CS 4780/5780 Homework 8 Solution

Problem 1: Regularization Mitigates Overfitting

In this question, we are going to investigate how adding l2 regularization can help mitigate the effect of overfitting for ordinary least square regression. First, recall that in our notes for lecture 10, we mention that we can rewrite the objective function of l2-regularized least square regression (or ridge regression)

$$\min_{\vec{w}} \sum_{i=1}^{n} (\vec{w}^T \vec{x}_i - y_i)^2 + \lambda ||\vec{w}||_2^2$$

as

$$\min_{\vec{w}} \sum_{i=1}^{n} (\vec{w}^T \vec{x}_i - y_i)^2 \text{ subject to } ||\vec{w}||_2^2 \leq B^2$$

To simplify our analysis, we are going to focus on the second expression. In addition, we are going to assume the following:

(i) Each data point $(\vec{x}_i, y_i)$ is drawn identically and independently from the distribution $\mathcal{P}$, namely, the dataset $\mathcal{D} \sim \mathcal{P}^n$

(ii) For any $(\vec{x}, y)$ sampled from $\mathcal{P}$, we have $||\vec{x}||_2^2 = 1$

With the above assumption, we are going to do the following:

(a) Notice that $\vec{w}(\mathcal{D})$ is a function of $\mathcal{D}$ and since $\mathcal{D}$ is random, so is $\vec{w}(\mathcal{D})$. Define $\bar{w} = \mathbb{E}_{\mathcal{D}}(\vec{w}(\mathcal{D}))$. Show that

$$||\vec{w}(\mathcal{D}) - \bar{w}||_2^2 \leq 4B^2$$

using the triangular inequality

$$||a - b||_2 \leq ||a||_2 + ||b||_2$$

(b) Define the model $h_{\mathcal{D}}(\vec{x}) = \vec{w}(\mathcal{D})^T \vec{x}$ and $\bar{h}(\vec{x}) = \mathbb{E}_{\mathcal{D}}(h_{\mathcal{D}}(\vec{x}))$. Show that the variance of the model

$$\mathbb{E}_{\vec{x},\mathcal{D}}((h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2) \leq 4B^2$$

by first showing that

$$h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}) = (w(\mathcal{D}) - \bar{w})^T \vec{x}$$

and then using the Cauchy-Schwarz inequality:

$$(a^T b)^2 \leq (a^T a)(b^T b)$$

to conclude the result.

Takeaway: By adding regularization, we essentially bound the variance of the model which reduces overfitting.

Solution

(a) Using the triangular inequality we have,

$$||\vec{w}(\mathcal{D}) - \bar{w}||_2 \leq ||\vec{w}(\mathcal{D})||_2 + ||\bar{w}||_2$$

Take square of each side.

$$||\vec{w}(\mathcal{D}) - \bar{w}||_2^2 \leq ||\vec{w}(\mathcal{D})||_2^2 + ||\bar{w}||_2^2 + 2||\vec{w}(\mathcal{D})||_2||\bar{w}||_2$$

Since $\bar{w} = \mathbb{E}_{\mathcal{D}}(\vec{w}(\mathcal{D}))$, we have

$$||\bar{w}||_2^2 \leq B^2$$

$$||\vec{w}(\mathcal{D}) - \bar{w}||_2^2 \leq ||\vec{w}(\mathcal{D})||_2^2 + ||\bar{w}||_2^2 + 2||\vec{w}(\mathcal{D})||_2||\bar{w}||_2 \leq B^2 + B^2 + 2B^2 = 4B^2$$

(b) Since $h_{\mathcal{D}}(\vec{x}) = \vec{w}(\mathcal{D})^T\vec{x}$, then

$$h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}) = \vec{w}(\mathcal{D})^T\vec{x} - \mathbb{E}_{\mathcal{D}}(\vec{w}(\mathcal{D})^T\vec{x})$$

