A Study of Density Estimation under Moment Constraints

Joshua Yang

Learning Objectives and Abstract This project explores the Constrained Density Estimation method introduced by Hall and Presnell [2]. The study has two primary objectives: first, to understand the derivation of the method and improve the optimization technique; second, to perform simulations of kernel density estimation with moment constraints.
The code for the project is available at `https://github.com/SZ-yang/BIOST527_ConstrainedKDE-`.

1 Introduction

Recall from the course, we have learned that the kernel density estimation, which is defined as $\hat{f}_n(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)$, where h is the bandwidth and K is a symmetric kernel function. While KDE offers many desirable properties, certain applications require the estimator to maintain specific characteristics such as the mean, median, or interquartile range of the original data. Consequently, researchers developed methods for density estimation under constraints. In the paper *Density Estimation Under Constraints*, Hall and Presnell designed a weighted bootstrap method, where the weights are selected by minimizing the distance from the uniform bootstrap distribution while satisfying the imposed constraints [2]. Although the authors mentioned other constraints like entropy and quantiles, we will mainly focus on moment constraints for this project due to limited time.

2 Methodology

2.1 Bootstrap and Kernel Density Estimation

Assume that the basic density estimator $\tilde{f}$, computed from a random sample $\mathcal{X} = \{x_1, \ldots, x_n\}$, is a bootstrap estimator in the sense that it may be written as

$$\tilde{f}(x) = E\{\phi(x_1^*, \ldots, x_n^*; x)|\mathcal{X}\}, \tag{1}$$

where ϕ is a known function and $\{x_1^*, \ldots, x_n^*\}$ is a resample drawn independently and uniformly from the empirical distribution determined by $\mathcal{X}$.

One example of an estimator of the form (1) is the kernel density estimator, where $\phi(x_1, \ldots, x_n; x) = \sum_i L(x_i, x)$ and the function L depends on n and incorporates the smoothing parameter h (bandwidth). In this case, the prescription at (1) produces $\tilde{f}(x) = \sum_i L(x_i, x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x-x_i}{h}\right)$.

2.2 Biased-Bootstrap

Given a vector $p = (p_1, \ldots, p_n)$ of probabilities for a multinomial distribution on $X_1, \ldots, X_n$, define $\{X_1^\dagger, \ldots, X_n^\dagger\}$ to be a resample drawn by sampling from $\mathcal{X}$ using the weighted bootstrap with these probabilities. The n resampling operations are independent, and in each of them X_i is drawn with probability p_i.

Now, define the biased-bootstrap of $\tilde{f}$ as:

$$\tilde{f}(x|p) = E\{\phi(X_1^\dagger, \ldots, X_n^\dagger; x)|\mathcal{X}\}.$$

If $p = p_{\text{unif}} = (n^{-1}, \ldots, n^{-1})$ we recover the basic estimator $\tilde{f}$, defined at (1).

Suppose that there are r constraints, of the form $T_j(f) = t_j$ for $1 \le j \le r$, where each T_j is a functional operating on the population density f, and $t_1, \ldots, t_r$ are constants. The biased-bootstrap form of the constraints is

$$T_j\{\tilde{f}(\cdot|p)\} = t_j \quad \text{for} \quad 1 \le j \le r. \tag{2}$$

Let $\hat{p} = (\hat{p}_1, \ldots, \hat{p}_n)$ denote the value of p that minimizes the distance from p to p_{unif} subject to $\sum_i p_i = 1$ and the constraints (2). Then, our constrained estimator of $f(x)$ is $\hat{f}(x) = \tilde{f}(x|\hat{p})$.

Now, we have successfully pivoted the problem of constrained kernel density estimation to a problem of finding an optimal p for the biased bootstrap (a kind of weighted bootstrap with p satisfying certain constraints)[1][2].

2.3 Estimators and Constraints Linear in p_i's

If the estimator is linear in the p_i's, then we have:

$$\tilde{f}(x|p) = \sum_{i=1}^{n} p_i K_i(x)$$

where K_i is a deterministic function of x_i. For instance, in this project we use kernel density estimation, $\tilde{f}(x|p)$ can be further expressed as:

$$\tilde{f}(x|p) = \sum_{i=1}^{n} p_i \frac{1}{h} K(\frac{x - x_i}{h}). \tag{3}$$

For the constraint function T_j that is also linear in the p_i's, we then would have:

$$\sum_{i=1}^{n} p_i T_j(K_i) = t_j, \tag{4}$$

where $j \in \{1, \ldots, r\}$, total r constraints. Additionally, we define that $T_o(K_i) = 1$ and $t_0 = 1$, to ensure that the sum of the weights p_i equals to 1, i.e., $\sum_{i=1}^{n} p_i = 1$.

