EE 381V: Large Scale Learning	Spring 2013
Lecture 27 — April 23	
Lecturer: Caramanis & Sanghavi	*Scribe: Si Si*

27.1 Last Time: Matrix Multiplication in Streaming Model

- Review of streaming model.
- Matrix multiplication using sketching in streaming model.
- Upper and lower bounds of storage requirement for matrix multiplication using sketching.
- Sketching for regression.

27.2 This Time: Deal with the Corrupted Data in High-dimensional Setting

27.2.1 Background

There are many types of noises: stochastic noise, Gaussian noise, arbitrary noise or combinations of different types of noises. We may come across the corrupted data in many applications, for example, sparse estimation and matrix completion. In this lecture, we consider the standard setting in two important ways:

- a constant fraction of the points are arbitrarily corrupted in a perhaps non-probabilistic manner;
- the number of data points is of the same order as the dimensionality or perhaps considerably smaller than the dimensionality.

27.2.2 Basic Definition

Definition (Breakdown point(BDPT)): a robustness measure which is defined as the percentage of corrupted points that can make the output of the algorithm arbitrarily bad. Next we show some examples for BDPT. Given samples $\mathbf{x}_1, \cdots, \mathbf{x}_n$ with some fraction of $\mathbf{x}_1, \cdots, \mathbf{x}_n$ corrupted and some $\hat{\theta_n}$ that we want to estimate:

- If $\hat{\theta_n} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$, the BDPT is 0;

- If $\hat{\theta_n} = \hat{\theta}_{median}$, the BDPT 50%;
- If the $\hat{\theta_n}$ is λ trimmed mean, the BDPT is λ.

Setting: Given n data points $Y = \{\mathbf{y}_1, \cdots, \mathbf{y}_n\} \subset \mathbb{R}^p$, $\lambda(\lambda \leq 0.5)$ percentage of n points are corrupted points(outliers). Thus there are $t = (1-\lambda)n$ authentic samples $\mathbf{z}_1, \cdots, \mathbf{z}_t \in \mathbb{R}^p$, λn corrupted points $\mathbf{o}_1, \cdots, \mathbf{o}_{n-t} \in \mathbb{R}^p$ and these corrupted points are arbitrary. The data set we can observe is $Y = \{\mathbf{y}_1, \cdots, \mathbf{y}_n\} = \{\mathbf{z}_1, \cdots, \mathbf{z}_t\} \cup \{\mathbf{o}_1, \cdots, \mathbf{o}_{n-t}\}$

Objective: Given a mix of authentic and corrupted points Y, the goal is to find a low-dimensional subspace that captures as much variance of the authentic points. The corrupted points are arbitrary in every way except their number, which is controlled.

27.3 The Challenge for PCA in High-dimensional Setting

Next let us explain why robust PCA gets into trouble when dealing with the high-dimensional noisy data. Let us consider a simple generative model in high-dimensional setting, where the number of data points n is close to p.

Suppose each authentic point is generated as $\mathbf{y}_i = A\mathbf{x}_i + \mathbf{v}_i$, where A is a $p \times d$ matrix with d representing the number of principal components; each $\mathbf{x}_i$ is drawn from a zero mean symmetric random variable, and $\mathbf{v}_i \sim N(0, I_p)$. In the high-dimensional setting, we have $n \approx p \gg \sigma_A = \sigma_{max}(A)$ and thus is much larger than d. By the standard calculation, we have $\sqrt{E(\|A\mathbf{x}\|_2^2)} \leq \sqrt{d}\sigma_A$, and $\sqrt{E(\|\mathbf{v}\|_2^2)} \approx \sqrt{p}$, with sharp concentration of the Gaussian around this value. We then may have $\sqrt{E(\|\mathbf{v}\|_2^2)} \approx \sqrt{p} \leq \sqrt{d}\sigma_A$. So in this case, the magnitude of the noise is much larger than the magnitude of the signal.

In the high-dimensional setting, each point $\mathbf{y}_i$ might be perpendicular to all other points, and thus to the direction we want to recover. A simple experiment can verify the above observations.

