Fast Algorithm for Computing Weighted Projection Quantiles, Quantile Regression and Data Depth for High-Dimensional Large Data Clouds

Ujjal Kumar Mukherjee
Carlson School of Management
University of Minnesota
Minneapolis, Minnesota 55455, USA
Email: mukh0067@umn.edu

Snigdhansu Chatterjee
School of Statistics
University of Minnesota
Minneapolis, Minnesota 55455, USA
Email: chatterjee@stat.umn.edu

***Abstract*—In this paper we present a new algorithm based on a weighted projection quantiles for fast and frugal real time quantile estimation of large sized high dimensional data clouds. We present a projection quantile regression algorithm for high dimensional data. Second, we present a fast algorithm for computing the depth of a point or a new observation in relation to any high-dimensional data cloud, and propose a ranking system for multivariate data. Third, we briefly describe a real time rapid monitoring scheme similar to statistical process monitoring, for actionable analytics with big data. We believe these algorithms would be very useful for real time analysis of high dimensional 'big data' sets including streaming data sets. The proposed algorithms would be of immense use in several application areas such as real time financial market analysis, real time remote health monitoring of patients using body area networked devices and real time pricing and inventory decisions in retail and manufacturing sector.**

Keywords: *'Big Data', weighted projection quantiles, quantile regression, data depth estimation, body area network, real time analysis, real time health monitoring*.

I. Introduction

In recent years, real time analytics of large streams of data is gaining popularity. In several application areas such as financial markets, healthcare, retail industry, and manufacturing, real time analytics is becoming an important tool for gaining competitiveness. With advancements in networking and sensor technologies like wearable mobile monitoring devices in health monitoring, radio-frequency identification (RFID) technology in manufacturing and retail, real time transaction and pricing data in financial markets and retail industry, and large volume of call data in telecommunications industry, demand for real time analytics of large streams of high dimensional data sets are becoming more and more ubiquitous.

One common usage of real time analytics on large volume of data is in real time health monitoring using wearable or implantable devices using wireless broadband communication and *General Packet Radio Service* (GPRS) technology. Usually many of these devices consist of a network of sensors and bio detectors that are connected via bluetooth to a radio device or a cell phone that transmits data to a central server on a real time basis ([11], [12]). Some of the bio-parameters that are measured on a real time basis are heart diastolic and systolic pressure, blood glucose levels, blood oxygen levels, heart rate, and other blood chemistry and cardiac parameters. Data from a large number of devices fitted on individual patients are received by servers on a real time basis. Large volume of data gets generated at high velocity and variety at the server. The server implements analytics to detect anomalies in these parameters and alert patients in real time for proper and timely medical intervention. The use of such technology is likely to increase exponentially in near future.

Often, from an analysis point of view, what is of interest is the detection of anomalies and abnormalities beyond usual range of these parameters. These anomalies and abnormalities act as signals for health related risk of individual patients. Note that standard statistical analysis using means, variances, correlations and simple linear regressions are unsuitable for answering questions that are of relevance to these genre of problems. As opposed to these off-the-shelf statistical techniques that are pervasive in data mining and other big data analysis techniques, for the class of problems described briefly above, we need (i) measures of non-centrality, (ii) relationships among multiple variables that can predict non-central tendency of a group of variables, (iii) actionable

techniques for detection of anomalies, as opposed to routine non-centrality. The nature of data that we are concerned with here is typically high dimensional, and typically arrives over a long span of time. In order to address the three kind of data analytic features mentioned above, we need measures of non-centrality in a multivariate context. Moreover, this measure has to be computable in real time, and has to retain sufficient mathematical and statistical fidelity in order to produce meaningful answers.

In this paper, we propose the best way of addressing these challenges is to consider *high-dimensional quantile estimation and inference* techniques. The notion of a multivariate quantile is non-trivial, even in two dimensions. This is because the standard notion of a quantile as the inverse function of a cumulative probability distribution function has no equivalent in dimensions greater than one. Several attempts to generalize this notion to multivariate random variables have been made in the last few decades, but these have generally failed to retain even some of the simplest and most desirable properties of the standard univariate quantile.

The first successful attempt in generalizing the notion of quantiles to arbitrary dimensional data may be found in [9], called *spatial quantile*. There, the spatial quantile was defined as a multivariate optimization problem, which matches with an equivalent property that is known for univariate quantiles. Several other necessary theoretical properties of the spatial quantile was obtained in that paper, or in later developments [4], [5], [6], [7], [8], [1], [2], [3]. However, all these developments suffer from two deficiencies that render them unusable for big data analysis, or for the case when there is a constant stream of high-dimensional data arriving. First, the computation of these spatial quantile variants, and essentially all other multivariate generalizations of quantiles which may not even have a mathematically sound foundation, are computationally intense and challenging. For example, several of these are iterative optimization schemes that require the entire data to be repeatedly processed till a desired level of convergence is reached. The second and perhaps more important deficiency is that these quantiles are generally not even defined for cases where the dimension is larger than the sample size, or sometimes even when the dimension is large.

