Multi-dimensional Probabilistic Regression over Imprecise Data Streams

Gao, Ran; Xie, Xike; Zou, Kai; Bach Pedersen, Torben

Published in:
WWW 2022 - Proceedings of the ACM Web Conference 2022

DOI (link to publication from Publisher):
10.1145/3485447.3512150

Publication date:
2022

Document Version
Accepted author manuscript, peer reviewed version

Multi-dimensional Probabilistic Regression over Imprecise Data Streams

Ran Gao*, Xike Xie*, Kai Zou*, and Torben Bach Pedersen†
gr0719@mail.ustc.edu.cn,xkxie@ustc.edu.cn,slnt@ustc.edu.cn,tbp@cs.aau.dk
*University of Science and Technology of China
† Daisy, Aalborg University, Denmark

ABSTRACT

In applications of Web of Things or Web of Events, a massive volume of multi-dimensional streaming data are automatically and continuously generated from different sources, such as GPS, sensors, and other measurement devices, which are essentially imprecise (inaccurate and/or uncertain). It is challenging to monitor and get insights over imprecise and low-level streaming data, in order to capture potentially important data changing trends and to initiate prompt responses. In this work, we investigate solutions for conducting multi-dimensional and multi-granularity probabilistic regression for the imprecise streaming data. The probabilistic nature of streaming data poses big computational challenges to the regression and its aggregation. In this paper, we study a series of techniques on multi-dimensional probabilistic regression, including aggregation, sketching, popular path materialization, and exception-driven querying. Extensive experiments on real and synthetic datasets demonstrate the efficiency and scalability of our proposals.

1 INTRODUCTION

In the context of Web of Things (WoT), Web of Events (WoE), Web of Energy, etc., massive volumes of streaming data are continuously and automatically generated from various sources in the under infrastructures, such as sensor networks, RFID devices, GPS, and so on [1, 2]. The stream of sensor data serves as the backbone of emerging smart applications, such as smart warehousing, smart buildings, and smart cities. Therefore, the real-time monitoring [3, 4] and analyzing [5, 6] of streaming data are important in intelligent WoT and WoE event processing and decision making [3–6].

Regression and Multi-dimensional Regression. Conventional multi-dimensional regression of precise data stream is fundamental in online streaming data analytics [7]. Regression analysis is the study of dependent variables on independent variables. For example, let time be the independent variable t_i, and measure (e.g., temperature) M_i be the dependent variable. Then, we get a series of tuples like $\langle (t_1, M_1), (t_2, M_2), ...\rangle$, which can be summarized by a *regression line* γ, with slope η and intersect θ.

$$\gamma : \hat{M} = \hat{\eta} * t + \hat{\theta}, \text{ where} \quad (1)$$

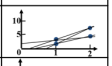
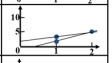
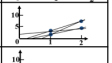
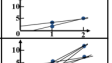
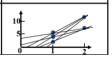
$$\hat{\eta} = \frac{\sum_i (t_i - \bar{t}) * (M_i - \overline{M})}{\sum_i (t_i - \bar{t})^2} \text{ and } \hat{\theta} = \overline{M} - \hat{\eta} * \bar{t}$$

Measure M_i can be specified by a given set of dimensions (e.g., type and location).

Imprecise Data Streams. However, the streaming data in WoT and WoE applications are often imprecise and inaccurate, the so-called veracity challenge. According to [8–12], data imprecision in WoT is inherent, and is caused by various reasons, including measurement inaccuracies [13, 14], interpolation errors [10], and transmission latencies [11]. Nevertheless, in WoE, events are usually non-deterministic due to unreliable data sources and networks [8]. R2C2: To tackle the veracity challenge, probabilistic modeling has been applied for capturing the imprecision of WoT streaming items[8, 9], in order to get the confidence of upper level event and query evaluation [8–12]. In this work, we model the imprecision of streaming data items by the general and commonly accepted *tuple-uncertainty* model in probabilistic data management [15], and study the problem of multidimensional probabilistic regression.

Object	Time	Instance	Location,Type	Measure	Prob
o_1	t_1	s_1	*US, Level 1*	2	0.6
		s_2	*US, Level 1*	1	0.4
o_2	t_2	s_3	*US, Level 1*	3	0.5
		s_4	*US, Level 1*	7	0.5
o_3	t_1	s_5	*UK, Level 1*	2	0.5
		s_6	*UK, Level 1*	3	0.5
o_4	t_2	s_7	*UK, Level 1*	4	1.0

(a) Probabilistic Streaming Raw Data

Cuboid(group-by)	*cell*	*Location*	*Type*	*PRLs*
(Location,Type)	C_1	*US*	*Level 1*	
	C_2	*UK*	*Level 1*	
(Location,)*	C_3	*US*	*	
	C_4	*UK*	*	
(,Type)*	C_5	*	*Level 1*	
	C_6	*	*Level II*	*NULL*
(all)	C_7	*	*	

(b) Regression data cube for Figure 1a

Figure 1: An Example of Probabilistic Streaming Regression

An Example. In Figure 1a, there are three streaming objects o_1 to o_3, with timestamps t_1 to t_2. Each object has multiple possible instances, due to multiple possible values caused by measurement errors, or multiple readings caused by signal bouncing, etc., which can be captured by the common tuple uncertainty model [15–18]. Each instance has multiple dimensions and a measure value, associated with a probability mass function (pmf), indicating the possibility of its appearance. For example, in Figure 1a, object o_1 has two possible instances, s_1 and s_2, with probabilities 0.6 and 0.4, respectively. The probabilistic data streams can thus be modelled as a sequence of imprecise objects $\langle (t_s, o_s), ..., (t_i, o_i), ..., (t_e, o_e)\rangle$, where o_i refers to the object arriving at time t_i.

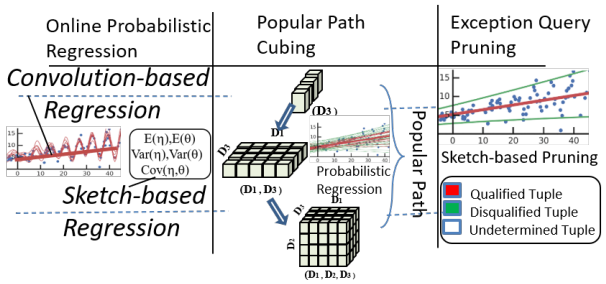


Figure 2: Framework

Framework. The framework for probabilistic streaming data regression aggregates the probabilistic regressions of multiple dimensions and multiple levels, as shown in Figure 2. Conceptually, a *cuboid* [19] corresponds to a group-by of a particular combination of dimensions. A *cell* refers to a tuple of a given cuboid. An example is shown in Figure 1b. There are 4 cuboids (group-bys), and cuboid (Location, Type) has 2 cells C_1 and C_2. Probabilistic regressions can be used for monitoring the trend by regressing imprecise data streams in real-time, alerting to streaming trend exceptions, and thus offering insights from multiple aspects.

