Streaming Anomaly Detection Using Randomized Matrix Sketching*

Hao Huang
Stony Brook University,
haohuangcssbu@gmail.com

Shiva Kasiviswanathan
General Electric Global Research
kasivisw@gmail.com

Abstract

Timely and accurate detection of anomalies in massive data streams have important applications in preventing machine failures, intrusion detection, and dynamic load balancing, etc. In this paper, we introduce a new anomaly detection algorithm, which can detect anomalies in a streaming fashion by making only one pass over the data while utilizing limited storage. The algorithm uses ideas from matrix sketching and randomized low-rank matrix approximations to maintain an approximate low-rank orthogonal basis of the data in a streaming model. Using this constructed orthogonal basis, anomalies in new incoming data are detected based on a simple reconstruction error test. We theoretically prove that our algorithm compares favorably with an offline approach based on global singular value decomposition updates. The experimental results show the effectiveness and efficiency of our approach over other popular fast anomaly detection methods.

1 Introduction

Detecting anomalies in huge volumes of data have many important real-life applications in areas such as machine health monitoring, intrusion detection systems, and novel pattern discovery in biological data [2]. However, it is also a challenging problem because in many modern applications the data arrives in a streaming fashion. The streaming data could be infinite, so offline algorithms that attempt to store the entire stream for analysis will not scale. Also in many situations, there is usually a lack of a complete (labeled) training set as new anomalous and non-anomalous patterns arise over time.

Although a lot of recent research has been focused on streaming anomaly detection [2], there is still lack of theoretically sound and practically effective algorithms that operate efficiently in a streaming model by making just one pass over the data. In practice, however, because of inherent correlations in the data, it is possible to reduce a large sized numerical stream into just a handful of hidden bases that can compactly describe the key patterns [10], and therefore dramatically reduce the complexity of further analysis. We exploit this observation in our proposed algorithm by maintaining a set of few orthogonal vectors that conceptually constitute previously seen normal patterns.

In this paper, we introduce a novel approach to anomaly detection in an unsupervised setting based on ideas from *matrix sketching*. We use matrix sketching to maintain (over time) a low-rank matrix with orthogonal columns that can (linearly) represent well all the identified non-anomalous datapoints previously-seen. We utilize this for anomaly detection as follows: let U be a low-rank matrix representing all non-anomalous datapoints till time $t-1$, for a new datapoint $\mathbf{y}$ arriving at time t, if there does not exist a good representation of $\mathbf{y}$ using U, then $\mathbf{y}$ does not lie close to the space of non-anomalous datapoints, and therefore $\mathbf{y}$ could be an anomaly. At the end of timestep t, the low-rank matrix is updated to capture all the non-anomalous points introduced at t.

*Due to space constraints some details are omitted in this extended abstract.

For efficient sketching, we adapt a recent deterministic sketching algorithm (called *Frequent Directions*) proposed by Liberty [7] and combine it with ideas from the theory of randomized low-rank matrix approximations. Our theoretical analysis is built upon the study of *Frequent Directions* by Liberty [7] and Ghashami *et al.* [4], and the recent results in matrix perturbation theory to prove that our randomized sketching-based algorithm has a similar performance to that of a global algorithm based on costly singular value decomposition updates. Our experimental results corroborate the performance and scalability of our approach on datasets drawn from gene sequencing, employee-activity logs, and broadcast news domains.

1.1 Preliminaries

Notation. We denote $[n] = 1 : n$. Vectors are denoted by boldface letters. For a vector $\mathbf{z} = (z_1, \ldots, z_m) \in \mathbf{R}^m$, $\mathrm{d}iag(z_1, \ldots, z_m) \in \mathbf{R}^{m \times m}$ denotes a diagonal matrix with $z_1, \ldots, z_m$ as its diagonal entries. Given a matrix Z, we abuse notation and use $\mathbf{y} \in Z$ to represent that $\mathbf{y}$ is a column in Z. Given a set of matrices, $Z_1, \ldots, Z_t$, we use the notation $Z_{[t]}$ to denote the matrix obtained by horizontally concatenating $Z_1, \ldots, Z_t$, i.e., $Z_{[t]} = [Z_1 | \ldots | Z_t]$.