- Generate the data points as $\mathbf{y}_i = \sqrt{\theta}\mathbf{v}x_i + \mathbf{w}_i$ with $\mathbf{w}_i \sim N(0, I_p)$;
- fix i and j and compute the angle between $\mathbf{y}_i$ and $\mathbf{y}_j(\frac{\mathbf{y}_i^T\mathbf{y}_j}{\|\mathbf{y}_i\|\|\mathbf{y}_j\|})$ and do the same thing for $\mathbf{v}$ and $\mathbf{y}_i$;
- find the angle between two closest points: $\mathbf{y}_i$ and $\mathbf{y}_j$ and the angle between $\mathbf{v}$ and closest $\mathbf{y}_i$;

27.4 The Main Algorithm: HR-PCA

High-dimensional Robust Principal Component Analysis(HR-PCA)[1] is efficient and robust to the outliers in the high-dimensional setting. HR-PCA can achieve BDPT 50% while others

Algorithm 1: HR-PCA

Input: Contaminated sample set $Y = \{\mathbf{y}_1, \cdots, \mathbf{y}_n\} \subset \mathbb{R}^p$, $\hat{d}$, $\bar{T}$, $\hat{t}$.
Output: $\bar{\mathbf{w}}_1, \cdots, \bar{\mathbf{w}}_{\hat{d}}$.
1) Let $\hat{\mathbf{y}}_i := \mathbf{y}_i$ for $i = 1, \cdots n$;$Y = \{\hat{\mathbf{y}}_1, \cdots, \hat{\mathbf{y}}_n\}$; $s := 0$; $Opt := 0$.
2) While $s \leq \bar{T}$, do
a) Compute the empirical variance matrix

$$\hat{\Sigma} := \frac{1}{n-s} \sum_{i=1}^{n-s} \hat{\mathbf{y}}_i \hat{\mathbf{y}}_i^T.$$

b) Perform PCA on $\hat{\Sigma}$. Let $w_1, \cdots, w_{\hat{d}}$ be the $\hat{d}$ principal components of $\hat{\Sigma}$.
c) If $\sum_{j=1}^{\hat{d}} \bar{V}_{\hat{t}}(\mathbf{w}_j) \geq Opt$, then let $Opt := \sum_{j=1}^{\hat{d}} \bar{V}_{\hat{t}}(\mathbf{w}_j)$ and let $\bar{\mathbf{w}}_j := \mathbf{w}_j$ for $j = 1, \cdots, \hat{d}$.
d) Randomly remove a point from $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s}$ according to
$\Pr(\hat{\mathbf{y}}_i \text{ is removed from } \hat{Y}) \propto \sum_{j=1}^{\hat{d}} (\mathbf{w}_j{}^T \hat{\mathbf{y}}_i)^2$.
e) Denote the remaining points by $\{\hat{\mathbf{y}}_i\}_{i=1}^{n-s-1}$.
f) $s := s + 1$.
3) Output $\bar{\mathbf{w}}_1, \cdots, \bar{\mathbf{w}}_{\hat{d}}$. End.

have the BDPT zero. Specifically, HR-PCA iteratively performs standard PCA, and at each iteration randomly casts out one point which is more likely to be corrupted point.

In HR-PCA, the trimmed variance is used as robust variance estimator(RVE):

$$\bar{V}_{\hat{t}}(\mathbf{w}) = \frac{1}{\hat{t}} \sum_{i=1}^{\hat{t}} |\mathbf{w}^T \mathbf{y}|^2_{(i)}.$$

where $\hat{t}$ is any lower-bound on the number of authentic points. The RVE works as first projecting $\mathbf{y}_i$ into the direction of $\mathbf{w}$, then removing the furthest $n - \hat{t}$, and finally computing the empirical variance for the rest $\hat{t}$ samples. By definition, RVE approximately measures the variance along a candidate direction $\mathbf{v}$ for all the authentic samples.

The main algorithm is shown in Algorithm 1.

There are some intuitions for HR-PCA. At each iteration, HR-PCA selects candidate directions using standard PCA to obtain directions with largest empirical variance. After obtaining the candidate directions $\mathbf{w}_1, \cdots, \mathbf{w}_d$, RVE will measure the variance of the $n - \hat{t}$ smallest points projected in those directions. If this is large, it means that many of the points have a large variance in this direction. In the other hand, if it is small, it is very likely that a number of the largest variance points are corrupted, and then it removes one of them randomly, in proportion to their distance in the candidate directions $\mathbf{w}_1, \cdots, \mathbf{w}_d$. So if the corrupted points have a very high variance along the candidate directions with large angle from the span of the principal components, then with higher probability, HR-

PCA will remove them; If they have a high variance in a direction "close to" the span of the principal components, then these points can only help in finding the "true" principal component; finally, if the corrupted points do not have a large variance, they may well survive the random removal process, but the distortion they can cause in the output of PCA is necessarily limited[1].

There are some interesting facts about HR-PCA:

- BDPT is 50%;
- There exists explicit lower bound for the error in [1];
- If the fraction of corrupted points go to 0, HR-PCA will recover the true solution.

Reference

[1] Huan Xu, Constantine Caramanis, Shie Mannor: Outlier-Robust PCA: The High-Dimensional Case. IEEE Transactions on Information Theory 59(1): 546-572 (2013)