Thus, while the kind of problems we are interested in require multivariate quantiles, existing multivariate quantiles are unsuitable. The estimation of the quantiles of the data distribution need to be on a real time basis and hence fast for meaningful usage of these real time health monitoring systems.

In this paper we present a very fast algorithm for approximate estimating of quantiles for high volume and high dimensional data streams on a continuous realtime basis, building on a recent work of [10]. The algorithm is based on *projection quantiles* and is called weighted search algorithm for estimating quantiles of high dimensional data. This algorithm is useful in fast and approximate estimation of quantiles in high dimensional high volume data where traditional optimization based methods would not suffice.

Our next algorithm has a dual purpose. First, given a cloud of high-dimensional observations and a specific point (which may be one of the observations), to quantify how deeply the specific point is embedded in the cloud. Thus, we quantify the *depth* of any given point with respect to a data cloud, which informs us whether the point is near the center of the data or in one of the more outlying regions. Second, carrying forward this analysis, we present a method for ranking multivariate observations. We recognize there can be no unique way of sorting or ordering multivariate data, but the present scheme is nevertheless deemed useful.

We also propose an algorithm that is similar to statistical process control and online monitoring, as an actionable decision making tool, where big data quantile-based tools may be used as inputs.

We present simulated and real data examples, and a concluding section lastly.

II. Projection Quantiles

In this paper, we will denote the open unit ball in p-dimensional Euclidean plane as $\mathcal{B}_p = \{x \in \mathbb{R}^p : ||x|| < 1\}$. The notation $||\mathbf{a}||$ stands for the Euclidean norm of a vector $\mathbf{a}$, while $< \mathbf{a}, \mathbf{b} >$ stands for the Euclidean inner product between two vectors. For convenience, we reserve the notation $\mathbf{0}$ for a vector of zeroes, and $\mathbf{1}$ for a vector of ones, in appropriate dimensions that will be specified in the right contexts. Also, we reserve the notation $\mathbf{u}$ to denote a typical element in this open unit ball. We will further reserve the notation $\mathbf{U}$ for the unit vector in the direction of $\mathbf{u} \in \mathcal{B}^p$ when $\mathbf{u} \neq \mathbf{0} \in \mathbb{R}^p$ and $\mathbf{0}$ otherwise.

Let $\mathbf{X} \in \mathbb{R}^p$ be a random variable in p dimensional space. The projection quantile has been defined in [10] as a function indexed by the unit ball in $\mathbb{R}^p$. Let $\mathbf{U}$ denote the unit vector in the direction of $\mathbf{u} \in \mathcal{B}^p$, thus $\mathbf{U} = \frac{\mathbf{u}}{||\mathbf{u}||}$. We define

$$\mathbf{X_U} = \langle \mathbf{X}, \mathbf{U} \rangle = \langle \mathbf{X}, \frac{\mathbf{u}}{||\mathbf{u}||} \rangle = ||\mathbf{u}||^{-1} \langle \mathbf{X}, \mathbf{u} \rangle.$$

For the moment, assume that the center of the distribution of $\mathbf{X}$ is the origin. The projection of $\mathbf{X}$ in the direction of $\mathbf{u}$ is, $\mathbf{X_U U} = ||\mathbf{u}||^{-2}\langle \mathbf{X}, \mathbf{u}\rangle \mathbf{u}$.

Let, $\mathbf{q_U}$ be the $(1+||\mathbf{u}||)/2$-th quantile of $\mathbf{X_U}$, i.e., $\mathbb{P}[\mathbf{X_U} \leq \mathbf{q_u}] = (1 + ||\mathbf{u}||)/2$. The $\mathbf{u}$-th projection quantile is defined as,

$$Q_{proj}(\mathbf{u}) = \mathbf{q_u}\frac{\mathbf{u}}{||\mathbf{u}||} = \mathbf{q_u U}.$$

The advantage of projection quantile is that the estimation of the quantile is linearly dependent on the number of dimensions p through the matrix multiplication operation in calculation of $\mathbf{X_U}$. Note that the data cloud $\mathbf{X}_1, \ldots, \mathbf{X}_n$ needs to be centered to apply the above technique for computing the projection quantiles. The co-ordinatewise median (the 0.5 quantile) of the data cloud acts as a good choice of the center.

III. Weighted Projection Quantile

The random vector $\mathbf{X}$ can be decomposed into sum of orthogonal components in the direction of $\mathbf{u}$ and along a hyperplane orthogonal to $\mathbf{u}$. Thus $\mathbf{X} = \mathbf{X_U U} + \mathbf{X_U}\perp$. Hence by Pythagoras Theorem we have $||\mathbf{X}||^2 = ||\mathbf{X_U}||^2 + ||\mathbf{X_U}\perp||^2$.