We use $\textsc{Svd}(Z)$ to denote the singular value decomposition of Z, i.e., $\textsc{Svd}(Z) = U \Sigma V^\top$. We follow the common convention to list the singular values in non-increasing order. For a symmetric matrix $S \in \mathbf{R}^{m \times m}$, we use $\textsc{Eig}(S)$ to denote its eigenvalue decomposition, i.e., $U \Lambda U^\top = \textsc{Eig}(S)$. The *best rank-k approximation* (in both the spectral and Frobenius norm) to a matrix $Z \in \mathbf{R}^{m \times n}$ is $Z_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k$ are the top-k singular values of Z, with associated left and right singular vectors $\mathbf{u}_i \in \mathbf{R}^m$ and $\mathbf{v}_i \in \mathbf{R}^n$, respectively. We use $\textsc{Svd}_k(Z)$ to denote the the singular value decomposition of Z_k, i.e., $Z_k = \textsc{Svd}_k(Z) = U_k \Sigma_k V_k^\top$.

2 Streaming Anomaly Detection

Streaming Anomaly Detection Task. We assume that the data arrives in streams. Let $\{Y_t \in \mathbf{R}^{m \times n_t},\ t = 1, 2, \ldots\}$ denote a sequence of streaming data matrices, where Y_t represents the datapoints introduced at timestep t. Here m is the size of the feature space, and n_t is the number of datapoints arriving at time t. We typically assume that there are more datapoints than number of features ($n_t > m$). We normalize Y_t such that each column (point) in Y_t has a unit L_2-norm. Under this setup, the goal of streaming anomaly detection is to identify "anomalous datapoints" in Y_t at every timestep t.

Our Anomaly Detection Framework. Our idea is based on maintaining, at every timestep t, an *approximate low-rank matrix with orthogonal columns* that can reconstruct "well" all the prior (till time $t-1$) non-anomalous datapoints that the algorithm has identified. To develop an intuition for our approach, let us first consider a *simpler* setting where we assume that we know all the anomalies in $Y_{[t-1]} = [Y_1 | \ldots | Y_{t-1}]$, i.e., we know a partition of $Y_{[t-1]} = [Y_{[t-1]_{good}} | Y_{[t-1]_{bad}}]$, with the interpretation here being that the columns in $Y_{[t-1]_{bad}}$ are the anomalous points and $Y_{[t-1]_{good}}$ contains the non-anomalous (normal) points. Consider the rank-k approximation of $Y_{[t-1]_{good}}$ (for an appropriately chosen parameter[1] k): $Y_{[t-1]_{good_k}} = \textsc{Svd}_k(Y_{[t-1]_{good}}) = U_{t-1_k} \Sigma_{t-1_k} V_{t-1_k}^\top$. First observation is that U_{t-1_k} is a rank-k matrix that can "well" represent all the points in $Y_{[t-1]_{good}}$.[2] This follows from the observation that by setting $X = \Sigma_{t-1_k} V_{t-1_k}^\top$:

$$\sum_{\mathbf{y}_j \in Y_{[t-1]_{good}}} \min_{\mathbf{x}_j} \|\mathbf{y}_j - U_{t-1_k} \mathbf{x}_j\|^2 = \min_X \|Y_{[t-1]_{good}} - U_{t-1_k} X\|_F^2 \leq \|Y_{[t-1]_{good}} - Y_{[t-1]_{good_k}}\|_F^2.$$

In situations, where rank-k approximation is interesting, most of the mass from $Y_{[t]_{good}}$ would be in its top k singular values (components), resulting in $\|Y_{[t-1]_{good}} - Y_{[t-1]_{good_k}}\|_F^2$ being small.

We can now use U_{t-1_k} to detect anomalies in Y_t by following a simple approach. Since U_{t-1_k} is a good basis to linearly reconstruct all the observed non-anomalous points in $Y_{[t-1]}$, we can use it to test whether a point $\mathbf{y}_i \in Y_t$ is "close" to space of non-anomalous points or not. This can be easily achieved by solving the following simple least-squares problem:

$$\min_{\mathbf{x}} \|\mathbf{y}_i - U_{t-1_k} \mathbf{x}\|. \tag{1}$$

[1] Readers could think of k as a small number, much smaller than m or n_t.