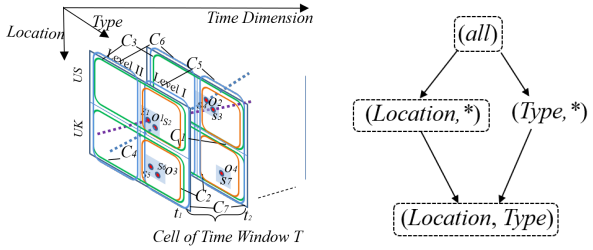
Table 1: PRL under Possible World Semantics

ID	Possible world W_i	γ_i
W_1	$\{(t_1, s_1), (t_2, s_3)\}$	$(1, 1, 0.3)$
W_2	$\{(t_1, s_1), (t_2, s_4)\}$	$(5, -3, 0.3)$
W_3	$\{(t_1, s_2), (t_2, s_3)\}$	$(2, -1, 0.2)$
W_4	$\{(t_1, s_2), (t_2, s_4)\}$	$(6, -5, 0.2)$

Probabilistic Regression. The regression result of a series of imprecise streaming objects, can be represented as a Probabilistic Regression Line (PRL *in short*). A PRL is essentially a set of triples $\{(\eta_i, \theta_i, p_i)\}_{i=1,2,\ldots}$, where a (η_i, θ_i) pair uniquely represents a possible regression and p_i is the corresponding probability. A PRL can be interpreted by Possible World Semantic (PWS *in short*) [17], which transforms a probabilistic time series into a set of non-probabilistic time series. Each non-probabilistic time series is called a possible world W_i associated with probability $p(W_i)$. W_i can be obtained by selecting an instance from each object at a time. Considering the regression aggregation of cell C_1 in Figure 3a, there are 4 possible worlds, as shown in Table 1. The probability of each possible world equals to the product of the corresponding instance probabilities, e.g., $p(W_1) = p(s_1) \cdot p(s_3) = 0.6 \times 0.5 = 0.3$. By doing that, we can get the regression result for each possible world, and formulate the PRL by integrating the regression results. Since each possible world has a probability, the integrated PRL is a two-dimensional pmf, i.e., probability distribution for all (η_i, θ_i) pairs, as shown in Table 1.

Challenges in Probabilistic Regression. Despite the semantic comprehensiveness of PWS, the computational cost is high. If there are n objects, each with m instances, there can be m^n possible worlds, which makes the aggregation infeasible in any case, in particular for real-time stream processing. To tackle that, we present a convolution-based method to process probabilistic regression in polynomial time. It satisfies PWS, so that the correctness is guaranteed [17]. Further, we offer an alternative aggregation of regression, called sketch-based regression, which approximates convolution-based regression in linear time.

Challenges in Popular-path Materialization. In analysis over multiple dimensions, cuboids are usually computed along a predefined popular path, instead of fully materializing all cuboids, since that would require excessive space and time. R3C1: A *popular path* [7] refers to a user-specified path from a lower to an upper level of cuboids, within the cuboid lattice. For example, we give two paths from *all* to $(Location, Type)$ in Figure 3b, and choose one path from (all) to$(Location, *)$ to $(Location, Type)$ as a popular-path.



(a) Multi-D Prob Stream Regression **(b) Popular path**

Figure 3: Interpretation of streaming cubes and popular path

R1C1: **Contribution.** In this paper, we present the first work on multi-dimensional probabilistic regression for imprecise data streams. For efficient regression processing, we propose convolution-based and sketch-based regressions, taking polynomial and linear time, respectively. We study intra- and extra-regressions, which support popular-path materialization in real-time. Finally, we study probabilistic exception-driven queries.

R2C1: **Relevance.** Web OLAP (WOLAP) has already been successfully used to analyze semi-static (semantic/linked) web data [20–23] and even semi-static WoT data [5]. Stream mining and analytics are becoming increasingly important in the Web community [4, 24, 25]. Recently, WoT technology has been applied to uncertain/imprecise data [8, 9]. Thus, there is an unmet need to extend current WOLAP techniques and engines to imprecise streaming sensor data from WoT devices. This paper fills this need. Specifically, our techniques support real-time WOLAP over high-speed sensor data streams, where data imprecision from disparate WoT streams is covered by probabilistic modeling.

Organization. The rest of the paper is organized as follows. Section 2 studies convolution-based aggregation, which handles probabilistic regression in polynomial time. Section 3 investigates sketch-based aggregation, which approximates probabilistic regression in linear time. Section 4 introduces drilling path query, based on convolution- and sketch-based aggregation. Section 5 presents comprehensive empirical studies on the performance of our proposals, with extensive experiments. Section 7 concludes the paper. Table 2 lists all notations used throughout this paper.

2 CONVOLUTION-BASED REGRESSION

2.1 Preliminaries

Existential Pmf and Measure Pmf. Of probabilistic data streams, an object corresponds to a set of possible instances, representing

Table 2: Notations

Notation	Meaning
o_i	Probabilistic data stream item
s_i	Data instance
t_i	Timestamp when data stream item o_i is read
η	Slope of regression line
θ	Intercept of regression line
γ	A probabilistic regression line
$\otimes$	Intra-regression
$\odot$	Extra-regression

its possible existence. Each instance is associated with a probability, indicating the likelihood of its appearance in the domain space. Of an object, the summation of all instances' possibilities equals to 1, indicating the object must exist. For example, object o_1 in Table 1 (a) arrives at t_1, with existential pmf as $\{(s_1, 0.6), (s_2, 0.4)\}$. Measure represents aggregated results of specific dimensions. Since objects are imprecise, the measure is also imprecise. Considering object o_1, the measure pmf of o_1 can thus be represented by $g_{o_1}(M) = \{(2, 0.6), (1, 0.4)\}$, if modifying the existential pmf by replacing its instances s_1 and s_2 by corresponding measures 2 and 1, respectively. Notice that $g_{o_1}(M)$ can also be represented as a binary function $g_{o_1}(\eta, \theta)$, because of the quantitative relation between measure M and parameters η and θ, according to Equation 1.

Regression Pmf. Designed for trend analysis, probabilistic regression targets on finding critical changes over imprecise data streams. In particular, the time dimension is taken as the independent variable, and the measure of specified dimensions is treated as the dependent variable. Since measure follows probabilistic pmf as aforementioned, the regression is evaluated in a probabilistic manner, called probabilistic regression lines (PRL *in short*). A PRL γ is represented by a set of repression lines, following a probability mass function f_γ. Since a regression line can be denoted as the slope-intercept pair (η_i, θ_i), the pmf of γ is essentially a two-dimensional distribution, written as $f_\gamma(\eta, \theta)$=$\{(\eta_1, \theta_1, p_1), (\eta_2, \theta_2, p_2), ...\}$.

2.2 Operations for Probabilistic Regression

Since streaming objects arrive sequentially, their regression should also be processed incrementally. For efficient processing, we introduce two operations for fulfilling probabilistic regression, *intra-regression* $\otimes$ and extra-regression $\odot$. The former refers to the case that the incoming object is "obsolete", meaning that the object's timestamp is not the latest in the current cell. The latter refers to the case that the object is the latest object of the current cell. Thus, the two operations are defined according to the two cases, respectively.

Intra-regression. The concept of intra-regression is formalized in Definition 1. An example is shown in Figure 4, before incoming object o arrives, there are three objects at time window $[1, 3]$. Initially there is only one regression line in the regression pmf γ_S in Figure 4, made up by these three objects $S = \{o_1, o_2, o_3\}$ in Figure 4b. After o arrives, whose timestamp is 2, we can get a regression pmf $\gamma_{S'}$ of 2 possible regression lines in Figure 4a. Noticed that the time windows is still $[1, 3]$. The correctness of operation $\otimes$ is guaranteed by Lemma 1.

Definition 1 (Operation $\otimes$). *Given a PRL $\gamma_S \sim f_{\gamma_S}(\eta, \theta)$ with time window $[t_s, t_e]$, where $S = \{o_1, o_2, ...\}$ refers to the set of objects formulating the regression. Suppose an incoming object o whose measure pmf is $g_o = \{(M_1, p_1), ..., (M_m, p_m)\}$. The timestamp of o is t_o, satisfying $t_o \in [t_s, t_e]$. The updated PRL is thus represented by $\gamma_{S'} \sim f_{\gamma_{S'}}$, where $S' = S \cup \{o\}$. Formally,*

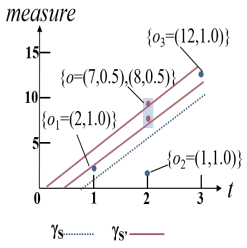
$$f_{\gamma_{S'}}(\eta, \theta) = (f_{\gamma_S} \otimes g_o)(\eta, \theta) = \sum_{(\eta_1, \theta_1)} f_{\gamma_S}(\eta_1, \theta_1) \cdot g_o(Y) \tag{2}$$

where Y is a function of $(\eta, \theta, \eta_1, \theta_1)$, defined as :

$$Y = \begin{cases} \dfrac{\eta - \eta_1}{A_1}, if \ \dfrac{\eta - \eta_1}{A_1} = \dfrac{\theta - \theta_1}{B_1} \\ NULL \end{cases} \tag{3}$$

$$where \begin{cases} A_1 = \dfrac{6(2K - T - 1)}{T^3 - T}, B_1 = \dfrac{2(2T - 3K + 1)}{T(T - 1)} \\ T = t_e - t_s + 1, K = t_o - t_s + 1 \end{cases} \tag{4}$$

When Y is $NULL$, the value of Y does not exist in g_o, and $g_o(NULL)$ is set as 0.