In estimating the projection quantiles described above we ignored the information on the orthogonal plane to $\mathbf{U}$. This can bring inaccuracy in the estimation. A data point $\{x\}$ may be sufficiently informative when $||\mathbf{X_U}\perp|| \ll ||\mathbf{X_U}||$. However, for complex data clouds this may not be the case with all data points. So, we propose a weighted estimation scheme for estimation of projection quantiles. We assign a weight to each point in the data cloud. For data points where $||\mathbf{X_U}||$ is relatively larger than the orthogonal component we assign a higher weight. This is done by assigning a weight which is a function of the proportional information of a data point in the direction of $\mathbf{u}$ for the purpose of quantile estimation. Hence

$$w_x = f\left(\frac{||\mathbf{x_U}||}{||\mathbf{x}||}\right).$$

There are several choices of the weight function. Below is a list of a few. $f(x) = x$ (linear), $f(x) = \exp(x)$ (exponential), $f(x) = \frac{1}{\sqrt{2\pi}}\exp\{-\frac{x^2}{2}\}$ (gaussian), $f(x) = \mathbb{I}_{\{|x|\leq\delta\}}$ (rectangular) for a suitable choice of δ, or a combination of the above such as $f(x) = \frac{1}{\sqrt{2\pi}}\exp\{-\frac{x^2}{2}\}\mathbb{I}_{\{|x|\leq\delta\}}$.

The conditions required for the weight function is that it should be an increasing function with decreasing proportion of the orthogonal component from the orthogonal decomposition of $\mathbf{X}$ described earlier. The choice of the specific weight function may depend on the data. A scheme for choice of the right weight function is to maximize the total weights associated with the chosen points as the $\mathbf{u}$-th quantile. Higher the total weight associated with the quantile function, the less is the deviation of the estimated quantile from the true quantile.

IV. Weighted Search Algorithm of the Projection Quantile

In estimating the projection quantile we order the projections of the data points on the vector $\mathbf{U}$ and choose the $(1+||\mathbf{u}||)/2 * n$-th data point. In doing so there is a possibility that we may choose a point $\mathbf{x}_p$ with relatively low associated weight, whereas, choosing the $x_{p\pm\alpha}$-th point (for small α) with higher associated weight may have been a much better choice in terms of the quantile estimation bias induced by the choice of a specific point. Let us assume that we choose the $x_{p\pm\alpha}$ as the best estimate of the quantile in the direction of $\mathbf{U}$. This will induce an error in the projection quantile estimation. There are two sources of errors in the estimation of the quantile function. First, instead of estimating the $(1 + ||\mathbf{u}||)/2$-th quantile, we may be estimating the $(1 + ||u|| \pm \gamma)/2$-th quantile, where γ represents the magnitude of error in not choosing the $\mathbf{u}$-th quantile. Second, the weight assigned to each of the data points in $\mathbf{X}$ is a measure of the information error (information in the orthogonal plane that is missed out) in estimation of the projection quantile. The weight $(1 - w)$ measures the amount of information we are unable to capture in choosing the fast projection quantile estimation instead of a full optimization based algorithm. However, to achieve the best estimate we would need to jointly minimize the two errors γ and w. The objective function of the weighted search algorithm can be formally stated as,

$$\min\left\{\gamma + \frac{||X_U\perp||}{||X||}\right\} \Rightarrow \quad \min\left\{\gamma + \frac{||X|| - ||X_U||}{||X||}\right\},$$
$$\Rightarrow \min\left\{\gamma - \frac{||X_U||}{||X||}\right\} \Rightarrow \qquad \min\{\gamma - w\}.$$

The minimization is obtained by a linear grid search over a vector of sorted projections and is not computationally difficult. Below we state the formal algorithm of the weighted projection quantile algorithm (algorithm 1).

V. Simulation

We ran several trials of the algorithm on simulated data. In figure 1 we show the simulation results for a un-weighted projection quantile estimate from several bivariate datasets. The left panel in this figure contains the displays of the standard projection quantile of [10], while the right panel consists of weighted projection quantiles. We used a combination of rectangular and gaussian weight function for the weighted projection quantiles. The figures show quantile estimates over

Algorithm 1 Weighted Search Algorithm for Projection Quantiles of Multidimensional Data Cloud

DATA: $X \in \mathbb{R}^p, \mathbf{u}$.
CREATE $U = \frac{u}{||u||}$.
CREATE $X_U = \langle X, U \rangle$.
CREATE PROJECTION VECTOR $X_P = \langle X, U \rangle U$.
CREATE WEIGHT VECTOR $w = f\left\{\frac{||X_U||}{||X||}\right\}$.
CREATE WEIGHTED PROJECTION VECTOR $X_{WP} = X_P * w$.
CREATE SORTED PROJECTION VECTOR $X_{WPS} = sort(X_{WP})$.
CREATE QUANTILE DEVIATION VECTOR $\gamma = 2 * POS(X_{WPS}) - 1 - ||u||$.
CREATE OBJECTIVE VECTOR $\gamma - w$.
SELECT QUANTILE $q_u = X_{WPS}[min(\gamma - w)]$.

a grid of vectors $\mathbf{u}$, with the specification that the norm $||\mathbf{u}||$ ranges from 0.15 to 0.95.