[2] It is possible to use other (non-SVD) approaches to construct a matrix to linearly represent $Y_{[t-1]_{good}}$, however, using a low-rank SVD is attractive because it naturally comes with strong guarantees on approximation error.

As the columns of U_{t-1_k} are orthogonal to each other, this least-squares problem has a simple closed-form solution $\mathbf{x}^* = (U_{t-1_k}^\top U_{t-1_k})^{-1} U_{t-1_k}^\top \mathbf{y}_i = U_{t-1_k}^\top \mathbf{y}_i$. The objective value of (1) at $\mathbf{x}^*$ is used as the anomaly score to decide if $\mathbf{y}_i$ is anomalous or not, with larger objective values denoting anomalies. An alternate way of stating this is that a point $\mathbf{y}_i$ in Y_t is labeled anomalous if the length of the orthogonal projection of $\mathbf{y}_i$ onto the orthogonal complement U_{t-1_k} is "big".

In Algorithm ANOMDETECT, we present a simple prototype procedure for constructing $Y_{[t-1]_{good}}$ and $Y_{[t-1]_{bad}}$ based on maintaining the left singular vectors (corresponding to the top-k singular values) of the streaming data. The algorithm alternates between an anomaly detection and singular vector updating step.

Algorithm 1: ANOMDETECT (prototype algorithm for detecting anomalies at time t)

Input: $Y_t \in \mathbf{R}^{m \times n_t}$ (new observance), $U_{t-1_k} \in \mathbf{R}^{m \times k}$ (low-rank matrix with orthogonal columns), and $\zeta \in \mathbf{R}$ (threshold parameter)

Anomaly score construction step:
$Y_{t_{good}} \leftarrow [\,]$, $Y_{t_{bad}} \leftarrow [\,]$
for *each point (column)* $\mathbf{y}_i \in Y_t$ **do**
  Solve the least-squares problem: $\mathbf{x}_i^* = \text{argmin}_{\mathbf{x}} \|\mathbf{y}_i - U_{t-1_k}\mathbf{x}\|$ ($\Longrightarrow \mathbf{x}_i^* \leftarrow U_{t-1_k}^\top \mathbf{y}_i$)
  Anomaly score: $a_i \leftarrow \|(\mathbb{I}_m - U_{t-1_k} U_{t-1_k}^\top)\mathbf{y}_i\|$
  if $a_i \leq \zeta$ **then**
    $Y_{t_{good}} \leftarrow [Y_{t_{good}} | \mathbf{y}_i]$
  end
  else
    $Y_{t_{bad}} \leftarrow [Y_{t_{bad}} | \mathbf{y}_i]$
  end
end
Updating the singular vectors:
Generate $U_{t_k} \in \mathbf{R}^{m \times k}$, a matrix with orthogonal columns which is (or approximates) the left singular vectors corresponding to top-k singular values of $Y_{[t]_{good}}$
Return $Y_{t_{good}}$, $Y_{t_{bad}}$, *and* U_{t_k}

The simplest way of updating the singular vectors (without any errors) is to simply (re)generate them from the globally collected sample set $Y_{[t]_{good}}$. We call this approach *global updating*. However, this approach is not scalable as both the computational and memory requirements will increase with time. There are faster techniques for updating the singular vectors, based on a line of work commonly referred to as *Incremental Principal Component Analysis* (PCA) (see [1] and references therein), that attempts to maintain a low-rank approximation of a matrix Z (using SVD and a small amount of bookkeeping) as rows/columns of Z arrive in a stream. However, as noted in [4], these approaches can have arbitrarily bad matrix approximation error on adversarial data. In Section 4, we also present experimental evidence demonstrating that for anomaly detection, our approach outperforms a recent incremental PCA technique proposed by Baker *et al.* [1].

3 Streaming Anomaly Detection using Matrix Sketching

In this section, we propose an anomaly detection scheme for streaming data based on matrix sketching, and also provide theoretical guarantees for its efficacy.

In his recent paper Liberty [7] showed that by adapting the Misra-Gries approach for approximating frequency counts in a stream [9], one could obtain additive error bounds for matrix sketching. Recently, Ghashami and Philips [4], reanalyzed the *Frequent Directions* algorithm of Liberty [7], to show that it provides relative error bounds for low-rank matrix approximation.