If the constraints only contain moments, i.e., we are asking that the jth moment of the kernel density estimator $\tilde{f}(x|p)$ equals the jth sample moment, then t_j in (4) can take the form: $t_j = n^{-1} \sum_{i=1}^{n} x_i^j$. Since constraints involving only moments are linear, then for any linear estimator, we can write

$$T_j(K_i) = \int_{\mathcal{S}} y^j K_i(y)\, dy,$$

where $\mathcal{S}$ is the support of the sampling distribution. Since we are using kernel density estimator with symmetric kernel K, then we can futher write it into

$$T_j(K_i) = \sum_{k=0}^{\lfloor j/2 \rfloor} \binom{j}{2k} X_i^{j-2k} h^{2k} \kappa_{2k}, \tag{5}$$

where $\lfloor j/2 \rfloor$ is the integer part of $j/2$, and $\kappa_l = \int y^l K(y)\, dy$.

To sum up, if we are using kernel density estimation with moment constraints, (4) is equivalent to

$$\sum_{i=1}^{n} p_i T_j(K_i) = t_j \quad \Leftrightarrow \quad \sum_{i=1}^{n} p_i \sum_{k=0}^{\lfloor j/2 \rfloor} \binom{j}{2k} X_i^{j-2k} h^{2k} \kappa_{2k} = n^{-1} \sum_{i=1}^{n} x_i^j.$$

2.4 Optimization Problem

Then the above question now becomes an optimization problem:

$$\hat{p} = \underset{p}{\operatorname{argmin}} D(p_{unif}, p) \tag{6}$$

subject to

$$(V)_j \coloneqq \sum_{i=1}^{n} p_i T_j(K_i) - t_j = 0 \qquad \forall j \in \{0, \ldots, r\} \tag{7}$$

Where D is the distance between p and p_{unif}, and $V \in \mathbb{R}^{r+1}$ is the vector of our constraints. In this project, we use the **Kullback-Leiber Divergence** as a measure of the distance between p_{unif} and p, according to the definition, it is calculated as:

$$D_{KL}(p_{unif||p}) = -\sum_{i=1}^{n} \log(np_i).$$

We then can apply **Lagrange multiplier** to solve this optimization problem by defining the Lagrangian as:

$$\begin{aligned}\mathcal{L}(p, \lambda) &= D(p_{unif}, p) + \lambda \cdot V \\ &= D(p_{unif}, p) + \sum_{j=0}^{r} \lambda_j \{\sum_{i=1}^{n} p_i T_j(K_i) - t_j\}\end{aligned}$$

where λ_j is the Lagrange multiplier of the corresponding constraint j.
Set $\nabla\mathcal{L} = 0$, the $\hat{p}$ can be calculated as:

$$\begin{aligned}\frac{\partial}{\partial p_i}\left(D_{KL}(p_{unif}, p) + \sum_{j=0}^{r} \lambda_j \{\sum_{i=1}^{n} p_i T_j(K_i) - t_j\}\right) &= 0 \\ -\frac{1}{np_i} n + \sum_{j=0}^{r} \lambda_j T_j(K_i) &= 0 \\ \hat{p}_i(\lambda) &= \{\sum_{j=0}^{r} \lambda_j T_j(K_i)\}^{-1}\end{aligned}$$

The next step is to find the Lagrange multiplier λ that satisfies the constraints. According to the paper, the author proposed solving this problem using the Newton-Raphson method. However, through multiple implementations, I found that the function might not converge with this method. Therefore, I used the **Gauss-Newton method** with line search, which yielded better results in the simulation experiments. The method can be broken down as follows:
The Gauss–Newton method iteratively finds the value of λ that minimize the sum of squares:

$$||V(\lambda)||^2 \leq tolerance,$$

recalling $V(\lambda) \in \mathbb{R}^{r+1}$ is the vector of our constraints described in (7).
For each iteration, we update λ using