The various component displays in this figure demonstrate that due to the influence of data points with high orthogonal component (points away from the vector $\mathbf{u}$), the quantile estimates are distorted outwards with circular shapes outside the data cloud. This is specially visible in the higher quantiles. This is the over-estimation bias induced by the loss in information due to ignoring the orthogonal component of the data cloud. In displays e and g we can observe the butterfly like shapes forming around the data cloud and the quantile function is not able to estimate the shapes properly. Another problem is overestimation of the quantiles. Displays a and c show that the outer quantiles are substantially over estimated in most of the data range. Also, the projection quantile estimates at the lower quantiles leads to substantial "looping" of the estimated quantile functions, where the different choices of $\mathbf{u}$ with the same norm loops around the center several times.

Figures 1b and 1d show quantile estimates using a linear weight function ($w = \frac{||X_U U||}{||X||}$) and figures 1f and 1h show quantile estimates using a mixture of rectangular and gaussian weight function ($w = \frac{1}{\sqrt{2\pi}} \exp\{-\frac{(\frac{||X_U U||}{||X||})^2}{2}\}\mathbb{I}_{\{|\frac{||X_U U||}{||X||}| \leq \delta\}}$). As we can see from the figures, the linear weight function takes care of the overestimation problem and estimates the quantiles much more correctly than the normal projection quantiles. The weighting scheme also takes care of the looping and shape distortion issues to a large extent. The estimation of the shape of data cloud in figure 1f and 1g with the mixture of rectangular and gaussian weight function is much more accurate than the normal projection quantile. However, due to the discontinuous nature of the weight function at the boundaries, the quantile function estimation has substantial variance and do not give a continuous quantile curve estimate. For practical purposes related to inference, this may be a problem. However, this problem can be addressed by using a full gaussian weight function or by incorporating further restrictions related to the steps in quantile function.

VI. A new kind of Rank or Depth

We first define a vector of indicators in $\mathbb{R}^p$ as follows: let $\boldsymbol{\epsilon}$ be a vector in $\mathbb{R}^p$, where each ϵ_i is either a 1 or a 0. Clearly, there are 2^p such vectors. We further define the notation

$$>^1 \equiv \quad >, \text{ and } >^0 \equiv \quad \leq .$$

We define the p-dimensional notation $\leq^{\boldsymbol{\epsilon}}$ co-ordinatewise, which will be made apparent below.

All vectors for us are column vectors. We use the notation a^T to denote the transpose of a vector or matrix. Suppose $\mathbf{X} = (X_1, \ldots, X_p)^T \in \mathbb{R}^p$ is a p-dimensional random variable with distribution F, and let $\mathbf{x} = (x_1, \ldots, x_p)^T \in \mathbb{R}^p$ be an arbitrary point. For $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_p) \in \mathbb{R}^p$ where each ϵ_i is either 0 or 1, define

$$\begin{aligned} \eta(\mathbf{x}; F, \boldsymbol{\epsilon}) &= \mathbb{P}\left[\mathbf{X} >^{\boldsymbol{\epsilon}} \mathbf{x}\right] \\ &= \mathbb{P}\left[X_1 >^{\epsilon_1} x_1, \ldots, X_p >^{\epsilon_p} x_p\right]. \end{aligned}$$

For example, if $p = 2$, and $\boldsymbol{\epsilon} = (0, 0)^T$, then

$$\begin{aligned} \eta(\mathbf{x}; F, \boldsymbol{\epsilon}) &= \mathbb{P}\left[X_1 >^{\epsilon_1} x_1, X_2 >^{\epsilon_2} x_2\right] \\ &= \mathbb{P}\left[X_1 \leq x_1, X_2 \leq x_2\right] = F(x_1, x_2). \end{aligned}$$

On the other hand, for $p = 2$ and $\boldsymbol{\epsilon} = (0, 1)^T$, we have

$$\begin{aligned} \eta(\mathbf{x}; F, \boldsymbol{\epsilon}) &= \mathbb{P}\left[X_1 >^{\epsilon_1} x_1, X_2 >^{\epsilon_2} x_2\right] \\ &= \mathbb{P}\left[X_1 \leq x_1, X_2 > x_2\right]. \end{aligned}$$

We use the notation $|\boldsymbol{\epsilon}| = \sum \epsilon_i$ to denote the number of ones in $\boldsymbol{\epsilon}$.

When we have a collection of n observations $\mathbf{X}_1, \ldots, \mathbf{X}_n$, construct the *empirical distribution function* $\mathbb{F}_n(\mathbf{x}) = n^{-1} \sum_{i=1}^n I_{\{\mathbf{X}_i \leq \mathbf{x}\}}$. Then, the *sample* version of η as $\eta(\mathbf{x}; \mathbb{F}_n, \boldsymbol{\epsilon})$, and for reasons of simplicity, we consider $n_{\boldsymbol{\epsilon}}(\mathbf{x}) = n\eta(\mathbf{x}; \mathbb{F}_n, \boldsymbol{\epsilon})$ as a fundamental object to work with. This is simply the number of data points that is present in the quadrant given by $\boldsymbol{\epsilon}$ with origin $\mathbf{x}$.

Define the 2^p-dimensional vector $\mathbf{n}(\mathbf{x})$, which lists the number of observations in each quadrant. We consider the vector in standard ordering scheme in terms of $\boldsymbol{\epsilon}$. Fro example, for $p = 3$, $\mathbf{n} = (n_{000}, n_{001}, n_{010}, n_{100}, n_{011}, n_{101}, n_{110}, n_{111})^T$. It can be seen that $\mathbf{n}(\mathbf{x})$ is really a multinomial vector.