Our approach for updating the singular vectors (outlined in Algorithm RANDSKETCH) is based on extending the *Frequent Directions* algorithm of Liberty [7] to a more general setting. In contrast to [7, 4], where one new row (or column) is added at every timestep t, we add $n_t \gg 1$ new columns. With this generality, for computational efficiency, at each timestep, we perform a low-rank SVD, instead of the full SVD as in [7, 4]. For constructing a low-rank SVD, we utilize a randomized low-rank matrix approximation technique suggested by Halko *et al.* [5] that is based on combining a randomized pre-processing step (multiplying by a random matrix and QR decomposition) along with a simple post-processing step (eigenvalue decomposition of a small matrix).

Algorithm 2: RANDSKETCH (randomized streaming update of the singular vectors at time t)

Input: $Y_{t_{good}} \in \mathbf{R}^{m \times n_t}$, and $E_{t-1} \in \mathbf{R}^{m \times \ell}$ the randomized matrix sketch computed at time $t-1$

$M_t \leftarrow [E_{t-1} | Y_{t_{good}}]$

$r \leftarrow 100\ell$

Generate an $m \times r$ random Gaussian matrix Ω

$Y \leftarrow M_t M_t^\top \Omega$

$QR \leftarrow \text{QR}(Y)$ (QR factorization of Y, computing orthogonal column basis for Y)

$A_t \breve{\Sigma}_t^2 A_t^\top \leftarrow \text{EIG}(Q^\top M_t M_t^\top Q)$ (with $\breve{\Sigma}_t^2 = diag(\breve{\sigma}_{t_1}^2, \ldots, \breve{\sigma}_{t_r}^2)$)

$\breve{U}_t \leftarrow QA_t$ ($\implies QQ^\top M_t M_t^\top QQ^\top$ is the produced approximation of $M_t M_t^\top$)

$\breve{U}_{t_\ell} \leftarrow [\mathbf{u}_1, \ldots, \mathbf{u}_\ell]$ (where $\breve{U}_{t_r} = [\mathbf{u}_1, \ldots, \mathbf{u}_r]$ and $\ell \leq r$)

$\breve{\Sigma}_{t_\ell}^{(trunc)} \leftarrow diag\left(\sqrt{\breve{\sigma}_{t_1}^2 - \breve{\sigma}_{t_\ell}^2}, \sqrt{\breve{\sigma}_{t_2}^2 - \breve{\sigma}_{t_\ell}^2}, \ldots, \sqrt{\breve{\sigma}_{t_{\ell-1}}^2 - \breve{\sigma}_{t_\ell}^2}, 0\right)$

$E_t = \breve{U}_{t_\ell} \breve{\Sigma}_{t_\ell}^{(trunc)}$

Return E_t *and* $\breve{U}_{t_k}$

At time t, the running time of the Algorithm RANDSKETCH is $O(\ell T_{mult} + (m + n_t)\ell^2)$, where T_{mult} denotes the cost of a matrix-vector multiplication with the input matrix M_t. The matrix-vector multiplication is a well-studied topic with numerous known efficient sequential/parallel algorithms. Between iterations the algorithm only needs to store the E_t (the up-to-date randomized matrix sketch) matrices which take $O(m\ell)$ storage. We discuss the setting of ℓ in the next section.

3.1 Bounding the Performance of Algorithm RANDSKETCH

In this section, we would show that the anomaly detection results obtained by using $\breve{U}_{t_k}$ (output of Algorithm RANDSKETCH) in Algorithm ANOMDETECT is similar to using the (true) singular vectors based on a global update. Due to space limitations all proofs and detailed discussions are omitted in this extended abstract.

Our first aim will be to bound the Frobenius norm of the difference between $Y_{[t]_{good_k}} Y_{[t]_{good_k}}^\top$ and $E_{t_k} E_{t_k}^\top$, for which we will use the following result from Halko *et al.* [5] that bounds the error due to randomized SVD.

