STAT 542: Homework 3

Ahmadreza Eslaminia (ae15)

Please make sure that your solutions are readable and the file size is reasonable. Typing the answers is highly encouraged.

Problem 1.

[3pts] Consider a regression problem with data $(x_i, y_i)_{i=1}^n$, where each x_i is one-dimensional. Consider a feature map $\phi(x) = \left(1, \frac{x}{2}, \frac{x^2}{4}, \ldots, \frac{x^{p-1}}{2^{p-1}}\right)$.

- Find an explicit expression for the kernel function $k(x, x') = \langle \phi(x), \phi(x') \rangle$, where $\langle , \rangle is$ the inner product in $\mathbb{R}^p$.
- Suppose that we want to estimate the regression function by solving

$$\hat{\theta} := \underset{\theta \in \mathbb{R}^p}{\arg\min} \left\{ \sum_{i=1}^{n} \left| \langle \phi(x_i), \theta \rangle - y_i \right|^2 + \lambda \|\theta\|_2^2 \right\} \tag{1}$$

where $\lambda > 0$, and setting $\hat{f}(x) = \langle \phi(x), \hat{\theta} \rangle$. Find an expression of $\hat{f}$ using only $K = [k(x_i, x_j)]_{1 \le i,j \le n}$ (and not using ϕ directly). Note that the resulting algorithm is more computationally efficient than directly solving (1) when n is much smaller than p.

- If $\lambda \le 0$, is the expression of $\hat{f}$ you found in the previous question still equivalent to (1)?

Solution

Part1

Given the feature map

$$\phi(x) = \left(1, \frac{x}{2}, \frac{x^2}{4}, \ldots, \frac{x^{p-1}}{2^{p-1}}\right),$$

the associated kernel function $k(x, x')$ is obtained by computing the dot product of $\phi(x)$ and $\phi(x')$:

$$k(x, x') = \langle \phi(x), \phi(x') \rangle .$$

Expanding the dot product, we have:

$$k(x, x') = 1 \cdot 1 + \frac{x}{2} \cdot \frac{x'}{2} + \frac{x^2}{4} \cdot \frac{x'^2}{4} + \ldots + \frac{x^{p-1}}{2^{p-1}} \cdot \frac{x'^{p-1}}{2^{p-1}}.$$

Simplification leads to:

$$k(x, x') = 1 + \frac{xx'}{2^2} + \frac{x^2 x'^2}{2^4} + \ldots + \frac{x^{p-1} x'^{p-1}}{2^{2(p-1)}}.$$

This is recognized as a geometric progression with the common ratio $\frac{xx'}{4}$ for $xx' \neq 4$. The sum of a geometric progression is given by $\frac{1-r^n}{1-r}$, where r is the ratio and n is the number of terms. Thus, we express the kernel function as:

$$k(x, x') = \frac{1 - \left(\frac{xx'}{4}\right)^p}{1 - \frac{xx'}{4}}.$$

However, if $xx' = 4$, the progression does not converge, and the kernel function is merely the sum of p terms, each of which equals 1, yielding $k(x, x') = p$.

Part2

Given the optimization problem to estimate the regression function:

$$\hat{\theta} = \arg\min_{\theta \in \mathbb{R}^p} \left\{ \sum_{i=1}^{n} \left(\phi(x_i)^\top \theta - y_i \right)^2 + \lambda \|\theta\|^2 \right\},$$

where $\lambda > 0$ and the regression function is defined as $f(x) = \phi(x)^\top \hat{\theta}$, we seek to find an expression for f using only the kernel matrix K where $K_{ij} = k(x_i, x_j)$ without directly using $\phi(x)$.

1. Represent θ as a linear combination of the feature mappings of the training data:
$$\theta = \Phi^\top \alpha,$$
where Φ is the matrix of feature vectors $\phi(x_i)$ and α is the vector of coefficients.

