GRAPH LAPLACIAN AND LINEAR SMOOTHER

1. Overview

Let $X_i \in \mathbb{R}^p, i = 1, \ldots, n$, and $X_i \sim_d P$ with density $p(x)$. Let the observations be x_i. Choose kernel function K and bandwidth $h > 0$ Let $S = \left(K\left(\frac{x_i - x_j}{h}\right)\right)_{i,j}$ be the similarity matrix.

Let D be the degree matrix for S, a diagonal matrix with row sums of S. Let L_{rw} be the random walk graph Laplacian defined as $I - D^{-1}W$. In this form, when we are thinking about Kernel Smoothing, then $D^{-1}S$ is actually the smoothing matrix. Apply L_{rw} to the smoothing function estimates the bias of the smoothing, and hence, deducting the estimated bias from the regression estimations, we can get a new "debiased" set of estimates. Let $\hat{Y} = (D^{-1}S)\mathbb{Y}$ be the estimates after kernel smoothing

$$\hat{Y} - L_{rw}\hat{Y} = (I - (I - D^{-1}S))\hat{Y} = (D^{-1}S)(D^{-1}S)\mathbb{Y}$$

2. Simulation Experiments

Draw independent variables X from 2-dimensional Normal distribution

$$X \sim Normal(\mu, \Sigma), \mu = (0, 0), \Sigma = \begin{pmatrix} 2 & 1 \\ 1 & 4 \end{pmatrix}$$

Then for the response, assume it follows the regression function:

$$Y_{true} = m(X) = m((a, b)) = a^2 \cos(a) + b - b^2 + ab$$

The observed responses are $Y = Y_{true} + \epsilon$ where ϵ are random noises drawn from $Normal(0, 2)$.

In this experiments, the sample size for X is $n = 1000$, and Gaussian kernel is used.

2.1. **Assessing Bias.**

To assess the bias of the smoothers, the generated X are fixed, while noises are generated 1000 times independently, and hence 1000 sets of observations are generated with the setting. The smoothing procedure is carried on each set of observations, generating 1000 sets of predictions. The mean of the 1000 predictions are used to analyze the bias.

2.2. **Semi-supervised Setting.**

To test the idea in Kernel Regression setting, m unlabeled points are drawn from the same distribution as X.

Date: April 18, 2021.

3. Kernel Smoothing

Let $L = D^{-1}S$ be the smoothing matrix, then the smoothed observations are $\hat{Y} = L\mathbb{Y}$. Let $\hat{Y}_2 = LL\mathbb{Y}$ be the result after applying the square of the original matrix to the observations. Let L^2_{elem} be the element-wise squaring of matrix L, and let $\hat{Y}_{el2} = L^2_{elem}\mathbb{Y}$ be the result after applying the element-wise squared smoothing matrix.

The results of the simulation experiments are:

Squaring the matrix, no matter using the correct matrix power or element-wise squaring, the performance in terms of MSE is not better than the original smoothing. This is compared based on the optimal bandwidth for each procedure. The optimal bandwidth for the squared matrix LL is smaller than the original smoothing matrix L, and the optimal bandwidth for L^2_{elem} is really small.

However, noticeable is that element-wise squaring the matrix greatly smoothes most of the values to 0.

In terms of bias, LL doesn't make much improvement (there is improvement in simulation1, but not the other two).The L^2_{elem} reduces the bias in simulation2 but not the others. The smoothing bias-reducing property of element-wise squaring may come from the small bandwidth used.

4. Kernel Regression

Let there be n labeled samples $X_1, \ldots, X_n$ with corresponding labels $Y_1, \ldots, Y_n$. Let there be m unlabeled samples $A_1, \ldots, A_m$ drawn from the same distribution as X_i's.

In this setting, let S be the $m \times n$ matrix with $S_{ij} = K\big(\frac{a_i - x_j}{h}\big)$. Let D be the $m \times m$ diagonal matrix with row sums of S as the entries. Hence define the $m \times n$ matrix $L = D^{-1}S$ and Kernel Regression Estimation gives $\hat{Y}_a = L\mathbb{Y}$ where $\mathbb{Y}$ is the column vector of the observed labels.