(a) PRL Interpretation (γ_S)

γ_S, where $S = \{o_1, o_2, o_3\}$

ID	(η, θ)	Prob
W_1	$(5, -5)$	1

$\gamma_{S'}$, where $S' = \{o_1, o_2, o_3, o\}$

ID	(η, θ)	Prob
W_2	$(5, -2.67)$	0.5
W_3	$(5, -2.33)$	0.5

(b) Possible World Interpretation

Figure 4: An Example of Operation $\otimes$

Lemma 1. *Given a pmf of a PRL, f_{γ_S}, with time window is $[t_s, t_e]$, and a measure pmf $g_o = \{(M_1, p_1), ..., (M_m, p_m)\}$ of an incoming object o, with timestamp $t_o \in [t_s, t_e]$, the intra-regression of f_{γ_S} and g_o is $f_{\gamma_{S'}}(\eta, \theta) = (f_{\gamma_S} \otimes g_o)(\eta, \theta)$, which satisfies PWS.*

Proof 1. *Let $I(x)$ be an indicator function, which returns 1 if x is true and 0 otherwise. W is the set of all possible worlds. We have:*

$$f_{\gamma_{S'}}(\eta, \theta) = \sum_{(\eta_1, \theta_1)} f_{\gamma_S}(\eta_1, \theta_1) \cdot g_o(Y) \tag{5}$$

$$= \sum_{W_i \in W} p(W_i) \cdot \underset{(\eta_1, \theta_1, M_i) \in W_i}{I} (\eta_1 + A_1 \cdot M_i = \eta \ and \ \theta_2 + B_2 \cdot M_i = \theta)$$

We proof that $\eta = \eta_1 + A_1 \cdot M_i$ and $\theta = \theta_1 + B_1 \cdot M_i$ for each possible world W_i. The measure of W_i at t_o before o arrived is M_t, and will become $M_t + M_i$ after o arrived, and the mean of measure $\overline{M}$ of W_i

become $\overline{M'} = \frac{T\bar{M}+M_i}{T}$.

$$\begin{aligned}\eta &= \frac{\sum_{t=1}^{T} t \cdot M'_t - T \cdot \bar{t} \cdot \bar{M'}}{\sum_{t=1}^{T}(t-\bar{t})^2} \\ &= \frac{\sum_{t=1}^{T} t \cdot M_t - T \cdot \bar{t} \cdot \bar{M}}{\sum_{t=1}^{T}(t-\bar{t})^2} + \frac{t_o \cdot M_i - \bar{t} \cdot M_i}{\sum_{t=1}^{T}(t-\bar{t})^2} \\ &= \eta_1 + \frac{6(2K-T-1)}{T^3-T} M_i\end{aligned}$$

Then we can use $\theta = \overline{M'} - \eta \cdot \frac{\sum_{t=1}^{T} t}{T}$ *to proof* $\theta = \theta_1 + B_1 \cdot M_i$

Extra-regression. The concept of extra-regression is formalized in Definition 2. An example is shown in Figure 5. Before o arrives, there are two objects at the time window $[1, 2]$, and there is only one PRL in regression pmf γ_S in Table 5b, made up by two objects $S = \{o_1, o_2\}$. After o arrives at $t_o = 3$, it expands the time window of the current cell from $[1, 2]$ to $[1, 3]$. We can get an updated regression pmf $\gamma_{S'}$ of 2 possible regression lines, as shown in Table 5b. The correctness of operation $\odot$ is formally proved in Lemma 2.

Definition 2 (Operation $\odot$). *Given a PRL* $\gamma_S \sim f_{\gamma_S}(\eta, \theta)$ *with time window* $[t_s, t_e]$, *where* $S = \{o_1, o_2, ...\}$ *refers to the set of objects formulating the regression. Suppose an incoming object* o *whose measure pmf is* $g_o = \{(M_1, p_1), ..., (M_m, p_m)\}$. *The timestamp of* o *is* t_o, *satisfying* $t_o = t_e + 1$. *The updated PRL is thus represented by* $\gamma_{S'} \sim f_{\gamma_{S'}}$, *where* $S' = S \cup \{o\}$. *Formally,*

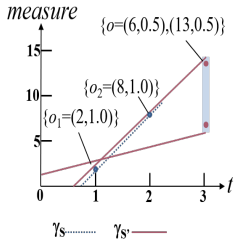
$$\begin{aligned}f_{\gamma_{S'}}(\eta, \theta) &= (f_{\gamma_S} \odot g_o)(\eta, \theta) \\ &= \sum_{(\eta_1, \theta_1)} f_{\gamma_S}(\eta_1, \theta_1) \cdot g_o(Y)\end{aligned} \tag{6}$$

where Y *is a function of* $(\eta, \theta, \eta_1, \theta_1)$, *defined as :*

$$Y = \begin{cases} \frac{\eta - C_1\eta_1 - C_2\theta_1}{C_3}, if\ \frac{\eta - C_1\eta_1 - C_2\theta_1}{C_3} = \frac{\theta - D_1\eta_1 - D_2\theta_1}{D_3} \\ NULL \end{cases} \tag{7}$$

$$where \begin{cases} C_1 = \frac{T-4}{T+2}, C_2 = -\frac{6}{(T+1)(T+2)}, C_3 = \frac{6}{(T+1)(T+2)} \\ D_1 = 2, D_2 = \frac{T+3}{T+1}, D_3 = -\frac{2}{T+1} \\ T = t_e - t_s + 1 \end{cases} \tag{8}$$

When Y *is* $NULL$, *the value of* Y *does not exist in* g_o, *and* $g_o(NULL)$ *is set as* 0.



(a) PRL Interpretation (γ_S)

γ_S, where $S = \{o_1, o_2, \}$

ID	(η, θ)	Prob
W_1	$(6, -4)$	1

$\gamma_{S'}$, where $S' = \{o_1, o_2, o\}$

ID	(η, θ)	Prob
W_2	$(2, 1.33)$	0.5
W_3	$(5.5, -3.33)$	0.5

(b) Possible World Interpretation

Figure 5: An Example of Operation $\odot$

Lemma 2. *Given a pmf of a PRL* f_{γ_S} *with time window* $[t_s, t_e]$, *and a measure pmf* $g_o = \{(M_1, p_1), (M_2, p_2), ..., (M_m, p_m)\}$ *of an object* o, *with timestamp* $t_o = t_e + 1$, *the extra-regression of* f_{γ_S} *and* g_o *is* $f_{\gamma_{S'}}(\eta, \theta) = (f_{\gamma_S} \odot g_o)(\eta, \theta)$ *which satisfies PWS.*

Proof 2. *Similar proof 1, we should prove*

$$\begin{aligned}f_{\gamma_{S'}}(\eta, \theta) = \sum_{W_i} p(W_i) \cdot \underset{(\eta_1, \theta_1, M_i) \in W_i}{I} (C_1\eta_1 + C_2\theta_1 \\ +C_3 \cdot M_i = \eta \text{ and } D_1\eta_1 + D_2\theta_1 + D_3 \cdot M_i = \theta)\end{aligned} \tag{9}$$