$$\lambda^{new} = \lambda - \alpha\Delta\lambda,$$

where α is the step size which can be determined using a line search method, and $\Delta\lambda$ can be solved using:

$$J^T J\Delta\lambda = -J^T V,$$

and $J \in \mathbb{R}^{(r+1)\times(r+1)}$ is the Jacobian matrix of the constraints. The element in entry (j, k) of the Jacobian matrix is calculated as:

$$\begin{aligned}(J)_{jk} = \frac{\partial(V)_j}{\partial\lambda_k} &= \frac{\partial}{\partial\lambda_k}\sum_{i=1}^{n} p_i T_j(K_i) - t_j \\ &= \frac{\partial}{\partial\lambda_k}\sum_{i=1}^{n}\{\sum_{j=0}^{r} \lambda_j T_j(K_i)\}^{-1} T_j(K_i) \\ &= -\sum_{i} p_i(\lambda)^2 T_j(K_i) T_k(K_i).\end{aligned}$$

Finally, after we have the $\hat{p} = (\hat{p}_1, \ldots, \hat{p}_n)$ that solves the optimization problem as described in (6,7), we can plug it into the equation (3) the compute our kernel density estimation under moment constraints.

3 Simulation

In this section, we implemented the methods discussed to validate their practical utility and compare the differences between kernel density estimations under various moment constraints. The results demonstrate the effectiveness of the constrained estimation approach.

3.1 Kernel and Test Setup

For our simulation, we used the biweight kernel $K(u) = \frac{15}{16}(1-u^2)^2$ to estimate an asymmetric bimodal normal mixture $\frac{1}{2}N(0,1) + \frac{1}{2}N\left(\frac{3}{2}, \frac{1}{9}\right)$. We conducted four tests, each with a different number of moment constraints: No constraints; First two moments constrained; first three moments constrained; First four moments constrained.

Each test used a sample size of $n = 100$, and the bandwidth was set to $h = 0.6$ across all experiments.

3.2 Simulation Results

Table 1 below shows the first four moments of the estimated density under different constraints compared to the sample moments. One can observe that our method performed really well in satisfying the imposed constraints. Each constrained method matched the sample moments exactly, from the lowest moment to the highest moment imposed in each test.

Additionally, the table presents the Mean Square Error (MSE) for each model.The results demonstrate that adding moment constraints systematically improves the accuracy of the density estimates, as reflected by the decreasing MSE values.

Figure (1,3,5,7) shows the estimated density and true density under different constraints and Figure (2,4,6,8) shows the weights p_i assigned to each data point under different constraints. The weights assigned to each data point also illustrate the impact of moment constraints. With no constraints, weights are relatively uniform. As more moment constraints are imposed, the weights become more varied to satisfy the constraints.

	First Moment	Second Moment	Third Moment	Fourth Moment	MSE
No Constraint	0.64	1.688	1.764	4.72	0.0027
Two Constraints	0.64	1.628	1.73	4.438	0.0021
Three Constraints	0.64	1.628	1.649	4.399	0.0014
Four Constraints	0.64	1.628	1.649	4.126	0.0008
Sample	0.64	1.628	1.649	4.126	NA