2. we have :
$$\phi(x_i) \cdot \theta = k(x_i, x')^\top \alpha$$

3. Substitute θ in the objective function with its dual representation:
$$L(\alpha) = (\Phi^\top \theta - y)^\top (\Phi^\top \theta - y) + \lambda \alpha^\top \Phi \Phi^\top \alpha.$$

4. : Using the kernel matrix $K = \Phi\Phi^\top$, we can simplify the objective function to:
$$L(\alpha) = (\alpha^\top K - y^\top)(K\alpha - y) + \lambda \alpha^\top K \alpha.$$

5. Find the gradient of $L(\alpha)$ with respect to α and set it to zero to find the optimal α:
$$\nabla_\alpha L(\alpha) = 2K(K\alpha - y) + 2\lambda K\alpha = 0.$$

6. *Solve for* α: Simplify and solve the resulting equation:
$$K^2\alpha + \lambda K\alpha = Ky,$$
$$(K + \lambda I)\alpha = y,$$
This linear system can be solved for α assuming $K+\lambda I$ is invertible, which typically requires K to be positive definite and $\lambda > 0$.

7. *Estimate of* θ: Calculate θ using the estimated α:
$$\hat{\theta} = \Phi^\top \alpha.$$

8. *Prediction Function* $f(x)$: Express $f(x)$ for new inputs using the kernel function:
$$f(x) = \phi(x)^\top \hat{\theta} = \sum_{i=1}^{n} \alpha_i k(x, x_i).$$

This method uses kernel matrix K to implicitly include the feature mapping $\phi(x)$, avoiding explicit computation of potentially high-dimensional feature vectors. This approach is computationally efficient, especially when the number of features p is much larger than the number of samples n.

Part3

Given the condition $\lambda \leq 0$, we are asked to consider the validity of the expression for $\hat{f}$ found in the previous question, which is contingent upon the value of the regularization parameter λ.

- When $\lambda > 0$, the matrix $K + \lambda I$ is positive definite and invertible if K is at least positive semi-definite. This guarantees a unique solution for α in the equation $(K + \lambda I)\alpha = y$.
- If $\lambda = 0$, we are essentially solving an ordinary least squares problem, which does not guarantee a unique solution if K is not full rank.
- For $\lambda < 0$, the regularization term becomes negative, which can reward complexity rather than penalize it, leading to a potentially non-invertible matrix $K + \lambda I$, and a non-unique or nonexistent solution for α.

Therefore, the expression for $\hat{f}$ relies on the assumption that $\lambda > 0$. If $\lambda \leq 0$, this assumption is violated, and the expression for $\hat{f}$ would not be applicable in the same manner, as it could lead to overfitting or a non-unique solution. We conclude:

If $\lambda \leq 0$, the expression for $\hat{f}$ is not equivalent to the form in equation (1).

Problem 2.

[2pts] Consider 3-NN in the setting of P6 in s8_knn.pdf. Show that as $n \to \infty$, the asymptotic error of $3-\text{NN}$ is upper bounded by

$$P_{3nn} \leq p(1-p)\left(4p - 4p^2 + 1\right) \tag{2}$$

where p is the Bayes probability of error. Show that the bound can be weakened to $P_{3nn} \leq 1.4p$, and compare it with the case of $1-\text{NN}$.

Hint: Notes_knn.pdf contains the main ingredients of the analysis.

Solution

Part1

In our analysis, we denote by p the probability $P(\hat{Y}_{3NN} = 1)$, which represents the probability that the 3-NN classifier predicts a data point as belonging to class '1'. We assume that $p < 0.5$.