Because $\bar{w} = \mathbb{E}_{\mathcal{D}}(\vec{w}(\mathcal{D}))$ and the expectation of $\vec{w}(\mathcal{D})$ does not depend on $\vec{x}$ we have

$$h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}) = \vec{w}(\mathcal{D})^T\vec{x} - \bar{w}^T\vec{x}$$

By the Cauchy-Schwarz inequality,

$$(h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2 \leq ((\vec{w}(\mathcal{D}) - \bar{w})^T(\vec{w}(\mathcal{D}) - \bar{w}))(\vec{x}^T\vec{x})$$

This can be written as,

$$(h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2 \leq ||\vec{w}(\mathcal{D}) - \bar{w}||_2^2 \cdot ||\vec{x}||_2^2$$

Because $||\vec{x}||_2^2 = 1$, we have,

$$(h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2 \leq ||\vec{w}(\mathcal{D}) - \bar{w}||_2^2$$

Using our result from 1a, we get,

$$(h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2 \leq 4B^2$$

Finally, taking the expectation we get,

$$\mathbb{E}_{\vec{x},\mathcal{D}}((h_{\mathcal{D}}(\vec{x}) - \bar{h}(\vec{x}))^2) \leq 4B^2$$

Problem 2: Kernelized Perceptron

In this problem, we are going to kernelize the perceptron algorithm. Recall the perceptron algorithm

Algorithm 1: Perceptron Algorithm

```
Initialize $\vec{w} = \vec{0}$ ;
while TRUE do
    m = 0 ;
    for $(x_i, y_i) \in D$ do
        if $y_i(\vec{w}^T \vec{x_i}) \leq 0$ then
            $\vec{w} \leftarrow \vec{w} + y_i \vec{x_i}$;
            $m \leftarrow m + 1$;
        end
    end
    if $m = 0$ then
        break
    end
end
```

Now recall that in homework 2, we have shown that if we know the number of misclassifications for each training point, say α_i for $\vec{x}_i$, then we deduce that

$$\vec{w} = \sum_{i=1}^{n} \alpha_i y_i \vec{x}_i$$

This observation allows us to modify the perceptron algorithm such that we only need to keep track the number of misclassifications for each training points, instead of updating $\vec{w}$.

(a) Fill in the skeleton code so that the perceptron algorithm only needs to keep track of the number of misclassifications for each training point, instead of updating $\vec{w}$

Algorithm 2: Modified Perceptron Algorithm

```
Initialize $\vec{\alpha} = \vec{0}$ ;
while TRUE do
    m = 0 ;
    for $(x_i, y_i) \in D$ do
        if $y_i \sum_{j=1}^{n} \alpha_j y_j \vec{x_j}^T \vec{x_i} \leq 0$ then
            $\alpha_i \leftarrow \alpha_i + 1$;
            $m \leftarrow m + 1$;
        end
    end
    if $m = 0$ then
        break
    end
end
```

(b) Now, how would you modify algorithm 2 to kernelize the perceptron algorithm?

Change $y_i \sum_{j=1}^{n} \alpha_j y_j \vec{x_j}^T \vec{x_i}$ to $y_i \sum_{j=1}^{n} \alpha_j y_j K(x_j, x_i)$, where K is the kernel function.

Problem 3: Constructing Kernels

In class, we have shown how we could use a few rules to construct new kernels from existing valid kernels. In this problem, we will prove that these rules indeed produce valid kernels.
Recall that there are two ways to show that a function $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a valid kernel function:

1. The matrix
$$K_{ij} = k(\vec{x}_i, \vec{x}_j)$$
is symmetric and positive semidefinite for any set of vectors $\vec{x}_1, ..., \vec{x}_n \in \mathbb{R}^d$
2. $k(\vec{x}_i, \vec{x}_j) = \phi(\vec{x}_i)^T \phi(\vec{x}_j)$ for some transformation ϕ

Suppose k_1, k_2 are valid kernel functions. Show that the following kernels are valid:

(a) $k(\vec{x}_1, \vec{x}_2) = ck_1(\vec{x}_1, \vec{x}_2)$ for any $c \geq 0$.

