# KDE-Track: An Efficient Dynamic Density Estimator for Data Streams (Extended Abstract)

Abdulhakim A. Qahtan
*QCRI, HBKU*
*Doha, Qatar*
*aqahtan@hbku.edu.qa*

Suojin Wang
*Department of Statistic, Texas A&M University*
*College Station, Texas 77843-3143, USA*
*sjwang@stat.tamu.edu*

Xiangliang Zhang
*CEMSE, KAUST*
*Thuwal 23955, KSA*
*xiangliang.zhang@kaust.edu.sa*

***Abstract*—Recent developments in sensors, global positioning system devices and smart phones have increased the availability of spatiotemporal data streams. Developing models for mining such streams is challenged by the huge amount of data that cannot be stored in the memory, the high arrival speed and the dynamic changes in the data distribution. Density estimation is an important technique in stream mining for a wide variety of applications. In this paper, we present a method called KDE-Track to estimate the density of spatiotemporal data streams. KDE-Track can efficiently estimate the density function with linear time complexity using interpolation on a kernel model, which is incrementally updated upon the arrival of new samples from the stream.**

## I. Introduction

Recent advances in computing technology allow for collecting vast amount of data that arrive continuously in data streams. Examples of data streams can be found in fields such as sensor networks, mobile data collection platform, and network traffic. The data need to be processed and analyzed once they arrive. However, the unbounded, rapid and continuous arrival of data streams disallow the usage of traditional data mining techniques. Therefore, the development of algorithms for processing data streams instantaneously becomes highly important.

Density estimation has been widely used in various applications. Estimating the Probability Density Function (PDF) for a given data set provides knowledge about the underlying distribution of the data. Consequently, dense regions can be recognized as clusters and quantities such as medians and centers of clusters can be computed [1]. By contrast, sparse regions are reported as outliers that can be used for fault detection, e.g., in sensor networks [2].

In our paper [3], we presented a method, called KDE-Track, to model the data distribution as a set of resampling points with their estimated PDF. To guarantee the estimation accuracy and to lighten the load on the model, an adaptive resampling strategy is employed to control the number of resampling points, i.e., more points are resampled in the areas where the PDF has a larger curvature, while less number of points are resampled in the areas where the function is approximately linear. In order to overcome the quadratic time complexity of KDE when evaluating the PDF for each new observation, linear interpolation is used with KDE for online density estimation. To timely track the evolving density, a sliding window strategy in is used to estimate the density using the most recent data samples.

## II. Challenges

Estimating the dynamic density that comes with evolving streams is a challenging task. Besides the problem of estimating the density using samples drawn from an unknown distribution in case of stationary data, data streams have more challenging properties that complicate the estimation of density. First, the data distribution changes dynamically in an unpredictable fashion. Therefore, density estimation should rely more on the recently received data samples [4], e.g., by using a sliding window. Second, an anytime-available model should be efficiently updated to allow real-time monitoring of the density. Meanwhile, the density function value of any new arriving data may need to be instantly estimated. Third, the spatial non-uniformity of data distribution requires higher resolutions in dense areas and lower resolutions in sparse areas, so that the estimation is accurate to catch the details.

Most of the existing approaches for estimating the density of data streams are based on the Kernel Density Estimation (KDE) method due to its advantages for estimating the true density [5]. Given a set of samples, $S = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$, where $\boldsymbol{x}_j \in R^d$. KDE estimates the density at a point $\boldsymbol{x}$ as:

$$\hat{f}(\boldsymbol{x}) = \frac{1}{n} \sum_{j=1}^{n} K_h(\boldsymbol{x}, \boldsymbol{x}_j), \tag{1}$$

where $K_h(\boldsymbol{x}, \boldsymbol{x}_j)$ is a kernel function, which is usually a radially symmetric unimodal function that integrates to 1. Eq. (1) shows that KDE uses all the data samples to estimate the PDF of any given point. In the problem of online density estimation of data stream, i.e., estimating the density of every arriving data sample, KDE has quadratic time complexity and linear space complexity with respect to (w.r.t.) the stream size.

## III. Contribution

Our KDE-Track has unique properties as follows: (1) it generates density functions that are available to visualize the dynamic density of data streams at any time. At any time $t$, after receiving one streaming data sample $\boldsymbol{x}_t$, KDE-Track

updates the PDF of the data stream and also estimates $f(\boldsymbol{x}_t)$; (2) it has linear time and space complexities w.r.t. the model size for maintaining the dynamic PDF of data stream upon the arrival of every new sample. It is thus $8-85$ times faster than traditional KDE depending on the window size; (3) the estimation accuracy is guaranteed by adaptive resampling and optimized bandwidth ($h$), which also address the spatial non-uniformity issue of spatiotemporal data streams. Comparing with a set of baseline methods, it achieves the lowest estimation error, especially when the density function is multimodal and complex.

Both theoretical analysis and experimental results on synthetic and real-world data show the effectiveness of our approach for estimating the dynamic density functions that come with spatiotemporal data streams.

