Prof. Yiping Lu
IEMS 402 Statistical Learning
November 4, 2024

Homework 2: Bias and Variance Trade-off

Question 1. (**Adaptive to Anisotropic Smoothness**) Consider the data $(X_1, Y_1), \ldots, (X_n, Y_n)$ where $X_i \in \mathbb{R}$ and $Y_i \in \mathbb{R}$. Inspired by the fact that $\mathbb{E}[Y|X = x] = \int yp(x, y)dy/p(x)$, define

$$\hat{m}(x) = \frac{\int y\hat{p}(x, y)dy}{\hat{p}(x)}$$

where

$$\hat{p}(x) = \frac{1}{n}\sum_i \frac{1}{h}K\left(\frac{X_i - x}{h}\right)$$

and

$$\hat{p}(x, y) = \frac{1}{n}\sum_i \frac{1}{h^2}K\left(\frac{X_i - x}{h}\right)K\left(\frac{Y_i - y}{h}\right).$$

Assume that $\int K(u)du = 1$ and $\int uK(u)du = 0$.

- Show that $\hat{m}(x)$ is exactly the kernel regression estimator that we defined in class.

Remark. If we understand the estimator as estimating $p(x)$ then calculate the expectation, the plug-in bound would lead to $n^{-\frac{s}{d+2s}}$. However if one consider the problem as estimating $p(x, y)$ then caculate the mean of y, the plug-in bound would lead to $n^{-\frac{s}{d+1+2s}}$. One reason is $p(x, y)$ is very smooth in y but is only C^s in x-direction. This question showed that kernel smoothing may adapt to the aniostropic smoothness. Becareful about this problem in your furture research. Reference : https://arxiv.org/abs/1910.12799

Question 2. (**Implicit Bias of Overparameterized Linear Regression**) Given n pairs of input data with d features and scalar label $(\mathbf{x}_i, t_i) \in \mathbb{R}^d \times \mathbb{R}$, we wish to find a linear model $f(\mathbf{x}) = \hat{\mathbf{w}}^\top \mathbf{x}$ with $\hat{\mathbf{w}} \in \mathbb{R}^d$ that minimizes the squared error of prediction on the training samples defined below. This is known as an empirical risk minimizer. For concise notation, denote the data matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ and the corresponding label vector $\mathbf{t} \in \mathbb{R}^n$. The training objective is to minimize the following loss:

$$\min_{\hat{\mathbf{w}}} \frac{1}{n}\sum_{i=1}^n (\hat{\mathbf{w}}^\top \mathbf{x}_i - t_i)^2 = \min_{\hat{\mathbf{w}}} \frac{1}{n}\|\mathbf{X}\hat{\mathbf{w}} - \mathbf{t}\|_2^2.$$

We assume $\mathbf{X}$ is full rank: $\mathbf{X}^\top\mathbf{X}$ is invertible when $n > d$, and $\mathbf{X}\mathbf{X}^\top$ is invertible otherwise. Note that when $d > n$, the problem is underdetermined, i.e., there are fewer training samples than parameters to be learned. This is analogous to learning an overparameterized model, which is common when training deep neural networks.

- (Underdetermined case) For the underdetermined case, what is the closed-form expression of the optimal linear predictor $\hat{w}$.
- (Overparameterized case) The training objective is to minimize the squared loss:

$$L(\mathbf{w}) = \frac{1}{2n}\sum_{i=1}^n (\mathbf{w}^\top \mathbf{x}_i - t_i)^2 = \frac{1}{2n}\|\mathbf{X}\mathbf{w} - \mathbf{t}\|_2^2,$$

where $\mathbf{X} \in \mathbb{R}^{n\times d}$ is the data matrix with rows $\mathbf{x}_i^\top$, and $\mathbf{t} \in \mathbb{R}^n$ is the vector of targets. In gradient flow, we consider continuous-time dynamics (an approximation to the gradient descent) where the weight vector $\mathbf{w}(t)$ evolves according to the negative gradient of the loss:

$$\frac{d\mathbf{w}(t)}{dt} = -\nabla L(\mathbf{w}(t)) = -\frac{1}{n}\mathbf{X}^\top(\mathbf{X}\mathbf{w}(t) - \mathbf{t}).$$

Assuming that the gradient flow starts from the origin, i.e., $\mathbf{w}(0) = \mathbf{0}$, we can solve for $\mathbf{w}(t)$ by integrating this differential equation.

The solution to this differential equation can be expressed as:

$$\mathbf{w}(t) = \mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top)^+\mathbf{t}\left(1 - e^{-\frac{t}{n}}\right),$$

where $(\mathbf{X}\mathbf{X}^\top)^+$ denotes the Moore-Penrose pseudoinverse of $\mathbf{X}\mathbf{X}^\top$.
  - As $t \to \infty$, where does the solution $\mathbf{w}(t)$ converges $\mathbf{w}^* = \lim_{t\to\infty} \mathbf{w}(t)$ to?
  - Show that the vector $\mathbf{w}^*$ is the minimum norm solution to the interpolation problem $\mathbf{X}\mathbf{w} = \mathbf{t}$, meaning it satisfies the interpolation conditions $\mathbf{X}\mathbf{w}^* = \mathbf{t}$ and has the smallest ℓ_2-norm among all such solutions.
    (*hint*: The minimum norm interpolation problem
    $$\min \|w\| \text{ s.t. } Xw = t$$
    has a closed form solution, which can be obtained using the lagrangian duality.)

  This happens because starting from the origin and following gradient flow imposes an implicit bias towards minimizing the norm of $\mathbf{w}$ while satisfying the interpolation constraint. This is why gradient flow training converges to the **minimum norm interpolator**.
- What are some potential benefits of the minimum-norm solution? *(Hint: reminds the Support Vector Machine example in homework 1)*

Question 3. (**Benefit of Overparameterization**, *Coding*) Visualize and compare under-parameterized with over-parameterized polynomial regression: https://colab.research.google.com/drive/1NCSHp-gkh1kGh Include your code snippets for the fit_poly function in the write-up. Does overparameterization (higher degree poly- nomial) always lead to overfitting, i.e. larger test error?