Theorem 1 (Restated from Corollary 10.9 [5]). *In Algorithm* RANDSKETCH, *let* $diag(\bar{\sigma}_{t_1}, \ldots, \bar{\sigma}_{t_m})$ *be the eigenvalues of* $M_t M_t^\top$, *then with probability at least* $1 - 6e^{-99\ell}$, $\|M_t M_t^\top - \breve{U}_t \breve{\Sigma}_t^2 \breve{U}_t\| \leq 38\bar{\sigma}_{t_{\ell+1}} + 2(\sum_{i=\ell+1}^m \bar{\sigma}_{t_i}^2)^{1/2}/\sqrt{\ell}$.

Similar to the above theorem, we can bound $\|M_j M_j^\top - \breve{U}_j \breve{\Sigma}_j^2 \breve{U}_j\|$, for every timestep j. We will need few additional notations:

$$N_t = QQ^\top M_t,$$

$$P_t = QA_t\breve{\Sigma}_t = \breve{U}_t\breve{\Sigma}_t (\text{note that by construction in Algorithm RANDSKETCH, } N_t N_t^\top = P_t P_t^\top),$$

$$E_{t_k} = \breve{U}_{t_k} \breve{\Sigma}_{t_k}^{(trunc)} \text{ (rank-}k\text{ approximation of } E_t),$$

$$\breve{\Delta}_t = \textstyle\sum_{j=1}^t \breve{\sigma}_{j\ell}^2,$$

$$\upsilon_j = 38\bar{\sigma}_{j_{\ell+1}} + \frac{2\left(\sum_{i=\ell+1}^m \bar{\sigma}_{j_i}^2\right)^{1/2}}{\sqrt{\ell}} (\text{error bound from Theorem 1, at timestep } j) \text{ and } \Upsilon_t = \textstyle\sum_{j=1}^t \upsilon_j, \quad (2)$$

$$\kappa = \sigma_1(Y_{[t]_{good}})/\sigma_k(Y_{[t]_{good}}), \text{ where } \sigma_1(Y_{[t]_{good}}) \geq \cdots \geq \sigma_m(Y_{[t]_{good}}) \text{ are the singular values of } Y_{[t]_{good}}.$$

As columns of Q are orthogonal to each other, $QQ^\top$ is a projection matrix, and therefore by standard properties of projection matrices and noting that $(QQ^\top)^\top = QQ^\top$,

$$\|M_t\|_F^2 \geq \|QQ^\top M_t\|_F^2 = \|N_t\|_F^2 = \|P_t\|_F^2,$$

$$\forall \text{ unit vectors } \mathbf{x} \in \mathbf{R}^m, \|M_t^\top \mathbf{x}\|^2 \geq \|QQ^\top M_t \mathbf{x}\|^2 = \|N_t^\top \mathbf{x}\|^2 = \|P_t^\top \mathbf{x}\|^2.$$

Lemma 2. *At timestep t, Algorithm* RANDSKETCH *maintains that:* $\|Y_{[t]_{good}}\|_F^2 - \|E_t\|_F^2 \geq \ell\breve{\Delta}_t$.

The following lemma shows that for any direction $\mathbf{x}$, $Y_{[t]_{good}}$ and E_t are w.h.p. not too far apart.

Lemma 3. *For any unit vector* $\mathbf{x} \in \mathbf{R}^m$*, at any timestep* t*, with probability at least* $1 - 6e^{-99\ell}$*,* $0 \leq \|Y_{[t]_{good}}^\top \mathbf{x}\|^2 - \|E_t^\top \mathbf{x}\|^2 \leq \breve{\Delta}_t + \Upsilon_t$.

Since for all unit vectors $\mathbf{x} \in \mathbf{R}^m$, $\|Y_{[t]_{good}}^\top \mathbf{x}\|^2 - \|E_t^\top \mathbf{x}\|^2 \geq 0 \Longrightarrow Y_{[t]_{good}} Y_{[t]_{good}}^\top \succeq E_t E_t^\top$. It can also be easily established that for all unit vectors $\mathbf{x} \in \mathbf{R}^m$, $\kappa \|Y_{[t]_{good_k}}^\top \mathbf{x}\| \geq \|Y_{[t]_{good}}^\top \mathbf{x}\|$. Therefore,

$$\kappa^2 Y_{[t]_{good_k}} Y_{[t]_{good_k}}^\top \succeq Y_{[t]_{good}} Y_{[t]_{good}}^\top \succeq E_t E_t^\top \succeq E_{t_k} E_{t_k}^\top.$$

Lemma 4. *Let* $Y_{[t]_{good_k}}$ *be the best rank-*k *approximation to* $Y_{[t]_{good}}$*. Then with probability at least* $1 - 6e^{-99\ell}$*,* $\breve{\Delta}_t \leq (\|Y_{[t]_{good}} - Y_{[t]_{good_k}}\|_F^2 + k\Upsilon_t)/(\ell - k)$.