After extra-regression, the time window becomes $[1, T+1]$. *According to Equation 1, the integrated slope* η_i *of possible world* W_i *is:*

$$\begin{aligned}\eta &= \frac{\sum_{t=1}^{T+1} t \cdot M'_t - (T+1) \cdot \bar{t} \cdot \bar{M'}}{\sum_{t=1}^{T+1}(t-\bar{t})^2} \\ &= \frac{\sum_{t=1}^{T}(t-\bar{t})^2}{\sum_{t=1}^{T+1}(t-\bar{t})^2}\eta + \frac{(T+1)M_i + (T+1)\frac{T}{2}\bar{M} - (T)\frac{T+1}{2}\bar{M'}}{\sum_{t=1}^{T+1}(t-\bar{t})^2}\end{aligned} \tag{10}$$

The variables $\overline{M}$ *and* $\overline{M'}$ *of Equation 10 are as follows.*

$$\overline{M} = \eta\frac{T}{2} + \theta \text{ and } \overline{M'} = \frac{\bar{M} \cdot T + M_i}{T+1} \tag{11}$$

Substituting the two items into Equation 10, we can have $\eta = C_1\eta_1 + C_2\theta_1 + C_3M_i$. *Similarly, we can proof* $\theta = D_1\eta_1 + D_2\theta_1 + D_3 \cdot M_i$.

The execution of the two operations are done in a convolution manner. Thus, we call the probabilistic regression based on the two operations as *convolution-based regression*. By defining the two operations, the probabilistic regression can be done in polynomial time.

Suppose each object has at most m instances. The operation $\otimes$ or $\odot$ over two objects takes at most $O(m^2)$ time cost and $2m$ space cost, because of the two loops in the convolution of Equation 2 or 9. Then, the pmf length of the two objects' PRL is at most $2m$. A convolution operation, i.e., $\otimes$ or $\odot$, between the PRL and an object can thus be evaluated in $O(m^2)$ time. The resulted PRL pmf takes at most $3m$ space. Notice that the pmf length increases as convolution continues. Convoluting n objects results in a pmf length $O(mn)$ at most, so that the time cost would be $O(m^2n^2)$. The time efficiency can be improved to $O(m^2n^2log(mn))$, if the two-dimensional Fast Fourier Transformation (FFT) is used. It can be simplified as $O(n^2log(n))$, if m is a constant.

Algorithm 1 Convolution-based regression

Input: A new object o, g_o **and regression pmf** f_{γ_S}
Output: Updated regression pmf $f_{\gamma_{S'}}$
  for each cuboid in popular-path **do**
    Find the cell match the value of dimensions
    $f_{\gamma_{S'}} = f_{\gamma_S} \otimes g_o$ or $f_{\gamma_{S'}} = f_{\gamma_S} \odot g_o$
  end for

Discussion. So far, we have a polynomial-time solution for regression aggregation. There are two questions to be answered. First, is the convolution-based method adequate for streaming setting? Second, when analyzing probabilistic regression lines from regression, do we need to query the regression pmf for all cells every time? For the first question, we devise an alternative solution called

sketch for fast approximating the regression aggregation and process online within the linear overhead. For the second problem, we chose pruning for certain cells which were of no analytical value with sketch, which can help improve query efficiency.

3 SKETCH-BASED REGRESSION

Convolution-based regression does not meet the velocity challenge of data streams. Therefore, we seek for alternative probability distribution representations for accelerating the aggregation of regression. In this section, we introduced cumulants, which are considered as effective features of probability distributions.

3.1 Cumulants: Sketch Variables for Regression

Cumulants are quantitative measures for defining the shape of a probability distribution. Usually, we call the first and second order cumulants the *mean* and *variance*. Theoretically, the collection of all cumulants of all orders, i.e., $k = 1, ...\infty$, can uniquely determine a probability distribution.

$$E(X) = \sum_x x f_X(x) \tag{12}$$
$$Var(X) = E(X^2) - E^2(X) = \sum_x x^2 f_X(x) - (\sum_x x f_X(x))^2$$

An alternative way of approximately representing a PRL is to evaluate its cumulants of a series of orders, instead of the exact form of probabilistic distributions. Ideally, a probabilistic distribution is equivalently to the series of all orders of cumulants. In practice, it is often adequate to approximate the cumulants of low orders, e.g., the first two orders of cumulants, according to [26] and [16].

So, for a cell, we maintain 5 sketching variables for representing the first two orders of cumulants, including the first two orders of cumulants $E(\eta)$, $E(\theta)$, $Var(\eta)$, $Var(\theta)$, and covariance $Cov(\eta, \theta)$. The covariance is needed, because the variables η and θ are not independent.

3.2 Operations for Sketch Regression

Now we study the operations for sketch-based regression. Suppose an incoming object o, which arrives at time t_o with measure pmf $f_o(M) = \{(M_i, p_i)\}_{i \le m}$, and the time span of current PRL is $[t_s, t_e]$. There are two possibilities for aggregating o into the regression, 1) o is the latest object, $t_o = t_e + 1$; 2) o is not the latest objects $t_o \in [t_s, t_e]$. For the first case, after the insertion of o, t_e is replaced by t_o.

We consider two types of sketch-based regression, intra- and extra-regressions, in consistent with convolution-based regressions. The two operations are covered by Lemmas 3 and 4, respectively.

Lemma 3 (Sketching Operation ⊗). *Given a PRL $\gamma_S \sim f_{\gamma_S}(\eta, \theta)$ represented by its first two order cumulants, and an incoming objet $o \sim g_o(M)$, the cumulants of updated PRL $\gamma'_S \sim f_{\gamma'_S}(\eta', \theta')$, where $f_{\gamma'_S}(\eta', \theta') = f_{\gamma_S} \otimes g_o(\eta', \theta')$ are as follows.*

$$\begin{aligned}
E(\eta') &= E(\eta) + A_1 E(M) \\
Var(\eta') &= Var(\eta') + A_1^2 Var(M) \\
E(\theta') &= E(\theta) + B_1 E(M) \\
Var(\theta') &= Var(\theta) + B_1^2 Var(M) \\
Cov(\eta', \theta') &= Cov(\eta, \theta) + A_1 B_1 Var(M)
\end{aligned}$$

$$where \begin{cases} A_1 = \dfrac{6(2K - T - 1)}{T^3 - T}, B_1 = \dfrac{2(2T - 3K + 1)}{T(T-1)} \\ T = t_e - t_s + 1, K = t_o - t_s + 1 \end{cases} \tag{13}$$

Proof 3. *As shown in Lemma 1, for operation ⊗, each possible world of γ_S can be expanded as m possible worlds of γ'_S after o is convoluted. Let pmf of o be $f_o = \{(M_1, p_{M1})...(M_m, p_{Mm})\}$. We have:*

$$\begin{aligned}
E(\eta') &= \sum \eta'_i p'_i = \sum (\eta + A_1 M_i) p'_i = \sum (\eta + A_1 M_i) p_i p_{Mi} \\
&= \sum \eta p_i + A_1 \sum M_i p_{Mi} = E(\eta) + A_1 E(M)
\end{aligned}$$

Then, $Var(\eta')$ can be directly obtained by substituting $E(\eta')$ into the equation of variance.

$$\begin{aligned}
&Var(\eta') = E(\eta'^2) - E(\eta')^2 = E((\eta + A_1 M_i)^2) - E(\eta + A_1 M_i)^2 \\
&= (E(\eta^2) - E(\eta)^2) + A_1^2 (E(M^2) - E(M)^2) = Var(\eta) + A_1^2 Var(M)
\end{aligned}$$

Similarly, we can derive $E(\theta')$, $Var(\theta')$, and $Cov(\eta', \theta')$. Hence, the lemma is proved.