The misclassification probability for the 3-NN classifier diverging from the true outcome Y_0 is computed as follows:

$$\begin{aligned} P(\hat{Y}_{3NN} \neq Y_0) = &\binom{3}{2} P(\hat{Y}_{3NN} = 1)^2 P(Y_0 = 0) + \binom{3}{2} P(\hat{Y}_{3NN} = 0)^2 P(Y_0 = 1) \\ &+ P(\hat{Y}_{3NN} = 0)^3 P(Y_0 = 1) + P(\hat{Y}_{3NN} = 1)^3 P(Y_0 = 0). \end{aligned}$$

Simplification by utilizing binomial probabilities and combining like terms yields:

$$\begin{aligned} P(\hat{Y}_{3NN} \neq Y_0) &= 3p^2(1-p)^2 + p^3(1-p) + (1-p)^3 p \\ &= 3p^2 - 2p^3 + p^4 + 3p^2 - 6p^3 + 3p^4 + p - 3p^2 + 3p^3 - p^4 \\ &= p(6p^2 - 8p^3 + 4p^4 + 1) \\ &= p(1-p)(4p - 4p^2 + 1), \end{aligned}$$

which is the established upper bound for P_{3nn}.

part 2

We aim to show that the inequality

$$(1-p)(4p - 4p^2 + 1) \leq 1.4$$

holds for $p \in [0, 0.5]$.

To demonstrate this, we proceed with the following steps:

1. Compute the derivative of the function $f(p) = (1-p)(4p - 4p^2 + 1)$ with respect to p:

$$f'(p) = \frac{d}{dp}[(1-p)(4p - 4p^2 + 1)].$$

2. Find the critical points by solving $f'(p) = 0$.

3. Evaluate $f(p)$ at the critical points within the interval $[0, 0.5]$ as well as at the endpoints $p = 0$ and $p = 0.5$:

$$f(0) = (1 - 0)(4 \cdot 0 - 4 \cdot 0^2 + 1), = 1$$

$$f(0.5) = (1 - 0.5)(4 \cdot 0.5 - 4 \cdot 0.5^2 + 1) = 1$$

4. The maximum value of $f(p)$ within this interval is then compared to 1.4 to establish the inequality.

After computation, we find that the maximum value of $f(p)$ in the interval [0, 0.5] is approximately 1.3156, which satisfies the inequality:

$$f(p) \leq 1.4 \text{ for all } p \in [0, 0.5],$$

thereby proving the statement.

part 3

We have established that the error probability for the 3-Nearest Neighbors classifier is bounded by

$$P_{3nn} \leq 1.4p.$$

In comparison, for 1-Nearest Neighbor from the lecture notes, the error probability is at most twice the Bayes error, giving us

$$P_{1nn} \leq 2p(1 - p) \leq 2p.$$

Comparing the coefficients in the bounds of P_{3nn} and P_{1nn}, we observe that the upper bound for 3-NN, $1.4p$, is less than the bound for 1-NN, $2p(1 - p)$, considering the range of p.

Problem 3.

[2pts] Kernel functions can be defined over objects as diverse as graphs, sets, strings, and text documents. Consider, for instance, a fixed set and define a nonvectorial space consisting of all possible subsets of this set. If A_1 and A_2 are two such subsets then one simple choice of kernel would be

$$k\left(A_1, A_2\right) = 2^{|A_1 \cap A_2|} \tag{3}$$

where $A_1 \cap A_2$ denotes the intersection of sets A_1 and A_2, and $|A|$ denotes the number of elements in A.

- Show that this is a valid kernel function, by constructing a feature map ϕ such that $k\left(A_1, A_2\right) = \langle \phi\left(A_1\right), \phi\left(A_2\right) \rangle$.

- If $k(A_1, A_2) = 4^{|A_1 \cap A_2|}$ instead, construct a ϕ such that $k(A_1, A_2) = \langle \phi(A_1), \phi(A_2) \rangle$.
- [1 additional bonus point] Construction for $k(A_1, A_2) = a^{|A_1 \cap A_2|}$, where $a > 1$ is arbitrary?

Hint: Part 1) is taken from Ex 6.12 in Bishop's book. The constructions are not unique, but a convenient method for part 2), 3) is to use part 1) and then use the construction in P21 s7_svm.pdf for a polynomial transform of a kernel.

