Kernelizing Algorithms

by Alvin Wan . alvinwan.com/cs189

This document summarizes approaches to kernelizing algorithms. Original sources, across course materials and notes, are cited. In general, we can perform the following steps, where w is our model.

- **Consider the original objective function.**
- **Plug in** $w = X^T\alpha + w_0$, where X is an $n \times d$ matrix and $Xw_0 = 0$. This decomposition always exists, by the fundamental theorem of linear algebra.
- **Convert the objective** into an optimization over α. You should find that $w_0 = 0$ yields the optimal α.
- **Solve for the closed-form solution** by taking the derivative and setting it equal to 0.

Contents

1 Kernelizing Least Squares

1.1 Ridge Regression Derivation

The following derivation can be found in Homework 1.

Take our original objective function, apply the fundamental theorem of linear algebra, and replace XX^T with K, where X is $n \times n$.

$$\min_w \|Xw - y\|_2^2 + \lambda \|w\|_2^2$$

We first define $w = X^T\alpha + w_0$, where $Xw_0 = 0$. In the third step, note that $w_0^T X^T = (Xw_0)^T = 0$ and that $\|w_0\|_2^2$ is minimized when $w_0 = 0$. Thus, we take $w_0 = 0$, and continue computing our optimal model.

$$\begin{aligned}
&\text{minimize}_w \|Xw - y\|_2^2 + \lambda \|w\|_2^2 \\
&= \text{minimize}_w \|X(X^T\alpha + w_0) - y\|_2^2 + \lambda \|X^T\alpha + w_0\|_2^2 \\
&= \text{minimize}_w \|XX^T\alpha - y\|_2^2 + \lambda(\|X^T\alpha\|_2^2 + \|w_0\|_2^2 + 2w_0^T X^T \alpha) \\
&= \text{minimize}_\alpha \|XX^T\alpha - y\|_2^2 + \lambda \|X^T\alpha\|_2^2
\end{aligned}$$

Take the gradient and set equal to 0 for our optimal model.

$$\begin{aligned}
2XX^T(XX^T\alpha - y) + 2\lambda XX^T\alpha &= 0 \\
(XX^T + \lambda I)XX^T\alpha &= XX^T y \\
\alpha &= (XX^T + \lambda I)^{-1} y \\
\alpha &= (K + \lambda I)^{-1} y
\end{aligned}$$

Plug back in to get w^*.

$$w^* = X^T\alpha^*$$

To predict, using w^*, we use $x^T w^*$.

$$\hat{y} = x_i^T w^* = x^T X^T \alpha^* = \sum_i k(x, x_i)\alpha_i^*$$

1.2 Kernelized Ridge Regression Algorithm

- Compute kernel matrix K.
- Compute $\alpha^* = (K + \lambda I)^{-1} y$.
- To predict x, $\hat{y} = \sum_i k(x, x_i)\alpha^*$.

2 Kernelizing K-Means

2.1 Kernelized K-means Derivation I

The following derivation can be found in the Final Review slides.

Take the original objective function, expand, and replace all dot products with the kernel function $k(\cdot, \cdot)$.

$$\text{minimize}_{C_k} \sum_{\{x_i \in C_k\}} \|x_i - \mu_k\|_2^2$$

We simply expand and replace. Assume there are K clusters and N x_i.

$$\begin{aligned}
&\text{minimize}_{C_k} \sum_{k=1}^{K} \sum_{\{i \in C_k\}} \|x_i - \mu_k\|_2^2 \\
&= \text{minimize}_{C_k} \sum_{k=1}^{K} \sum_{\{i \in C_k\}} \|x_i\|_2^2 + \|\mu_k\|_2^2 - 2x_i^T \mu_k \\
&= \text{minimize}_{C_k} \sum_{k=1}^{K} \sum_{\{i \in C_k\}} k(x_i, x_i) + k(\mu_k, \mu_k) - 2(x_i, \mu_k) \\
&= \text{minimize}_{C_k} k(\mu_k, \mu_k) + \sum_{i=1}^{N} k(x_i, x_i) - 2\sum_{k=1}^{K} \sum_{\{i \in C_k\}} 2(x_i, \mu_k)
\end{aligned}$$