Lemma 4 (Sketching Operation ⊙). *Given a PRL $\gamma_S \sim f_{\gamma_S}(\eta, \theta)$ represented by its first two order cumulants, and an incoming objet $o \sim g_o(M)$, the cumulants of updated PRL $\gamma'_S \sim f_{\gamma'_S}(\eta', \theta')$, where $f_{\gamma'_S}(\eta', \theta') = f_{\gamma_S} \odot g_o(\eta', \theta')$ are as follows.*

$$\begin{aligned}
E(\eta') &= C_1 E(\eta) + C_2 E(\theta) + C_3 E(M) \\
E(\theta') &= D_1 E(\eta) + D_2 E(\theta) + D_3 E(M) \\
Var(\eta') &= C_1^2 Var(\eta) + C_2^2 Var(\theta) + 2C_1 C_2 Cov(\eta, \theta) + C_3^2 Var(M) \\
Var(\theta') &= D_1^2 Var(\eta) + D_2^2 Var(\theta) + 2D_1 D_2 Cov(\eta, \theta) + D_3^2 Var(M) \\
Cov(\eta', \theta') &= (C_1 D_2 + C_2 D_1) Cov(\eta, \theta) + C_1 D_1 Var(\eta) \\
&\quad + C_2 D_2 Var(\theta) + C_3 D_3 Var(M)
\end{aligned}$$

$$where \begin{cases} C_1 = \dfrac{T-4}{T+2}, C_2 = -\dfrac{6}{(T+1)(T+2)}, C_3 = \dfrac{6}{(T+1)(T+2)} \\ D_1 = 2, D_2 = \dfrac{T+3}{T+1}, D_3 = -\dfrac{2}{T+1} \\ T = t_e - t_s + 1 \end{cases} \tag{14}$$

The proof of Lemma 4 is similar to that of Lemma 3, and are omitted due to page limits. For ease of understanding, we show an example of sketch-based operations in Table 3. In particular, we show how the expectation of θ' is derived from cumulants of θ and g_o in operation ⊗. Here, T equals $3 - 1 + 1 = 3$ and K equals $2-1+1 = 2$, according to Equation 13 and the timestamp information of $\{o_1, o_2, o_3, o\}$, as shown in Figure 4a. The value of B_1 can be calculated by $\frac{2(2\times3-3\times2+1)}{3\times(3-1)} = \frac{1}{3}$. The expectation of o's measure $E(M)$ is $7 \times 0.5 + 8 \times 0.5 = 7.5$. By substituting $E(M)$ and B_1 into $E(\theta')$, we get that $E(\theta') = -5+7.5/3 = -2.5$. The result is consistent with the one derived from $f_{\gamma'}$, that is $-2.67\times0.5+-2.33\times0.5 = -2.5$.

It shows that the sketching operation, ⊗ or ⊙, can be done in linear time complexity w.r.t. the number of imprecise objects, which is much more efficient that convolution-based regression. Thus, sketch-based regression is appropriate for handling imprecise data streams. Next, we show how sketch-based regression supports queries.

Table 3: Example of Lemma 3 For Fig 4

$f_{Y_S}(\eta, \theta) = \{(5, -5, 1.0)\}, g_o(M) = \{(7, 0.5), (8, 0.5)\}$
$f_{Y_{S'}}(\eta', \theta') = \{(5, -2.67, 0.5), (5, -2.33, 0.5)\}$

$E(\eta)$	5
$Var(\eta)$	0
$E(\theta)$	-5
$Var(\theta)$	0
$Cov(\eta, \theta)$	0

$E(\eta')$	5
$Var(\eta')$	0
$E(\theta')$	-2.5
$Var(\theta')$	6.31
$Cov(\eta', \theta')$	0

4 EXCEPTION-DRIVEN QUERIES WITH PROBABILISTIC REGRESSIONS

In this section, we discuss how probabilistic regressions support exception detection for imprecise data streams.

4.1 Exception-driven Queries

By analyzing PRLs, we can monitor the trend of the probabilistic data streams for a specific set of dimensions over a period of time. Similar to [7], we define exceptions based on the existence of slope outliers, i.e., the slope is higher than a threshold. Since the regression is probabilistic, we would like to return the exceptions which are above user-specified confidence threshold. Therefore, we define the probabilistic exception queries as follows.

Definition 3 (Probabilistic Exception Queries). *Over a set of given dimensions, a probabilistic exception query (PEQ in short) returns cells, whose slope values are higher than the slope threshold τ_η, and the qualification probability is higher than the probability threshold τ_p.*

Let $\eta \sim F_c(\eta)$ be a random variable representing the slope value of cell C. The qualification probability equals to the integration of pmf F_c over the range satisfying the query threshold. Then, PEQ returns all cells meeting τ_η and τ_p, formally,

$$\text{PEQ} = \{C_k | Prob(\eta \geq \tau_\eta) > \tau_p\} = \{C_k | \sum_{\eta \geq \tau_\eta} F_{C_k}(\eta) > \tau_p\}$$

This way, if a cell is returned by PEQ as exceptions, indicating the abnormal trend alerted. In Definition 3, the query semantics of PEQ focus on retrieving cells with the increasing trend faster than user-specific thresholds. Notice that the query semantics can also be extended to handle more general cases with slight modification.

After we gave the PEQ based on the exception definition of the Regression Pmf, the real-time response of the data only updated the Sketch variables. The cumulants of PRL pmfs can be used to get the upper and lower bounds for approximating PEQ qualification probabilities. Intuitively, if a cell's lower bound is above τ_p, it qualifies for the query. Otherwise, if a cell's upper bound is below τ_p, it is not qualified for the query answer. With so-called sketched-based pruning, the efforts on refinement process can be much saved. The formalization of upper and lower bounds are shown in Lemmas 5 and 6.

Lemma 5 (Upper Bounds). *If the expectation of the aggregated variable η is at most τ_η, the probability that $\eta > \tau_\eta$ is at most $\frac{Var(\eta)}{Var(\eta)+a^2}$, where $a = |\tau_\eta - E(\eta)|$. In other words, if $E(\eta) \leq \tau_\eta$, then $Prob(\eta > \tau_\eta) \leq \frac{Var(\eta)}{Var(\eta)+a^2}$.*

Lemma 6 (Lower Bounds). *If the expectation of the aggregated variable η is at least τ_η, the probability that $(\eta > \tau_\eta)$ is at most $\frac{a^2}{Var(\eta)+a^2}$, where $a = |\tau_\eta - E(\eta)|$. In other words, if $E(\eta) \geq \tau_\eta$, then $Prob(\eta > \tau_\eta) \leq \frac{a^2}{Var(\eta)+a^2}$.*

The error bounds can be efficiently derived based on sketching variables, e.g., $E(\eta)$ and $Var(\eta)$. Notice that there is no need to explicitly derive both of the two bounds, when applying sketch-based approximation. If the expectation of η is higher than threshold τ_η, the upper bound is derived for approximating the qualification probabilities. Otherwise, the lower bound is derived. Both bounds can be directly obtained from one-sided Chebyshev's inequality, so the proofs are omitted.

5 EXPERIMENTAL EVALUATION

We introduce the experimental setting in Section 5.1. We present the experimental result in Section 5.2

5.1 Setup

Datasets. We use the synthetic data (an adapted version of the well known TPC-H[1]) and real data (US Climate 2014[2]). For each tuple in the dataset, we viewed it as a multi-dimensional point. The uncertainty is injected into the tuple by creating a set of 5, 10, and 15 possible instances. The instance is randomly generated from a multi-dimensional orthogonal region, which is centered at the tuple. The side length of the region is about 20% of its dimensional value. By default, there are 10 possible instances for each object. The probability distribution of the instances are generated randomly. Then, we can get two probabilistic datasets, denoted as TPC-H and Climate. The statistics of the two datasets are shown as below.