(b) $k(\vec{x}_1, \vec{x}_2) = k_1(\vec{x}_1, \vec{x}_2) + k_2(\vec{x}_1, \vec{x}_2)$

(c) $k(\vec{x}_1, \vec{x}_2) = k_1(\vec{x}_1, \vec{x}_2)k_2(\vec{x}_1, \vec{x}_2)$.

Solution: Suppose ϕ_1 and ϕ_2 are the transformation associated with k_1 and k_2 repsectively.

(a) Notice that $k(\vec{x}_i, \vec{x}_j) = ck_1(\vec{x}_i, \vec{x}_j) = c\phi_1(\vec{x}_i)^T\phi_1(\vec{x}_j) = (\sqrt{c}\phi_1(\vec{x}_i))^T(\sqrt{c}\phi_1(\vec{x}_j))$. We can take $\phi_4(\vec{x}_i) = \sqrt{c}\phi_1(\vec{x}_i)$ as a transformation for $ck_1(\vec{x}_i, \vec{x}_j)$

(b) Observe that $k(\vec{x}_i, \vec{x}_j) = k_1(\vec{x}_i, \vec{x}_j) + k_2(\vec{x}_i, \vec{x}_j) = \phi_1(\vec{x}_i)^T\phi_1(\vec{x}_j) + \phi_2(\vec{x}_i)^T\phi_2(\vec{x}_j) = \begin{bmatrix}\phi_1(\vec{x}_i)\\ \phi_2(\vec{x}_i)\end{bmatrix}^T \begin{bmatrix}\phi_1(\vec{x}_i)\\ \phi_2(\vec{x}_i)\end{bmatrix}$.
We can take $\phi_5(\vec{x}_i) = \begin{bmatrix}\phi_1(\vec{x}_i)\\ \phi_2(\vec{x}_i)\end{bmatrix}$ as a transformation for $k_1(\vec{x}_1, \vec{x}_2) + k_2(\vec{x}_1, \vec{x}_2)$

(c) Notice that
$$\begin{aligned} k(\vec{x}_i, \vec{x}_j) &= k_1(\vec{x}_i, \vec{x}_j)k_2(\vec{x}_i, \vec{x}_j) \\ &= \phi_1(\vec{x}_i)^T\phi_1(\vec{x}_j)\phi_2(\vec{x}_i)^T\phi_2(\vec{x}_j) \\ &= \sum_{a=1}^{n_1}[\phi_1(\vec{x}_i)]_a[\phi_1(\vec{x}_j)]_a \sum_{b=1}^{n_2}[\phi_2(\vec{x}_i)]_b[\phi_2(\vec{x}_j)]_b \\ &= \sum_{a=1}^{n_1}\sum_{b=1}^{n_2}[\phi_1(\vec{x}_i)]_a[\phi_2(\vec{x}_i)]_b[\phi_1(\vec{x}_j)]_a[\phi_2(\vec{x}_j)]_b \end{aligned}$$

Suppose $\phi_6(\vec{x}_i) = [[\phi_1(\vec{x}_i)]_1[\phi_2(\vec{x}_i)]_1, ..., [\phi_1(\vec{x}_i)]_1[\phi_2(\vec{x}_i)]_{n_2}, [\phi_1(\vec{x}_i)]_2[\phi_1(\vec{x}_1)]_1, ..., [\phi_1(\vec{x}_1)]_{n_1}[\phi_1(\vec{x}_1)]_{n_2}]^T$. Then,
$$\phi_6(\vec{x}_i)^T\phi_6(\vec{x}_j) = \sum_{a=1}^{n_1}\sum_{b=1}^{n_2}[\phi_1(\vec{x}_i)]_a[\phi_2(\vec{x}_j)]_b[\phi_1(\vec{x}_i)]_a[\phi_2(\vec{x}_j)]_b = k_1(\vec{x}_i, \vec{x}_j)k_2(\vec{x}_i, \vec{x}_j)$$