Lemma 5. *Algorithm* RANDSKETCH *satisfies with probability at least* $1 - 6e^{-99\ell}$*:*

$$0 \leq \|Y_{[t]_{good_k}}\|_F^2 - \|E_{t_k}\|_F^2 \leq k\Upsilon_t + k(\|Y_{[t]_{good}} - Y_{[t]_{good_k}}\|_F^2 + k\Upsilon_t)/(\ell - k).$$

Using this above lemma and the fact that $\kappa^2 Y_{[t]_{good_k}} Y_{[t]_{good_k}}^\top \succeq E_{t_k} E_{t_k}^\top$, we can prove the following proposition.

Proposition 6. *At timestep* t*,* E_{t_k} *generated by Algorithm* RANDSKETCH *satisfies,* $\|\kappa^2 Y_{[t]_{good_k}} Y_{[t]_{good_k}}^\top - E_{t_k} E_{t_k}^\top\|_F \leq \kappa^2 \|Y_{[t]_{good_k}}\|_F^2 - \|E_{t_k}\|_F^2$.

We need couple of more definitions. Define Φ_a as,

$$\Phi_a = \frac{\kappa^2 \|Y_{[t]_{good_k}}\|_F^2 - \|E_{t_k}\|_F^2}{\|Y_{[t]_{good_k}}\|_F^2 - \|E_{t_k}\|_F^2}. \tag{3}$$

Note that $\Phi_a \geq 1$ as $\|Y_{[t]_{good_k}}\|_F^2 \geq \|E_{t_k}\|_F^2$ (from Lemma 5). In fact, for small k's (as in our setting), typically κ (the ratio between the largest and kth largest singular value of $Y_{[t]_{good}}$) is bounded, yielding $\Phi_a = O(1)$. Define Φ_b as,

$$\Phi_b = \frac{\|\kappa^2 Y_{[t]_{good}} Y_{[t]_{good}}^\top - E_t E_t^\top\|}{\|\kappa^2 Y_{[t]_{good_k}} Y_{[t]_{good_k}}^\top - E_{t_k} E_{t_k}^\top\|}. \tag{4}$$

Claim 7. Φ_b *satisfies:* $\Phi_b \leq 1 + 2/(\kappa^2 - \|E_t\|^2/\|Y_{[t]_{good}}\|^2)$.

Remember that $\|E_t\|^2 \leq \|Y_{[t]_{good}}\|^2$. Typically κ is also bounded away from 1, yielding $\Phi_b = O(1)$.

We now use the theory of matrix perturbation to relate $\breve{U}_{t_k}$ (from Algorithm RANDSKETCH) with the (true) left singular vectors corresponding to top-k singular values of $Y_{[t]_{good}}$. There is lot of prior work in matrix perturbation theory that relates the eigenvalues, singular values, eigenspaces, and singular subspaces, etc., of the matrix $Z + Z'$ to the corresponding quantity in Z, under various conditions on the matrices Z and Z'. Here we use a recent result from Chen, Li, and Xu [3] that studies behavior of the eigenvector matrix of a Hermitian (symmetric) matrix under perturbation.

Theorem 8 (Restated from Theorem 2.1 [3])**.** *Let* $A \in \mathbf{R}^{m \times m}$ *be a symmetric matrix with distinct eigenvalues with* $\text{EIG}(A) = U \Lambda U^\top$ *where* $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_m)$*. Let* $A_{\text{per}} = A + \Phi$ *be a symmetric matrix. Let* $L = L(\Lambda) = \min_{i \neq j} |\lambda_i - \lambda_j| > 0$*,* $\beta = \|\Phi\|_F / L$*, and* $\alpha = 2\|A\|/L$*, with* β *satisfying:* $\beta \leq 1/(1 + 4\alpha)$*. Then* $\text{EIG}(A_{\text{per}}) = U_{\text{per}} \Lambda_{\text{per}} U_{\text{per}}^\top$ *such that* $\|U - U_{\text{per}}\|_F \leq \sqrt{2\beta}/(1 + 4\alpha^2)^{1/4}$.