2.2 Kernelized Alternating Minimization Derivation II

2.2.1 Minimizing Over μ_k

Take our original objective function, apply the fundamental theorem of linear algebra, and replace XX^T with K_i.

$$\text{minimize}_{C_k} \sum_{\{i \in C_k\}} \|x_i - \mu_k\|_2^2$$

Plug in $\mu_k = X^T\alpha + \mu_0$, where $X\mu_0 = 0$. In the third step below, the cross-term goes to zero, because $\mu_0^T x_i$ and $\mu_0^T X^T = (X\mu_0)^T = 0$. We also note that $\|\mu_0\|_2^2$ is minimized when μ_0 is 0, so we set $\mu_0 = 0$ and continue with our computation for the optimal clusters.

$$\begin{aligned}
&\sum_{\{i \in C_k\}} \|x_i - \mu_k\|_2^2 \\
&= \sum_{\{i \in C_k\}} \|x_i - X^T\alpha - \mu_0\|_2^2 \\
&= \sum_{\{i \in C_k\}} \|x_i - X^T\alpha\|_2^2 + \|\mu_0\|_2^2 - 2\mu_0^T(x_i - X^T\alpha) \\
&= \sum_{\{i \in C_k\}} \|x_i - X^T\alpha\|_2^2
\end{aligned}$$

Now, take the gradient and set equal to 0. In the third step, note that

$$Xx_i = [k(x_i, x_1), k(x_i, x_2), \cdots k(x_i, x_n)]^T$$

, so $Xx_i = k_i$, where $K = XX^T$. For the fourth step, note that $K^{-1}k_i$ is 1 when dotting the ith row of K^{-1} with k_i and 0 elsewhere.

$$\sum_{\{i \in C_k\}} -2X(x_i - X^T\alpha) = 0$$

$$\sum_{\{i \in C_k\}} Xx_i = |C_k| XX^T\alpha$$

$$\alpha^* = \frac{1}{|C_k|} \sum_{\{i \in C_k\}} (XX^T)^{-1} Xx_i$$

$$\alpha^* = \frac{1}{|C_k|} \sum_{\{i \in C_k\}} K^{-1} k_i$$

$$\alpha^* = \frac{1}{|C_k|} \sum_{\{i \in C_k\}} e_i$$

Plugging back in,

$$\mu_k^* = X^T\alpha^* = X^T \frac{1}{|C_k|} \sum_{\{i \in C_k\}} e_i = \frac{1}{|C_k|} \sum_{\{i \in C_k\}} x_i$$

2.2.2 Minimizing Over x_i

For each x_i, assign the cluster that minimizes the following quantity:

$$\begin{aligned}
&\text{minimize}_k \|x_i - X^T\alpha_k\|_2^2 \\
&= \text{minimize}_k \|x_i\|_2^2 + \|X^T\alpha_k\|_2^2 - 2x_i^T X^T \alpha_k \\
&= \text{minimize}_k \|x_i\|_2^2 + \alpha_k^T XX^T \alpha_k - 2(Xx_i)^T \alpha_k \\
&= \text{minimize}_k \|x_i\|_2^2 + \alpha_k K \alpha_k - 2K_i^T \alpha_k
\end{aligned}$$

2.3 Kernelized K-means Algorithm

We effectively run Lloyd's algorithm.

- Initialize cluster means μ_k.
- Compute kernel matrix K.
- Compute the new cluster index for each sample, by taking $\text{minimize}_k \|x_i - \mu_k\|_2^2$.
- Update the cluster centers.
- Repeat until convergence.

3 Kernelizing PCA

See Stephen's notes for the derivation. The following algorithm is taken straight from his notes.

- Compute kernel matrix K.
- Take the first p eigenvectors of K.
- Compute $\phi(z)_i = k(x_i, z)^T \alpha_i$.