## IV. Theoretical Bases

KDE estimates the density $\hat{f}(\boldsymbol{x})$ by Eq. (1). For the 2-d spatial samples, where $\boldsymbol{x}_j = (x_{1j}, x_{2j})^T \in R^2$, kernel functions $K_h(\boldsymbol{x}, \boldsymbol{x}_j)$ are defined as $\frac{1}{h_1 h_2} K\left(\frac{x_1 - x_{1j}}{h_1}, \frac{x_2 - x_{2j}}{h_2}\right)$, where $h_i$ is the smoothing parameter, called the bandwidth, on dimension $i$ [5]. A popular kernel function in case of multivariate data is called the multiplicative (product) kernel [5], which uses the product of univariate kernel functions on each dimension, and computes $\hat{f}(\boldsymbol{x})$ as: $\hat{f}(\boldsymbol{x}) = \frac{1}{n}\sum_{j=1}^{n}\prod_{i=1}^{2}\left\{\frac{1}{h_i}K\left(\frac{x_i - x_{ji}}{h_i}\right)\right\}$. The choice of a kernel function is relatively unimportant provided that a kernel function is continuous with finite support [6]. The estimation accuracy of KDE is affected by the bandwidth value [6].

## V. KDE-Track Method

We model the distribution of the streaming data as a grid of resampling points and their corresponding estimated density values. Let $\mathcal{U}^1 = \{u_0^1, u_1^1, \ldots, u_{U^1-1}^1\}$ and $\mathcal{U}^2 = \{u_0^2, u_1^2, \ldots, u_{U^2-1}^2\}$ be the set of points that discretize the range of the data on the first and the second dimensions, respectively. The KDE-Track model $\mathcal{M}$ is defined as the set of the grid points from $\mathcal{U}^1 \times \mathcal{U}^2$ with their estimated densities. That is, $\mathcal{M} = \{M_0, M_1, \ldots, M_{q-1}\}$, where $q = U^1 U^2$ is the number of the resampling points and $M_s$ is an ordered pair representing a grid point and its estimated PDF ($M_s = (\boldsymbol{m}_s, \hat{f}(\boldsymbol{m}_s))$). Here $\boldsymbol{m}_s = (u_k^1, u_l^2) \in \mathcal{U}^1 \times \mathcal{U}^2$ is the $s$-th resampling point with $l, k$ being the quotient and the remainder of the division of $s$ by $U^1$ and $\hat{f}(\boldsymbol{m}_s)$ is the density estimated using KDE at $\boldsymbol{m}_s$.

Density estimation using bilinear interpolation is based on constructing the grid of resampling points and estimating their corresponding density values. Estimating the PDF at a data sample $\boldsymbol{a}$ by bilinear interpolation of the resampling points has two steps:

(1) fetch the estimated PDF values at points $\boldsymbol{m}_{s1}, \boldsymbol{m}_{s1+1}, \boldsymbol{m}_{s2}$ and $\boldsymbol{m}_{s2+1}$ that surround the point $\boldsymbol{a}$. Let $y^{(i)}$ be the projection of vector $\boldsymbol{y}$ on $i$-axis, then $m_{s1}^{(1)} = m_{s2}^{(1)} \leq a^{(1)} < m_{s1+1}^{(1)} = m_{s2+1}^{(1)}$ and $m_{s1}^{(2)} = m_{s1+1}^{(2)} \leq a^{(2)} < m_{s2}^{(2)} = m_{s2+1}^{(2)}$;

(2) estimate the density at $\boldsymbol{a}$ using linear interpolation.

***Adaptive resampling model:*** The accuracy of the linear interpolation depends on 1) the distance between two adjacent resampling points; and 2) the second derivative of the density function. To minimize the error while keeping the number of resampling points within a reasonable margin, we add more resampling points in the regions where the PDF has high curvature. By contrast, in the regions where the PDF is approximately linear, we use less resampling points.

In spatiotemporal data streams, the distribution is spatial non-uniform and dynamic. Therefore, high resolution with sufficient resampling points is required 1) in dense areas with high PDF values to catch the details; and 2) in sensitive areas which are the boundary between dense and sparse to catch dynamic changes. Adaptive resampling meets the requirement perfectly because both dense areas and sensitive areas generally have density function with high curvature.

***Updating the resampling model:*** As the density function is changing over time in evolving data streams, the resampling model should be updated online to capture the current trends in the stream. Updating the resampling involve three main operations: 1) updating the density function at the set of resampling points; 2) extending and shrinking the model to cover the current density function's support; 3) adding/removing the resamling points depending on the changes on the curvature of the density function.

***Application:*** One of the main advantages of using KDE-Track for density estimation is the availability of the density function values at the set of resampling points at any time point, which can be used for monitoring the dynamic density. In 2013, more than 170 million taxi trips were recorded in the city of New York. Visualizing the Taxi dataset allows for discovering interesting patterns of community behavior that differ in regular working days from weekends and national holidays, which helps service planners to forward taxicabs to the areas where they are required more.

## References


[1] A. Zhou, Z. Cai, L. Wei, and W. Qian, "M-kernel merging: Towards density estimation over data streams," in *DASFAA*, 2003.

[2] S. Subramaniam, T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos, "Online outlier detection in sensor data using non-parametric models," in *VLDB*, 2006.

[3] A. Qahtan, S. Wang, and X. Zhang, "KDE-Track: An efficient dynamic density estimator for data streams," *TKDE*, vol. 29, no. 3, pp. 642–655, Mar. 2017.

[4] C. Heinz and B. Seeger, "Cluster kernels: Resource-aware kernel density estimators over streaming data," *TKDE*, vol. 20, pp. 880–893, 2008.

[5] D. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons, 1992.

[6] B. Silverman, *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.