We consider the point $\mathbf{x}$, for which $\mathbb{V}(\mathbf{n}(\mathbf{x}))$ is zero (or minimum). In case there are multiple such points, choose the co-ordinatewise minimum among them. This happens when each $n_{\boldsymbol{\epsilon}}(\mathbf{x}) = 2^{-p}n$. Also note that the co-ordinatewise median satisfies this definition.

The maximum possible value of $\mathbb{V}(\mathbf{n}(\mathbf{x}))$ is $0.75n^2$. This is achieved when one of the $n_{\boldsymbol{\epsilon}}(\mathbf{x})$ values is n, the rest is zero.

We define the depth of a point $\mathbf{x}$ as $\delta(\mathbf{x}) = \exp(-\mathbb{V}(\mathbf{n}(\mathbf{x}))$. Thus, the median from our definition is also the point of maximum depth.

We define the "rank" of a data point as a function, indexed by a parameter $\alpha \in [0, 1]$. We define is as follows:

$$\begin{aligned} & R(\mathbf{x}; \alpha) \\ = & \left[\sum_{\boldsymbol{\epsilon}} n_{\boldsymbol{\epsilon}}(\mathbf{x}) I_{\{|\boldsymbol{\epsilon}|\in\{0,n\}\}} + \alpha \sum_{\boldsymbol{\epsilon}} n_{\boldsymbol{\epsilon}}(\mathbf{x}) I_{\{|\boldsymbol{\epsilon}|\in(0,n)\}}\right]^{-1} \\ & n_{\boldsymbol{\epsilon}}(\mathbf{x}) I_{\{|\boldsymbol{\epsilon}|=0\}}. \end{aligned}$$

Consider the case $p = 2$. Then the above is

$$R(\mathbf{x}; \alpha) = \frac{n_{(0,0)}(\mathbf{x})}{n_{(0,0)}(\mathbf{x}) + \alpha n_{(0,1)}(\mathbf{x}) + \alpha n_{(1,0)}(\mathbf{x}) + n_{(1,1)}(\mathbf{x})}.$$

The crucial idea here is that we are defining the depth and rank with a method that considers $\mathbf{x}$ as the "center of the universe", and asks a "what if" question.

A. Algorithm

The major stumbling block in implementing the above is that we require 2^p computations for each point of interest $\mathbf{x}$, since the above scheme essentially treats $\mathbf{x}$ as the origin, and considers the number of observations in each of the 2^p quadrants from that origin.

The follow algorithm is a fast way of getting the same, assuming 2^p parallel processes can be run. Define the random vector $\mathbf{X}_{\boldsymbol{\epsilon}}$ as follows:

$$\mathbf{X}_{\boldsymbol{\epsilon},i} = \begin{cases} X_i & \text{if } \epsilon_i = 1, \text{ and} \\ -X_i + 10^{-8} & \text{if } \epsilon_i = 0. \end{cases}$$

We define a non-random value transformation almost identically:

$$\mathbf{x}_{\boldsymbol{\epsilon},i} = \begin{cases} x_i & \text{if } \epsilon_i = 1, \text{ and} \\ -x_i & \text{if } \epsilon_i = 0. \end{cases}$$

Notice that there are 2^p such $\mathbf{X}_{\boldsymbol{\epsilon}}$ (or $\mathbf{x}_{\boldsymbol{\epsilon}}$) values. We assume these 2^p random vectors can be analyzed in parallel. Now notice that we can obtain each co-ordinate of $\mathbf{n}(\mathbf{x})$ by computing $\mathbb{P}\left[\mathbf{X}_{\boldsymbol{\epsilon}} \leq \mathbf{x}_{\boldsymbol{\epsilon}}\right]$ for each possible choices of $\boldsymbol{\epsilon}$.

The additional adjustment factor of 10^{-8} in $\mathbf{X}_{\boldsymbol{\epsilon}}$ is just for technicalities, since we had $\leq$ on one side and $>$ on the other. This may be ignored for all practical purposes.

VII. Real time analytics and decision making

We convert each new incoming high dimensional observation to a number α_i between 0 and 1. This could be the depth of the observation, the depth of the residual after fitting a PQLMR, or the rank of the observation of the PQLMR residual. It could be a much more involved quantity, where weighted projection quantiles, depths and ranks are involved in multiple stages. Then this single number is converted to $Z_i = \Phi^{-1}(\alpha_i)$ to project it on the entire real line. We then compute the recursive statistic:

$$C_n = \max\{C_{n-1} + Z_i - \delta, 0\},$$

and signal a "distress" if $C_n > L_1$ and an "outage" if $C_n > L_2$. The thresholds L_1 and L_2 are set according to the required probability bounds. The constant δ is a tuning constant, so that there are not too many false signals.