We now apply Proposition 6 and Theorem 8 to bound $\|\hat{U}_{t_k} - \breve{U}_{t_k}\|_F$. To do so we construct matrices: $A = \kappa^2 Y_{[t]_{good}} Y_{[t]_{good}}^\top$ and $A_{\text{per}} = E_t E_t^\top$. Let ℓ be such that:

$$\frac{\sqrt{m} \Phi_b \Phi_a k \Upsilon_t}{L} + \frac{\sqrt{m} \Phi_b \Phi_a k (\|Y_{[t]_{good}} - Y_{[t]_{good_k}}\|_F^2 + k\Upsilon_t)}{(\ell - k) L} \leq \frac{L}{L + 4\kappa^2 \|Y_{[t]_{good}}\|^2}. \tag{5}$$

In the above equation both terms in the left-hand side are decreasing functions in ℓ (for the first term note that Υ_t decreases with ℓ).

Claim 9. *Let* λ_i *be the* i*th eigenvalue of* $Y_{[t]_{good}} Y_{[t]_{good}}^\top$ *and* $L = \min_{i \neq j} |\lambda_i - \lambda_j| > 0$*. If* ℓ *satisfies* (5) *for* $\Upsilon_t, \Phi_a, \Phi_b$ *defined in* (2)*,* (3)*,* (4) *respectively, then with probability at least* $1 - 6e^{-99\ell}$*,*

$$\|\hat{U}_{t_k} - \breve{U}_{t_k}\|_F \leq \sqrt{2} L / \left(\sqrt{L + 8\kappa^2 \|Y_{[t]_{good}}\|^2} \sqrt[4]{L^2 + 16\kappa^4 \|Y_{[t]_{good}}\|^4} \right).$$

Neither the numerical constants nor the precise form of the bound on ℓ are optimal because of the slackness in Theorem 8.

Remark: The assumption of $L > 0$ is something that is commonly satisfied in practice, especially if m is reasonably smaller than the number of datapoints in $Y_{[t]_{good}}$. The above bound on ℓ should be treated as an existential result, as setting ℓ using the established bound is tricky. Practically, we noticed that setting $\ell \approx \sqrt{m}$ suffices to get good results. Another important point to remember is that the Algorithm RANDSKETCH can be used *with any value* of ℓ, the above bound on ℓ is only to ensure that its performance is similar to using global singular value decomposition updates in Algorithm ANOMDETECT as established in the following theorem.

Theorem 10 (Comparing Anomaly Scores)**.** *Let $Y_{1_{good}}, \ldots, Y_{t_{good}}$ be a sequence of matrices with $Y_{[t]_{good}} = [Y_{1_{good}} | \ldots | Y_{t_{good}}]$. Let $Y_{[t]_{good_k}} = \hat{U}_{t_k} \hat{\Sigma}_{t_k} \hat{V}_{t_k}^\top$ be the best rank-k approximation to $Y_{[t]_{good}}$. Then for any unit vector $\mathbf{y} \in \mathbf{R}^m$, $\breve{U}_{t_k}$ (generated by the Algorithm* RANDSKETCH*), under conditions from Claim 9, with probability at least $1 - 6e^{-99\ell}$, satisfies:*

