Quality and Efficiency in
Kernel Density Estimates for Large Data

THE UNIVERSITY OF UTAH

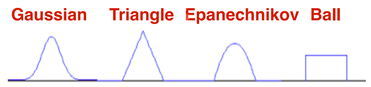
Yan Zheng, Jeffrey Jestes, Jeff M. Phillips, Feifei Li

Kernel Density Estimates

Definition

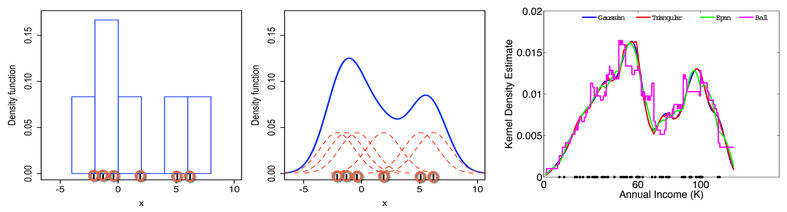
For $x \in \mathbb{R}^d$ $\text{KDE}_P(x) = \frac{1}{|P|}\sum_{p\in P} K(p,x)$

Examples of Kernels in R^2



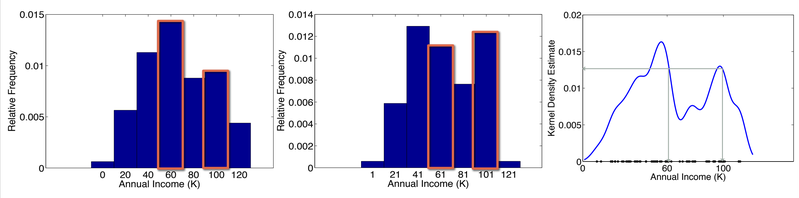
- Gaussian: $K(p,x) = \frac{1}{2\pi\sigma^2}\exp\left(\frac{-\|p-x\|^2}{2\sigma^2}\right)$
- Triangle: $K(p,x) = \frac{3}{\pi\sigma^2}\max\left\{0, 1-\frac{\|p-x\|}{\sigma}\right\}$
- Epanechnikov: $K(p,x) = \frac{2}{\pi\sigma^2}\max\left\{0, 1-\frac{\|p-x\|^2}{\sigma^2}\right\}$
- Ball: $K(p,x) = \begin{cases} 1/\pi\sigma^2 & \text{if } \|p-x\| < \sigma \\ 0 & \text{otherwise} \end{cases}$

Histogram and Kernel Density Estimates



Drawback of Histogram

Query value change significantly across boundary of bins
The choice of origin can have quite an effect



Approximate Kernel Density Estimate

Given the input P, σ, and some error parameter $\varepsilon > 0$, the goal is to produce a small set Q to ensure

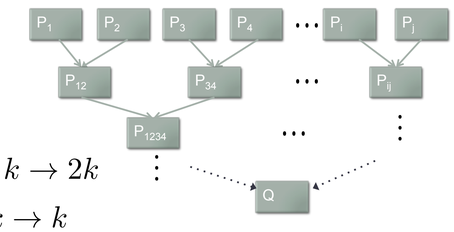
$$\max_{x\in\mathbb{R}^d} |\text{KDE}_P(x) - \text{KDE}_Q(x)| = \|\text{KDE}_P - \text{KDE}_Q\|_\infty \leq \varepsilon,$$

and we call this an *ε-approximation*

Algorithm

MergeReduce framework

- Initialization phase: Decompose P into $P_1, P_2, \ldots, P_k$
- Combination phase:
  - Merge Step $k+k \to 2k$
  - Reduce Step $2k \to k$
  - Proceeds in $\log(n/k)$ rounds
- Useful for distributed and streaming versions



Matching

Min-cost matching

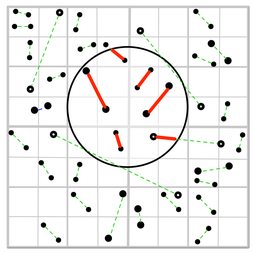
Running time $O(\frac{n}{\varepsilon^2}\log^2 n \log\frac{1}{\varepsilon})$