VIII. Application Example

We present one application example for the use of weighted quantile estimation in practice. The example is pertaining to a wearable networked sensor based computing platform designed for monitoring a variety of day-to-day motion related activities and to be used as a 24/24h digital personal assistant. A network of several accelerometers are fitted in the human body and the sensors collect various data related to daily activities and movements. Activity recognition is a fast emerging science which has wide application in research, medicine, health care, surveillance, human behavior, human machine interaction and remote patient monitoring [14]. Early research on activity recognition focussed on complex audio visual data. However, with rapid advancement in sensor technology, accelerometers have widely and rapidly gained popularity due to their small size, low power requirement, non-intrusiveness and capacity to provide direct measurements of movement components with respect to a reference frame.

Identifying human activities depends on the way motion data is collected and reported by accelerometers. Most accelerometers report motion by decomposing motion into three orthogonal components on the cartesian coordinate axes system. The classification of various motion and activities can be done using a combined index of these motion components. One common parameter that is used for classifying changes in human motion and activity levels is the combined acceleration component from the three axes. If a_x, a_y, and a_z are the three components of the accelerometer data then the combined acceleration is give by $a = \sqrt{a_x^2 + a_y^2 + a_z^2}$. Small differences and known patterns of variations of the combined acceleration

component can help classify various human activities [14]. The accelerometers transmit data on a real time basis and often the extreme quantiles of the data distribution provide information about the movement pattern. Also, since the data velocity is very high it is important to calculate the quantiles of the streaming data distribution using a fast algorithm that can calculate the quantiles in a given direction on a real time basis.

The data we analyze here is taken from the University of California, Irvine's Machine Learning Repository database (https://archive.ics.uci.edu/ml/datasets). The database consists of data from fifteen persons with wearable accelerometers. We analyzed the data from one person's accelerometer data. The data consists of 162K observations. The variables are X Acceleration, Y Acceleration and Z Acceleration. We would like to emphasize that our methods described above can be used in real time for much higher dimensional data and with many more observations, and this example is for illustration only.

We computed weighted fast projection quantile ($q = 0.75$). We analyzed based on three schemes. Scheme 1: we took a block of 500 initial datapoints. We computed the 0.75 quantile of the initial 500 block of data using $u = (0, 0, 0.5)$ which corresponds to $q = 0.75$ in the Z direction. This is because the Z acceleration is the most important and informative component in motion recognition. However, we could have taken any other direction depending on the application and specific requirement. Then we computed the quantile values for the data. Then we successively added chunks of 50 data points till the complete data set is included in the quantile calculation. The output of the quantile estimation was done in R and is shown in figure 3. A quantile index of the combines X, Y, and Z accelerations were calculated and shown. Scheme 2: data was taken in blocks of 500. After the initial block was analyzed, 50 new data points were added and 50 oldest data points were discarded. So the block of data remained to be 500 for every run. The output of this run is shown in figure 4. Scheme3: chunks of subsequent data blocks of size 100 were taken and analyzed. The output of this analysis is shown in figure 5.

As we can see all the schemes fairly estimate the change in data pattern. Many more schemes of analysis can be devised based on specific requirements. Each of the analysis were almost instantaneous and took virtually seconds to execute. However, by parallelizing the program and running in a distributed multiple core environment can lead to much faster real time analysis of really large data streams. We believe that these algorithms can help analyse fast large data streams with high velocity and variety on a real time basis.

IX. Conclusion

We have proposed a fast algorithm for computing weighted projection quantiles of high dimensional high volume data for real time data analysis. We also extended the algorithm to a proposed projection quantile regression and data depth calculation for high dimensional data based on empirical distribution function. We have also, provided several simulated and real life examples to illustrate the use of the algorithms. We believe that the proposed algorithms will be extremely useful in practice for analysis of streaming real time data.

References

[1] Chakraborty, B. On multivariate median regression. *Bernoulli*, **5** (1999), no 4, 683–703.

[2] Chakraborty, B. On affine equivariant multivariate quantiles. *Ann. Inst. Statist. Math.* **53** (2001), no. 2, 380–403.

[3] Chakraborty, B. On multivariate quantile regression. *J. Statist. Plann. Inference* **110** (2003), no. 1-2, 109–132.

[4] Chakraborty, B., and P. Chaudhuri On a transformation and re-transformation technique for constructing an affine equivariant multivariate median. *Proc. Amer. Math. Soc.* **124** (1996), no. 8, 2539–2547.

[5] Chakraborty, B., and P. Chaudhuri On an adaptive transformation-retransformation estimate of multivariate location. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **60** (1998a), no. 1, 145–157.

[6] Chakraborty, B., and P. Chaudhuri On multivariate rank regression. *L_1-statistical procedures and related topics (Neuchâtel, 1997)*, 399–414, IMS Lecture Notes Monogr. Ser., **31**, Inst. Math. Statist., Hayward, CA, 1997.

[7] Chakraborty, B., and P. Chaudhuri On affine invariant sign and rank tests in one- and two-sample multivariate problems. *Multivariate analysis, design of experiments, and survey sampling*, 499–522, Statist. Textbooks Monogr., **159**, Dekker, New York, 1999.