Table 1: Comparison of Moments and MSE with Different Constraints

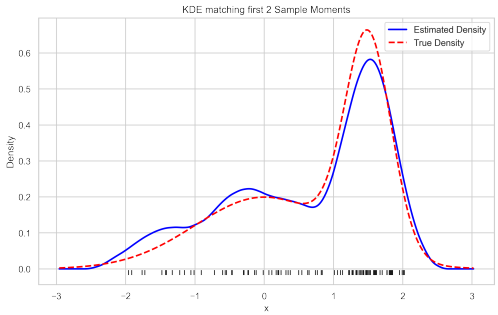

Figure 1: No Constraint

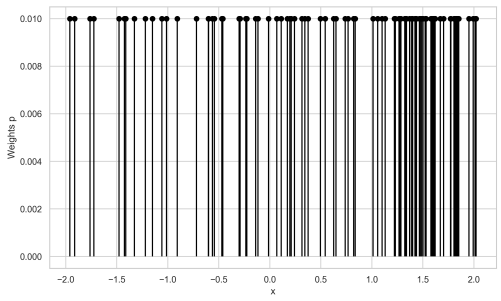

Figure 2: Weight No Constraint

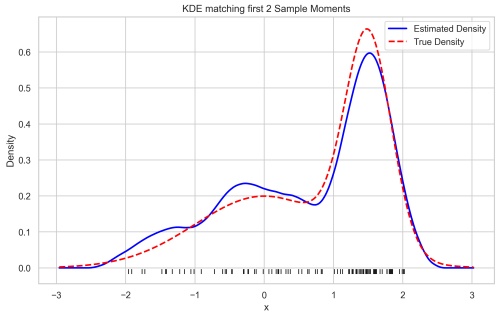

Figure 3: Two Moment

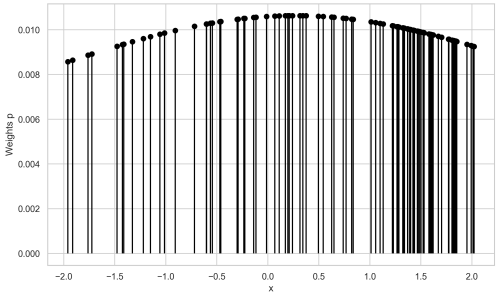

Figure 4: Weight Two Moment

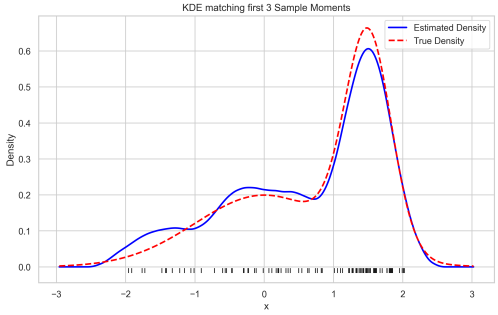

Figure 5: Three Moment

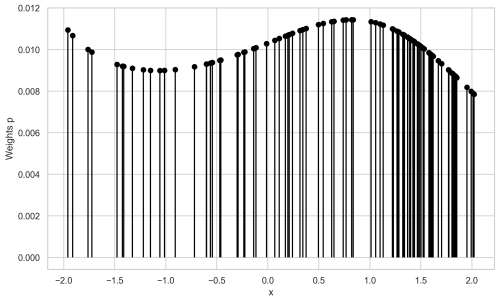

Figure 6: Weight Three Moment

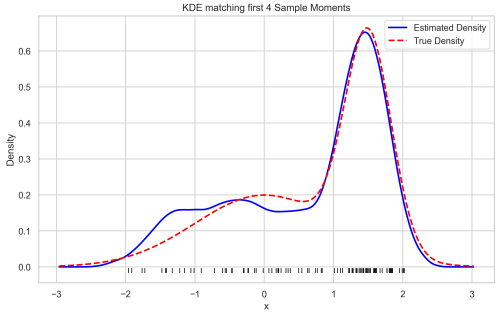

Figure 7: Four Moment

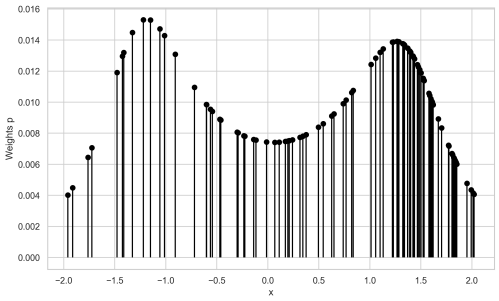

Figure 8: Weight Four Moment

4 Discussion

In this project, I've successfully meet the learning objectives to understand Hall's method in the constrained density estimation, and improved his algorithm by adopting Gauss-Newton method. On the other hand, the simulation results validate the theoretical benefits of constrained kernel density estimation. By incorporating moment constraints, the estimated densities more closely match the underlying distribution, which is beneficial for applications requiring precise density estimates that maintain specific sample characteristics.

Acknowledgement

I would like to thank Professor Eardi Lila and TA Ethan Ancell for their dedication throughout this quarter. I wish I could have attended more lectures in person if it weren't for my injuries. I look forward to taking more of your courses in the future!
R.I.P. Prof. Peter Hall

References

[1] HALL, P., DICICCIO, T. J., AND ROMANO, J. P. On Smoothing and the Bootstrap. *The Annals of Statistics 17*, 2 (1989), 692 – 704.

[2] HALL, P., AND PRESNELL, B. Density estimation under constraints. *Journal of Computational and Graphical Statistics 8*, 2 (1999), 259–277.