Sample size $|Q| = O(\frac{1}{\varepsilon}\log n\sqrt{\log\frac{1}{\varepsilon}})$

Grid matching

Running time $O(n\log\frac{1}{\varepsilon})$

Sample size $|Q| = O(\frac{1}{\varepsilon}\log n\log^{1.5}\frac{1}{\varepsilon})$

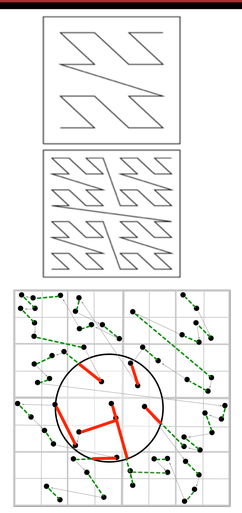


Grid Matching Algorithm

- Starting with i = 0, construct G_i
- Inside of each cell, match points arbitrarily
- Only the unmatched points survive to the next round
- Each cell in G_{i+1} is the union of 4 cells from G_i
- After $\log\frac{1}{\sigma\epsilon} + 1$ rounds, match point arbitrarily

Z-Order Selection

- Zrandom
  Compute the Z-order of all points, and of every two points discard one at random
  Repeat this discarding of half the points until the remaining set is sufficiently small
- Zorder
  Select one point from each set of $|P|/k$ points in sorted order and using the ϵ-approximate quantiles algorithm



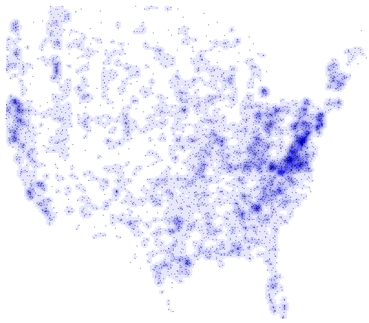
Experiment

General Setup

Data sets: OpenStreetMap
160million records in 6.6GB

Test points:
4000 randomly from P
1000 from the domain of P

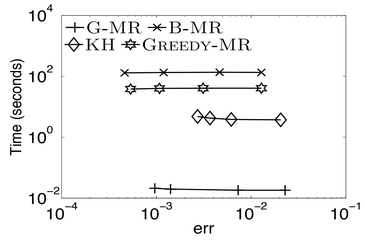
Default setting:
10 million records
10 random trials
$\delta = 0.001$ $\sigma = 200$



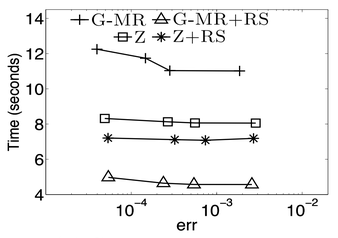
Centralized Experiments

- Baseline Methods:
  Blossom-MR (B-MR)
  Greedy-MR
  Kernel Herding (KH)
- Our Method:
  Grid-MR (G-MR)

  Ramdom Sampling as a pre-processing step
  - Improve the building time
  - Larger but acceptable error

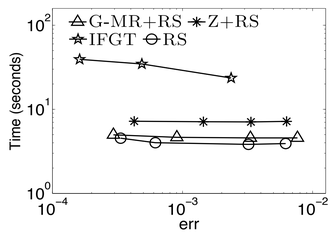


(a) Construction time vs. err

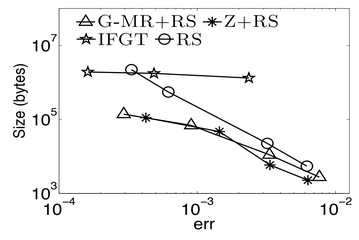


(b) Construction time vs. err

- Other Baseline Methods:
  Ramdom Sampling (RS): Comparable construction time but larger sample size
  IFGT: Both construction time and size is not as good as our methods
- Our Methods:
  Grid-MR+RS (G-MR+RS) ZOrder+RS (Z-MR+RS)



(a) Construction time vs. err



(b) Size vs. err