldsymbol{x}^{(i)}), \boldsymbol{\phi}(\boldsymbol{x}^{(j)})\rangle = k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$. Hence, the matrix $\boldsymbol{X}\boldsymbol{X}^\top$ can be computed entry by entry using the kernel function. Once $\boldsymbol{X}\boldsymbol{X}^\top$ is computed, $(\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}$ can be computed. Next, we observe that $\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$ is a column vector of size n whose i-th entry is given by $\langle\boldsymbol{\phi}(\boldsymbol{x}^{(i)}), \boldsymbol{\phi}(\boldsymbol{x})\rangle = k(\boldsymbol{x}^{(i)}, \boldsymbol{x})$. Thus $\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$ can be computed entry by entry. Finally, we can compute the prediction by computing the matrix product $\boldsymbol{y}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$. The pseudocode is given below.

1: Input: Training set S, regularization parameter λ, kernel function k, and test point $\boldsymbol{x}$.
2: Compute the matrix $\boldsymbol{X}\boldsymbol{X}^\top$ entry by entry setting the i,j-th entry to be $k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$ for $i, j \in [n] \times [n]$.
3: Compute the matrix $(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda)^{-1}$ numerically.
4: Compute the vector $\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$ entry by entry setting the i-th entry to be $k(\boldsymbol{x}^{(i)}, \boldsymbol{x})$, for $i \in [n]$.
5: Compute the product $\boldsymbol{y}^\top(\boldsymbol{X}\boldsymbol{X}^\top + n\lambda\boldsymbol{I}_n)^{-1}\boldsymbol{X}\boldsymbol{\phi}(\boldsymbol{x})$ and output it.

Extra credit 3. As described in part 1, the matrix $\boldsymbol{X}\boldsymbol{X}^\top$ can be computed entrywise using the kernel function: the i,j-th entry is $k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$. Once the matrix $\boldsymbol{X}\boldsymbol{X}^\top$ is computed, we can compute its eigendecomposition, $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$, where the columns of $\boldsymbol{U}$ are the eigenvectors of $\boldsymbol{X}\boldsymbol{X}^\top$, and $\boldsymbol{\Lambda}$ is a diagonal matrix with the eigenvalues along the diagonal. Now we can construct

the diagonal matrix $\boldsymbol{S}_\ell$ by choosing the top ℓ eigenvalues, taking their square roots, and arranging them on the diagonal of $\boldsymbol{S}_\ell$. The matrix $\boldsymbol{U}_\ell$ is the formed column-wise by choosing the corresponding ℓ eigenvectors.

Part 2. $\boldsymbol{S}_\ell$ is an $\ell \times \ell$ matrix and $\boldsymbol{U}_\ell$ is an $n \times \ell$ matrix. As in part 1, we can can compute $\boldsymbol{X}\boldsymbol{X}^\top$ entrywise using the kernel function: the i, j-th entry is $k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$. We already have the matrices $\boldsymbol{S}_\ell$ and $\boldsymbol{U}_\ell$, and then the rank ℓ PCA representation of the training set is given by the columns of $\boldsymbol{S}_\ell^{-1}\boldsymbol{U}_\ell^\top \boldsymbol{X}\boldsymbol{X}^\top$.