[8] Chakraborty, B., and P. Chaudhuri A note on the robustness of multivariate medians. *Statist. Probab. Lett.* **45** (1999), no. 3, 269–276.

[9] Chaudhuri, P., (1996), On a geometric notion of quantiles for multivariate data, *J. Amer. Statist. Assoc.*, **91** (434), 862–872.

[10] (N. Mukhopadhyay and Chatterjee, S.) (2011), High dimensional data analysis using multivariate generalized spatial quantiles, *J. Mult. Anal.*, **102** (4), 768–780.

[11] Oliver, Nuria, and Fernando Flores-Mangas. "HealthGear: a real-time wearable system for monitoring and analyzing physiological signals." *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on.* IEEE, 2006.

[12] Pantelopoulos, Alexandros, and Nikolaos G. Bourbakis. "A survey on wearable sensor-based systems for health monitoring and prognosis." *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 40.1* (2010): 1-12.

[13] Koenker, R. and Bassett, G. (1978) Regression quantiles, *Econometrica*, **46**, 33-50.

[14] Casale, Pierluigi, Oriol Pujol, and Petia Radeva. "Human activity recognition from accelerometer data using a wearable device." Pattern Recognition and Image Analysis. Springer Berlin Heidelberg, 2011. 289-296.

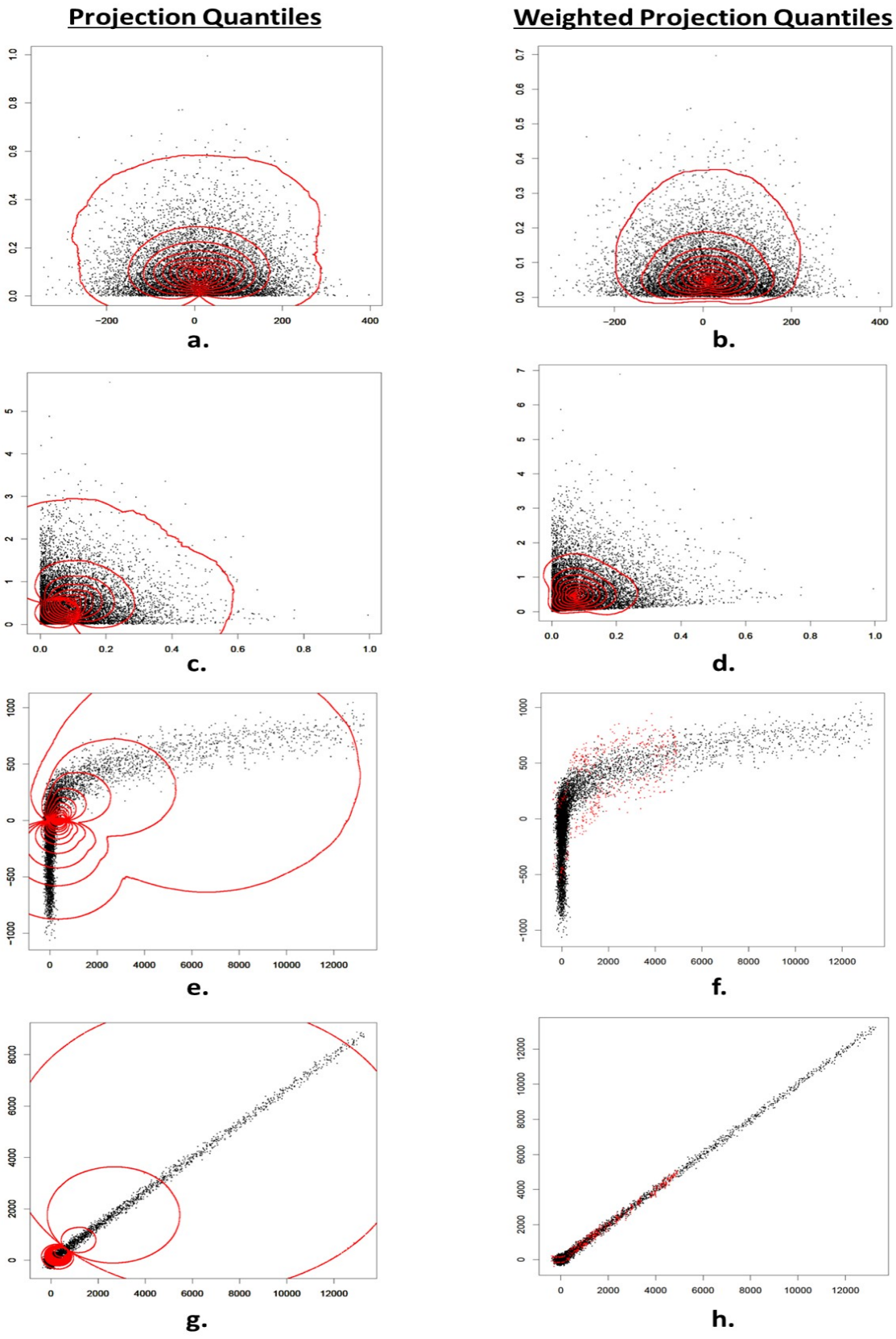

Fig. 1. **Comparison of usual projection quantiles with weighted projection quantiles**