Solution

Part 1

Let us consider a finite set Ω and its power set $\mathcal{P}(\Omega)$, which consists of every conceivable subset of Ω. Define a mapping ϕ that projects a subset B within Ω to a vector space such that each dimension corresponds to a particular subset within $\mathcal{P}(\Omega)$. Specifically, the mapping is given by:

$$\phi(B) = (\phi_V(B))_{V \in \mathcal{P}(\Omega)}$$

Here, each component $\phi_V(B)$ of $\phi(B)$ is defined as follows:

$$\phi_V(B) = \begin{cases} 1 & \text{if } V \subseteq B \\ 0 & \text{otherwise} \end{cases}$$

For any two subsets $A1, A2 \subseteq \Omega$, we define the kernel function $k(A1, A2)$ to be $2^{|A1 \cap A2|}$. This expression effectively enumerates the subsets common to both $A1$ and $A2$, weighting them with a factor of 2.

To validate that k represents an inner product within the devised vector space, one needs to show congruence between $k(A1, A2)$ and the dot product $\langle \phi(A1), \phi(A2) \rangle$.

Calculating the dot product of $\phi(A1)$ with $\phi(A2)$ yields a sum of element-wise products:

$$\langle \phi(A1), \phi(A2) \rangle = \sum_{V \in \mathcal{P}(\Omega)} \phi_V(A1) \cdot \phi_V(A2)$$

The value of each product is unity when V is a subset common to $A1$ and $A2$, and zero otherwise. The contributing subsets are exclusively those found in the intersection $A1 \cap A2$. Since each element in the intersection affords a binary choice — to include it in a subset or not — the count of subsets is $2^{|A1 \cap A2|}$.

Consequently, the following expression is derived:

$$\langle \phi(A1), \phi(A2) \rangle = \sum_{V \subseteq A1 \cap A2} 1 = 2^{|A1 \cap A2|}$$

This result confirms the kernel $k(A1, A2)$ is indeed tantamount to the dot product of $\phi(A1)$ and $\phi(A2)$ in the constructed feature space, verifying the legitimacy of k as a kernel function.

Part 2

Starting from the kernel function established in Problem a, we denote this kernel as $k(A1, A2)$. We can form a polynomial kernel based on this existing kernel as follows:

$$k_{poly}(A1, A2) = (c + k(A1, A2))^d$$

For our particular case, we select the constants $c = 0$ and $d = 2$. This yields a transformed kernel:

$$k_{poly}(A1, A2) = (k(A1, A2))^2$$

Given that in Problem a, the kernel is defined as $k(A1, A2) = 2^{|A1 \cap A2|}$, substituting this into our polynomial kernel transformation gives us:

$$k_{poly}(A1, A2) = \left(2^{|A1 \cap A2|}\right)^2$$

Simplifying the right-hand side, where the exponentiation of an exponentiation leads to the multiplication of exponents, we obtain the desired kernel function for Problem b:

$$k_{poly}(A1, A2) = 4^{|A1 \cap A2|}$$

This confirms that by applying a polynomial transform with specific choices for c and d, we can derive the kernel function required in Problem b directly from the kernel in Problem a.

Part 3

We extend the approach used in Problem b to create a new kernel function in Problem c. We use the kernel function from Problem a, $k(A1, A2) = 2^{|A1 \cap A2|}$, and apply a polynomial transformation with the degree d set to $\log_2(a)$:

$$k_{new}(A1, A2) = (c + k(A1, A2))^d$$

Setting $c = 0$ and $d = \log_2(a)$, the new kernel becomes:

$$k_{new}(A1, A2) = \left(2^{|A1 \cap A2|}\right)^{\log_2(a)}$$

Using the property of exponents that $b^{x \cdot y} = (b^x)^y$, we rewrite the above expression as:

$$k_{new}(A1, A2) = a^{|A1 \cap A2|}$$

Thus, by selecting d as the logarithm of a to the base 2, we obtain the kernel function specified for Problem c using the polynomial transformation of the kernel from Problem a. This process showcases the flexibility of kernel methods in machine learning, where transformations can adapt the kernel to various types of data relationships.