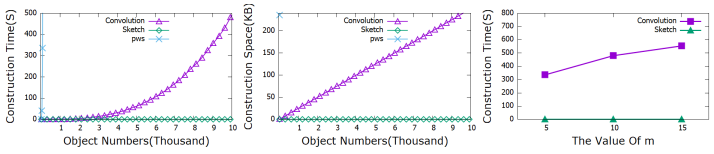
- TPC-H. We select Items, Suppliers, Consumers, and Locations as dimensions, besides the temporal dimension. We use price as the measure for the trend analysis.
- Climate. We identify attributes Wind Direction, Wind Speed, and Sea level pressure as dimensional attributes, besides the temporal dimensional attribute. We use Temperature as the measure. The analysis is on monitoring the temperature trend of sea area.

Implementation. In convolution aggregation, the slope and intercept, which range from negative infinite to positive infinite, are rounded to integers. The sketch-based method uses the first two orders of cumulants. All our programs were implemented in C++, with VS2019 IDE, and run on a PC with a 2.6GHz processor and 8GB RAM. Each point is the average of 10 runs.

5.2 Results

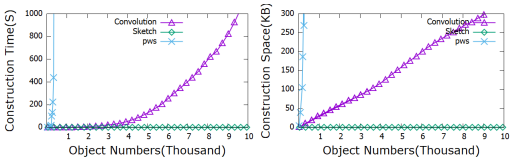
5.2.1 Aggregation. We compare the performance of aggregation for three methods, PWS, convolution-based method, and sketch-based method. We show the time cost of aggregation on TPC-H in Fig. 6, and the time cost on Climate in Fig. 7. The results show that the time cost of PWS increases dramatically w.r.t. to the number of objects. R2C4: On both datasets, PWS does not finish in acceptable time. It times out after 1000s when processing 1000 data objects.

[1]http://www.tpc.org/tpch/
[2]http://www.ncdc.noaa.gov/cdo-web/datasets



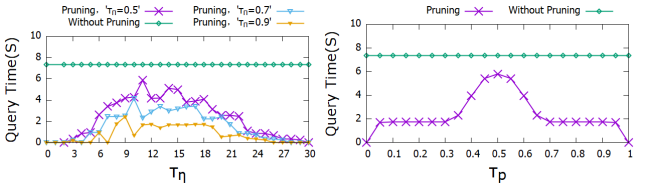
(a) Time vs Obj.(TPC-H) (b) Space vs Obj.(TPC-H) (c) Time vs m.(TPC-H)

Figure 6: Aggregation(TPC-H)R2C3



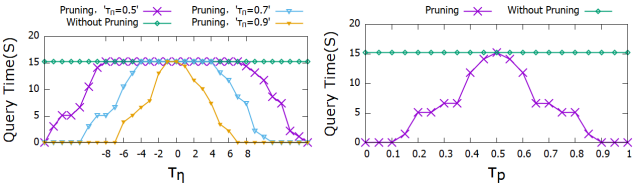
(a) Time vs Obj.(Climate) (b) Space vs Obj.(Climate)

Figure 7: Aggregation(Climate)R2C3



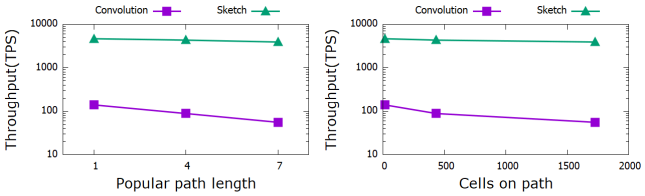
(a) Time vs of τ_η.(TPC-H) (b) Time vs of τ_p.(TPC-H)

Figure 8: Query(TPC-H)



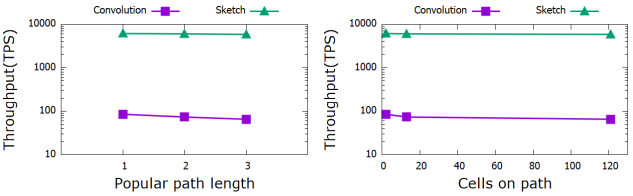
(a) Time vs of τ_η.(Climate) (b) Time vs of τ_p.(Climate)

Figure 9: Query(Climate)



(a) TPS vs of path length.(TPC-H) (b) TPS vs cells.(TPC-H)

Figure 10: Throughput(TPC-H)



(a) TPS vs of path length.(Climate) (b) TPS vs cells.(Climate)

Figure 11: Throughput(Climate)

In comparison to that, the performance of convolution- and sketch-based methods are much more efficient. It is therefore consistent with the time complexity analysis that PWS runs in exponential time, and the other two can be done in polynomial time. Also, sketch-based aggregation consumes orders of magnitudes less than convolution-based method in terms of time cost. For example, in Fig. 6a, when 10,000 objects are aggregated, convolution-based method takes about 480 seconds, while the sketch-based method takes about 0.05 seconds. Similar observations can be drawn from the Climate dataset, in Fig. 7a.

Then, we consider the space cost in Fig. 6b and Fig. 7b. The space cost of all the three methods increases with respect to the number of objects. PWS methods consumes highest space cost of the three competitors, since the number of possible worlds is exponential. Convolution-based aggregation takes space for storing pmfs of PRLs, so the time cost increases polynomially. The trend is super-linearly, because the pmf becomes longer for a larger number of objects. In contrast, sketch-based method only store 5 sketching variables, including the two expectation, two variance, and the covariance, for η and θ. So, the time cost increases linearly w.r.t. the number of objects. In particular, when 5,000 objects are aggregated, convolution-based method takes 127 KB, while sketch-based method takes merely 0.12 KB, in Fig. 6b. Similar observations can be drawn from Climate in Fig. 7b.

We also examine the effect of uncertainty by varying the number of possible instances of an object in Fig. 6c. It shows that both convolution- and sketch-based increase sub-linearly w.r.t. the number of instances m. In particular, when m grows 3 times (i.e., from 5 to 15), the time cost of convolution-based method only increases by 0.6 times. The time overhead of sketch-based method in insensitive to the increase of m. For example, when m grows 3 times, the time cost of sketch-based method increases by 0.3 times. From now on, we only show the results of convolution- and sketch-based methods, since PWS is completely dominated by the two.

5.2.2 Queries. We examine the performance of probabilistic exception queries, in Fig. 8 and Fig. 9. We test the result by varying the parameter τ_p and τ_η. In all testings, curves are in the bell shape, the query time first increases then decreases w.r.t. to τ_p and τ_η. This is as expected. For example, when τ_p equals 50%, it refers to the most ambiguous case so that the pruning effect is the mostly weakened. When τ_p increased to 70% and 90%, the pruning effect with different τ_η improved significantly in Fig. 8a and Fig. 9a(When we test the relationship between pruning action and τ_p, we set τ_p close to the majority of a PRL slope to avoid the effect of τ_p). When τ_p increases, more cells are disqualified with PEQ inquiry by using sketch-based pruning bounds. Reversely, if τ_p is set too low, most cells are directly qualified, because their lower bound easily above the threshold. Similar facts on τ_η can also be observed in Fig. 8b and Fig. 9b(When we test the relationship between pruning action and τ_η, we set τ_p equals 50% to avoid the effect of τ_p). When τ_η is close to the majority of a PRL slope, a tuple becomes harder to be pruned. This is consistent with Lemmas 5 and 6. If the value of τ_η is close to the expectation of η, a is close to zero, and the value of $\frac{Var(\eta)}{Var(\eta)+a^2}$ is close to 1, making the pruning bounds trivial. It is worth noting that the pruning bounds are adequate for exception

queries. Because a normal exception threshold is used to detect minorities instead of majorities, and the threshold value should be set away from the majority.

Then, we analyze the query processing to examine the effect of sketch-based pruning. Recall that the query overhead includes three parts, cell retrieval, pruning, and online cubing. The query performance can be much accelerated if a larger number of cells can be qualified or disqualified. Therefore, we compare the result with and without sketch-based pruning. The results show that the pruning effect is significant except extreme setting, as aforementioned. Also, we can observe that, in all testing, the query can be finished within 11 seconds, even for the worst case. Notice that in the worst case, few cells can be rejected by pruning bounds and all cells on the drilling path are cubing online. For example, in Fig. 8a, the majority is below 4 seconds. It demonstrate the efficiency of our proposed techniques.