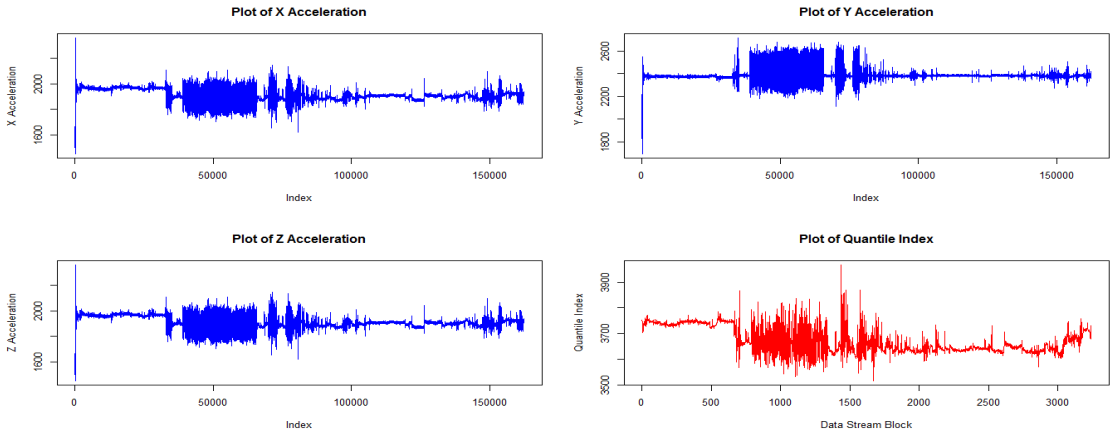

Fig. 2. **Quantile Index Estimation using Analysis Scheme 1**

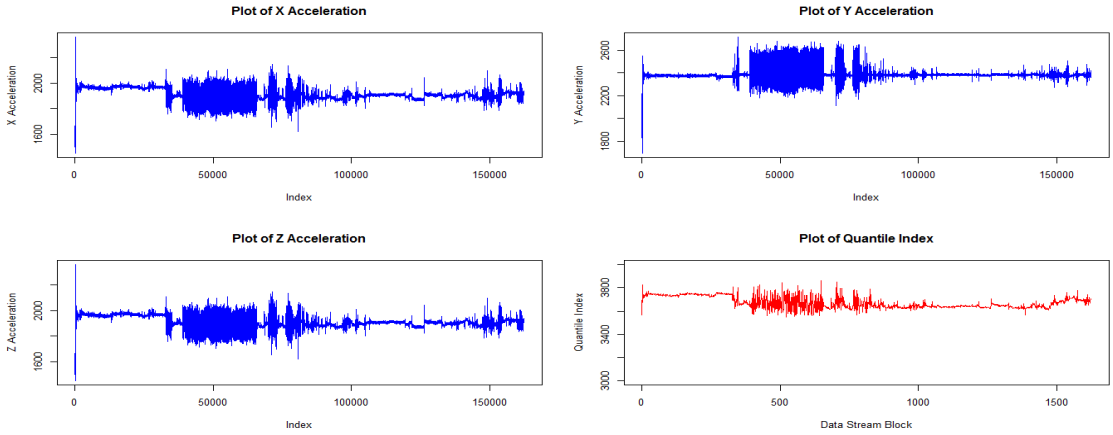

Fig. 3. **Quantile Index Estimation using Analysis Scheme 1**

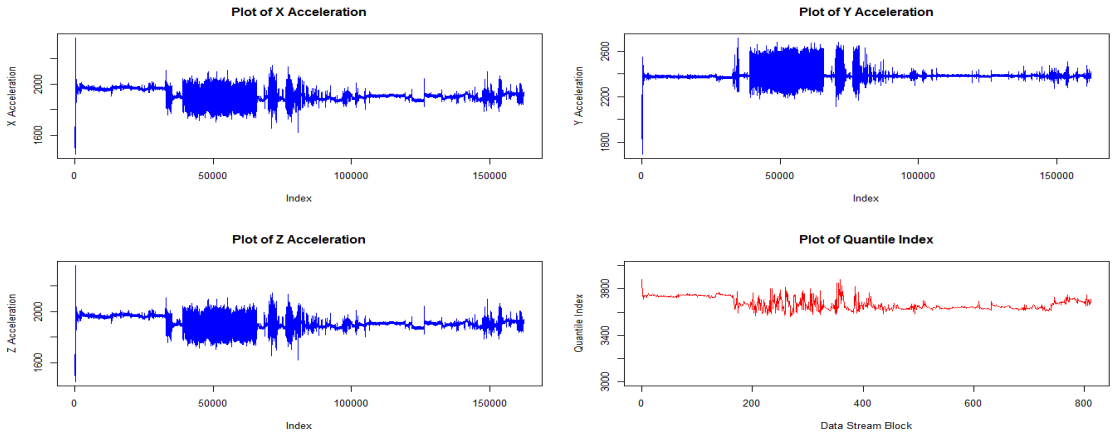

Fig. 4. **Quantile Index Estimation using Analysis Scheme 1**