Since the smoothing matrix now is not necessarily a square matrix, we cannot directly square such a matrix, hence a 2-step procedure is used:

1. First do the regular Kernel regression estimation and get the m predicted labels $\hat{Y}_a = L\mathbb{Y}$ where $\mathbb{Y}$.
2. Treats the $\hat{Y}_a$'s as given labels and append them to the existing labels to get $\bar{\mathbb{Y}}$. Construct the $m \times (n+m)$ matrix $\bar{S}$ such that $S_{ij} = K\big(\frac{a_i - x_j}{h}\big)$ for $j \leq n$ and $S_{ij} = K\big(\frac{a_i - a_{j-n}}{h}\big)$ for $j > n$. Let $\bar{D}$ be the degree matrix fro $\bar{S}$. Calculate the predictions as $\hat{Y}_a^{'} = \bar{D}^{-1}\bar{S}\bar{\mathbb{Y}}$.

The second step can be thought of as getting the Kernel smoothing matrix with the $(n+m)\times(n+m)$ matrix between all the points from X and A combined, apply to the combined labels $\bar{\mathbb{Y}}$, and extract the last m terms. The final predictions are obtained after two rounds of kernel smoothing in this procedure, so it is implicitly squaring the smoothing matrix. However, in the second step, all the pairwise information between the independent variables are used.

However, for a matrix that is not square matrix, we can still directly do the element-wise squaring. Again let L^2_{elem} be the element-wise square of the kernel regression matrix L.

The results of the simulation experiments are:

The two-step procedure, using the same bandwidth as the original smoothing matrix, performs better in terms of MSE. However, in terms of bias, the two-step procedure works better with the smaller bandwidth (the optimal bandwidth in terms of MSE found in the smoothing setting).

The element-wise squared smoothing matrix performs really poor regarding MSE and the biases. The nice graph from last time is unfortunately by some mistake...

4.1. **Using two-step procedure to smooth the known labels.** Thinking of the second step as smoothing for the observed Y values, the performance of the two-step procedure as smoothing is better in terms of bias, especially when using the smaller bandwidth.

In the given setting, without additional manipulation, we have the smoothed known labels as

$$\hat{Y}_i = \frac{\sum_{j=1}^n K\big(||X_j - X_i||/h\big)Y_j}{\sum_{j=1}^n K\big(||X_j - X_i||/h\big)} for \quad i = 1, \dots, n$$

and the predictions for the unlabeled data as

$$\hat{Y}_{a_l} = \frac{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)Y_j}{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)} for \quad l = 1, \dots, m$$

.

Then with the two-step procedure, the smoothed knwon labels become:

$$\begin{aligned}\hat{Y}_i^{'} &= \frac{\sum_{j=1}^n K\big(||X_j - X_i||/h\big)Y_j + \sum_{l=1}^m K\big(||a_l - X_i||/h\big)\hat{Y}_{a_l}}{\sum_{j=1}^n K\big(||X_j - X_i||/h\big) + \sum_{l=1}^m K\big(||a_l - X_i||/h\big)} \\ &= \frac{\sum_{j=1}^n K\big(||X_j - X_i||/h\big)Y_j + \sum_{l=1}^m K\big(||a_l - X_i||/h\big)\frac{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)Y_j}{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)}}{\sum_{j=1}^n K\big(||X_j - X_i||/h\big) + \sum_{l=1}^m K\big(||a_l - X_i||/h\big)}\end{aligned}$$

and the predictions for the unlabeled data are

$$\begin{aligned}\hat{Y}_{a_k}^{'} &= \frac{\sum_{j=1}^n K\big(||X_j - a_k||/h\big)Y_j + \sum_{l=1}^m K\big(||a_l - a_k||/h\big)\hat{Y}_{a_l}}{\sum_{j=1}^n K\big(||X_j - a_k||/h\big) + \sum_{l=1}^m K\big(||a_l - a_k||/h\big)} \\ &= \frac{\sum_{j=1}^n K\big(||X_j - a_k||/h\big)Y_j + \sum_{l=1}^m K\big(||a_l - a_k||/h\big)\frac{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)Y_j}{\sum_{j=1}^n K\big(||X_j - a_l||/h\big)}}{\sum_{j=1}^n K\big(||X_j - a_k||/h\big) + \sum_{l=1}^m K\big(||a_l - a_k||/h\big)}\end{aligned}$$

which are re-weighted results of the known labels and the 1-step predictions. **Does such re-weighting has similar effect as renormalization that the density of X is better controlled for?**