5.2.3 Throughput. Hereby, we examine the throughput of online processing for cuboids on the popular-path in Fig. 10 and Fig. 11. In our experiments, we set different length of popular-path. R3C3. Notice that the length of a popular path reflects two things: 1) the number of the cuboid of the path to be materialized; 2) the dimensionality of cuboids, i.e., a longer path corresponds to a cuboid with more dimensions. Also, for a cuboid with more dimensions, there tends to be more cells for aggregation. The corresponding computation overhead is thus higher. Then, we report the throughput w.r.t. the varied popular-path length in Fig. 10a and the number of cells on path in Fig. 11a. It shows that 1) the sketch-based online processing is capable of handling thousands of updates per second (minimum 3.9K TPS on TPC-H and minimum 5.8K on Climate); 2) the convolution-based online processing can handle about tens to hundreds of times per second depending on the path length(minimum 55 and maximum 140 TPS on TPC-H,minimum 65 and maximum 85 TPS on Climate); 3) the performance decreases with increasing path length, and the sketch-based update is relatively stable, while the convolution-based update performance degrades significantly as the path becomes longer. This is due to the different complexity of algorithms. We also test the performance w.r.t. the number of cells on popular-path. As the path becomes longer, the number of cells on the path increases accordingly, which increases the amount of computation and leads to performance decline in both Fig. 10b and Fig. 11b.

6 RELATED WORKS

There have been many works proposed to process or apply probabilistic data aggregation. In particular, [27] studies aggregation with the algebraic structures of semirings and semimodules. [26] considers the aggregation representation with histograms and wavelets. [28] and [29] investigate ranking semantics for probabilistic data. [18] uses frequency moments for efficient probabilistic data aggregation. Other query variants include reverse nearest neighbor queries [30] and skyline queries [31], etc. [16] and [32] study OLAP queries and cube materialization for static probabilistic objects. [33] scales down the large deterministic optimization problem in probabilistic query evaluation into a sequence of smaller near-optimal problems so as to improve the efficiency and quality. [34] extends tuple-uncertainty model for probabilistic databases to supporting probabilistic graphs. [35] utilizes probabilistic database model to represent Debugging Query 2.0 statements as a differentiable functions in order to support model inference and influence analysis. [36] studies workflow performance optimization by considering system variations as time-dependent random variables with probabilistic distributions. However, these works are not for probabilistic data streams.

There have also been works on probabilistic aggregation on data streams. [37] studies efficient inference with probabilistic graph models under temporal correlations. [38] calculates expectation and variance for probabilistic variables of data streams. [39] and [40] model event streams as probabilistic strings in supporting decision making of IoT applications. [41] studies tracking of probabilistic items in distributed environments. [42] investigates efficient evaluation of top-k queries for probabilistic data streams. Other query variants include range queries [43], join queries [44], nearest neighbor queries [45, 46], trajectory queries [47, 48], reverse nearest neighbor queries [30, 46], and skyline queries [31], data cleansing [49, 50], etc. However, none of them focuses on multidimensional regression analysis.

The problem of multi-dimensional regression for non-probabilistic streams is proposed in [7] and [51], where the ideas of o- and m-layers are firstly initiated. [52] generalizes the regression aggregation to pattern recognition so that interesting patterns can be automatically discovered from multi-dimensional data. [53] further proposes a structured knowledge representation for multi-dimensional data, and designs score functions to qualify commonness and exceptions. [54] uses multivariate Gaussian model to approximately represent multi-dimensional cuboids over massive data points in order to support fast interactive data analyzing. Those works are not for probabilistic data, where objects are represented in probabilistic distributions instead of precise values.

R3C2. In summary, there exist works on streaming regression for non-probabilistic data, and exist works on non-streaming aggregation for probabilistic data. As aforementioned, there is an unmet need for integrating the two in WoT computing environment. To our best knowledge, our work is the first on multi-dimensional probabilistic regression.

7 CONCLUSION

In this paper, we study the problem of multi-dimensional regression over probabilistic streaming data. To tackle the challenge of computation intensiveness, we study a series of techniques, including convolution- and sketch-based aggregation and probabilistic exception querying. For the aggregation, we propose convolution- and sketch-based aggregation, covering intra-regression and extra-regression on popular-path, and can be processed online. For both two aggregation methods, we prove their correctness through possible world semantics and analyze their complexity. We also study how the probabilistic exception queries can be facilitated by the aforementioned techniques, defines an exception query PEQ which focuses on trend analysis, and give a way for pruning determined cells/tuples with sketch. Experiment results show that our solution achieves good performance in combating the velocity and veracity challenge of probabilistic data streams.

REFERENCES

[1] Alexandra Moraru, Dunja Mladenic, Matevz Vucnik, Maria Porcius, Carolina Fortuna, and Mihael Mohorcic. Exposing real world information for the web of things. In *IIWeb In Conjunction with WWW 2011*, 2011.

[2] Lina Yao and Quan Z. Sheng. Correlation discovery in web of things. In *WWW*, Companion, page 215–216, 2013.

[3] David Vernet, Agustín Zaballos, Ramon Martin de Pozuelo, and Víctor Caballero. High performance web of things architecture for the smart grid domain. *IJDSN*, 11(12):347413, 2015.

[4] Yongqiang Liu and Xike Xie. Xy-sketch: on sketching data streams at web scale. In *TheWebConf*, pages 1169–1180, 2021.

[5] Muntazir Mehdi, Ratnesh Sahay, Wassim Derguech, and Edward Curry. On-the-fly generation of multidimensional data cubes for web of things. In *IDEAS*, pages 28–37, 2013.

[6] Hongming Cai and Athanasios V Vasilakos. Web of things data storage. In *Managing the Web of Things*, pages 325–354. Elsevier, 2017.

[7] Yixin Chen, Guozhu Dong, Jiawei Han, Benjamin W Wah, and Jianyoung Wang. Multi-dimensional regression analysis of time-series data streams. In *VLDB*, pages 323–334, 2002.

[8] Nathalie Moreno, Manuel F Bertoa, Gala Barquero, Loli Burgueño, Javier Troya, Adrián García-López, and Antonio Vallecillo. Managing uncertain complex events in web of things applications. In *ICWE*, pages 349–357. Springer, 2018.

[9] Samir Awad, Abdelhamid Malki, and Mimoun Malki. Composing wot services with uncertain and correlated data. *Computing*, pages 1–17, 2021.

[10] Jang-Ping Sheu and Huang-Fu Lin. Probabilistic coverage preserving protocol with energy efficiency in wireless sensor networks. In *WCNC*, pages 2631–2636. IEEE, 2007.

[11] Mohamed Hefeeda and Hossein Ahmadi. Energy-efficient protocol for deterministic and probabilistic coverage in sensor networks. *TPDS*, 21(5):579–593, 2009.

[12] Qianqian Yang, Shibo He, Junkun Li, Jiming Chen, and Youxian Sun. Energy-efficient probabilistic area coverage in wireless sensor networks. *IEEE Trans. Veh. Technol.*, 64(1):367–377, 2014.

[13] Junkun Li, Jiming Chen, and Ten H Lai. Energy-efficient intrusion detection with a barrier of probabilistic sensors. In *INFOCOM*, pages 118–126. IEEE, 2012.

[14] Marco AF Pimentel, Peter H Charlton, and David A Clifton. Probabilistic estimation of respiratory rate from wearable sensors. In *Wearable electronics sensors*, pages 241–262. Springer, 2015.

[15] Dan Suciu, Dan Olteanu, Christopher Ré, and Christoph Koch. *Probabilistic Databases*. Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.

[16] Xike Xie, Xingjun Hao, Torben Bach Pedersen, Peiquan Jin, and Jinchuan Chen. Olap over probabilistic data cubes i: Aggregating, materializing, and querying. In *ICDE*, pages 799–810, 2016.

[17] Nilesh N. Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. In *VLDB*, pages 864–875, 2004.

[18] T. S. Jayram, Satyen Kale, and Erik Vee. Efficient aggregation algorithms for probabilistic data. In *SODA*, 2007.

[19] Jiawei Han, Micheline Kamber, and Jian Pei. Data mining concepts and techniques, third edition, 2012.

[20] Nurefsan Gür, Jacob Nielsen, Katja Hose, and Torben Bach Pedersen. Geosemolap: Geospatial olap on the semantic web made easy. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 213–217, 2017.

[21] Jovan Varga, Lorena Etcheverry, Alejandro A Vaisman, Oscar Romero, Torben Bach Pedersen, and Christian Thomsen. Qb2olap: enabling olap on statistical linked open data. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, pages 1346–1349. IEEE, 2016.

[22] Alberto Abelló, Oscar Romero, Torben Bach Pedersen, Rafael Berlanga, Victoria Nebot, Maria Jose Aramburu, and Alkis Simitsis. Using semantic web technologies for exploratory olap: a survey. *IEEE transactions on knowledge and data engineering*, 27(2):571–588, 2014.

[23] Nurefşan Gür, Torben Bach Pedersen, Katja Hose, and Mikael Midtgaard. Multidimensional enrichment of spatial rdf data for solap. *Semantic Web*, (Preprint):1–35, 2020.

[24] Zhuoyi Wang, Yigong Wang, Yu Lin, Evan Delord, and Khan Latifur. Few-sample and adversarial representation learning for continual stream mining. In *TheWebConf*, pages 718–728, 2020.

[25] Zhuoyi Wang, Yuqiao Chen, Chen Zhao, Yu Lin, Xujiang Zhao, Hemeng Tao, Yigong Wang, and Latifur Khan. Clear: Contrastive-prototype learning with drift estimation for resource constrained stream mining. In *TheWebConf*, pages 1351–1362, 2021.

[26] Graham Cormode and Minos N. Garofalakis. Histograms and wavelets on probabilistic data. In *ICDE*, 2009.

[27] Robert Fink, Larisa Han, and Dan Olteanu. Aggregation in probabilistic databases via knowledge compilation. In *VLDB*, 2012.

[28] Graham Cormode, Feifei Li, and Ke Yi. Semantics of ranking queries for probabilistic data and expected ranks. In *ICDE*, 2009.

[29] Jian Li, Barna Saha, and Amol Deshpande. A unified approach to ranking in probabilistic databases. In *VLDB*, 2009.

[30] Xiang Lian and Lei Chen. Efficient processing of probabilistic reverse nearest neighbor queries over uncertain data. *VLDB J.*, 18(3):787–808, 2009.

[31] Jian Pei, Bin Jiang, Xuemin Lin, and Yidong Yuan. Probabilistic skylines on uncertain data. In *VLDB*, 2007.

[32] Xike Xie, Kai Zou, Xingjun Hao, Torben Bach Pedersen, Peiquan Jin, and Wei Yang. OLAP over probabilistic data cubes II: parallel materialization and extended aggregates. *TKDE*, 32(10):1966–1981, 2020.

[33] Matteo Brucato, Nishant Yadav, Azza Abouzied, Peter J Haas, and Alexandra Meliou. Stochastic package queries in probabilistic databases. In *SIGMOD*, pages 269–283, 2020.

[34] Antoine Amarilli and İsmail İlkan Ceylan. A dichotomy for homomorphism-closed queries on probabilistic graphs. In *ICDT*, 2020.

[35] Weiyuan Wu, Lampros Flokas, Eugene Wu, and Jiannan Wang. Complaint-driven training data debugging for query 2.0. In *SIGMOD*, pages 1317–1334, 2020.

[36] Amelie Chi Zhou, Weilin Xue, Yao Xiao, Bingsheng He, Shadi Ibrahim, and Reynold Cheng. Taming system dynamics on resource optimization for data processing workflows: A probabilistic approach. *TPDS*, 33(1):231–248, 2022.

[37] Bhargav Kanagal and Amol Deshpande. Efficient query evaluation on temporally correlated probabilistic streams. In *ICDE*, pages 1315–1318, 2009.

[38] Graham Cormode and Minos Garofalakis. Sketching probabilistic data streams. In *SIGMOD*, pages 281–292, 2007.

[39] Christopher Ré, Julie Letchner, Magdalena Balazinksa, and Dan Suciu. Event queries on correlated probabilistic streams. In *SIGMOD*, pages 715–728, 2008.

[40] Rajeev Alur, Yu Chen, Kishor Jothimurugan, and Sanjeev Khanna. Space-efficient query evaluation over probabilistic event streams. In *LICS*, pages 74–87, 2020.

[41] Yongxin Tong, Xiaofei Zhang, and Lei Chen. Tracking frequent items over distributed probabilistic data. *WWW*, 19(4):579–604, 2016.

[42] Effective and efficient top-k query processing over incomplete data streams. *Information Sciences*, 544:343–371, 2021.

[43] Xike Xie, Hua Lu, and Torben Bach Pedersen. Efficient distance-aware query evaluation on indoor moving objects. In *ICDE*, pages 434–445, 2013.

[44] Xike Xie, Hua Lu, and Torben Bach Pedersen. Distance-aware join for indoor moving objects. *IEEE Trans. Knowl. Data Eng.*, 27(2):428–442, 2015.

[45] Reynold Cheng, Xike Xie, Man Lung Yiu, Jinchuan Chen, and Liwen Sun. UV-diagram: A voronoi diagram for uncertain data. *ICDE*, 2010.

[46] Xike Xie, Reynold Cheng, Man Lung Yiu, Liwen Sun, and Jinchuan Chen. UV-diagram: A voronoi diagram for uncertain spatial databases. *VLDBJ*, 2013.

[47] Xike Xie, Man Lung Yiu, Reynold Cheng, and Hua Lu. Scalable evaluation of trajectory queries over imprecise location data. *TKDE*, 2013.

[48] Xike Xie, Benjin Mei, Jinchuan Chen, Xiaoyong Du, and Christian S. Jensen. Elite: an elastic infrastructure for big spatiotemporal trajectories. *VLDB J.*, 25(4):473–493, 2016.

[49] Asif Iqbal Baba, Manfred Jaeger, Hua Lu, Torben Bach Pedersen, Wei-Shinn Ku, and Xike Xie. Learning-based cleansing for indoor RFID data. In *SIGMOD*, pages 925–936, 2016.

[50] Reynold Cheng, Eric Lo, Xuan S. Yang, Ming-Hay Luk, Xiang Li, and Xike Xie. Explore or exploit? effective strategies for disambiguating large databases. *Proc. VLDB Endow.*, 3(1):815–825, 2010.

[51] Jiawei Han, Yixin Chen, Guozhu Dong, Jian Pei, Benjamin W Wah, Jianyong Wang, and Y Dora Cai. Stream cube: An architecture for multi-dimensional analysis of data streams. *DPD*, 18(2):173–197, 2005.

[52] Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. Quickinsights: Quick and automatic discovery of insights from multi-dimensional data. In *SIGMOD*, pages 317–332, 2019.

[53] Pingchuan Ma, Rui Ding, Shi Han, and Dongmei Zhang. Metainsight: Automatic discovery of structured knowledge for exploratory data analysis. In *SIGMOD*, pages 1262–1274, 2021.

[54] Zhe Wang, Nivan Ferreira, Youhao Wei, Aarthy Sankari Bhaskar, and Carlos Scheidegger. Gaussian cubes: Real-time modeling for visual exploration of large multidimensional datasets. *TVCG*, 23(1):681–690, 2017.