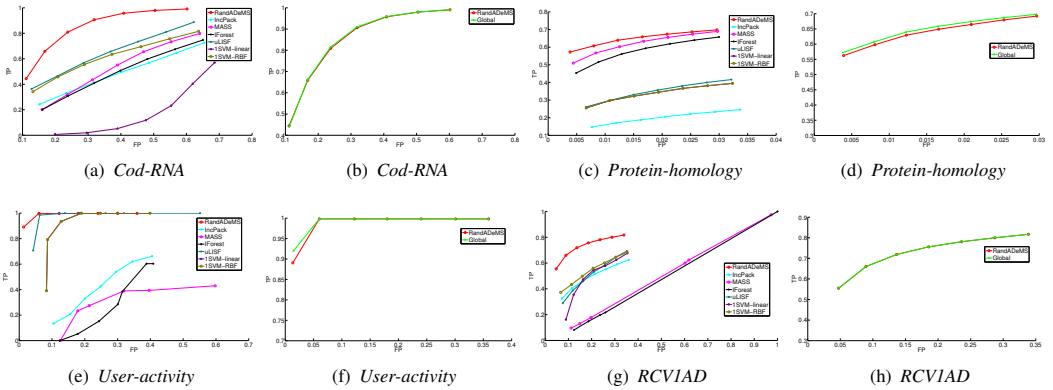
$$\left| \min_{\mathbf{x} \in \mathbf{R}^k} \|\mathbf{y} - \hat{U}_{t_k}\mathbf{x}\| - \min_{\mathbf{x} \in \mathbf{R}^k} \|\mathbf{y} - \breve{U}_{t_k}\mathbf{x}\| \right| \leq \sqrt{2}L/(\sqrt{L + 8\kappa^2 \|Y_{[t]_{good}}\|^2} \sqrt[4]{L^2 + 16\kappa^4 \|Y_{[t]_{good}}\|^4}).$$

The above theorem shows that the anomaly scores (in Algorithm ANOMDETECT) constructed by using either matrices $\breve{U}_{t_k}$ or $\hat{U}_{t_k}$ (true top-k singular vectors) are "almost" the same.

4 Experimental Results

We experimentally test our proposed approach in terms of effectiveness and efficiency. We use datasets drawn from a diverse set of domains ranging from gene sequencing (*Cod-RNA*, *Protein-homology*), to employee activity log (*User-activity*), to broadcast news (*RCV1AD*). We refer to Algorithm ANOMDETECT with singular vectors updated using a global SVD approach as **GLOBAL** and using Algorithm RANDSKETCH as **RANDADEMS** .

Baselines. We compare against some popular algorithms that were chosen for their scalability on large datasets. **1SVM-linear** and **1SVM-RBF** are one-class support vector machine classifiers with linear/radial-basis as kernel function. **IForest** [8] and **Mass** [11] use modeling of *attribute distribution* to detect anomalies, which is known to be very efficient as they rely on simple data processing. **Unconstrained Least-Squares Importance Fitting** (uLSIF) [6] uses *density ratio estimation* to detect anomalies. **IncPack** uses incremental PCA to update the singular vectors.



(a) *Cod-RNA* (b) *Cod-RNA* (c) *Protein-homology* (d) *Protein-homology*

(e) *User-activity* (f) *User-activity* (g) *RCV1AD* (h) *RCV1AD*

Figure 1: ROC curves for compared approaches on various datasets.

From Figure 1, it is evident that RANDADEMS outperforms the other algorithms on all the datasets (except for the experiments on *User-activity* dataset, Figure 1(e), which shows a partial overlap between RANDADEMS, 1SVM-RBF, and uLSIF). Due to updates to the basis vectors, RANDADEMS can successfully deal with the concept shift problem in the normal data (i.e., new patterns of normal data appearing over time). RANDADEMS also has extremely similar performance to GLOBAL (Figures 1(b), 1(d), 1(f), and 1(h)). It once again confirms that the Algorithm RANDSKETCH produces a desired approximation to the top-k singular vectors.

In terms of the running time, on average, RANDADEMS is over 100 times faster than 1SVM-linear, 1SVM-RBF, and uLSIF, and is over 5 times faster than IForest, Mass, IncPack, and GLOBAL.

References

[1] C. G. Baker, K. A. Gallivan, and P. Van Dooren. Low-rank incremental methods for computing dominant singular subspaces. *Linear Algebra and its Applications*, 2012.

[2] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–72, 2009.

[3] X. Chen, W. Li, and W. Xu. Perturbation analysis of the eigenvector matrix and singular vector matrices. *Taiwanese Journal of Mathematics*, 16(1):pp–179, 2012.

[4] M. Ghashami and J. M. Phillips. Relative errors for deterministic low-rank matrix approximations. In *SODA*, pages 707–717, 2014.

[5] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2), 2011.

[6] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori. Statistical outlier detection using direct density ratio estimation. *KAIS*, 26(2):309–336, 2011.

[7] E. Liberty. Simple and deterministic matrix sketching. In *ACM SIGKDD*, pages 581–588, 2013.

[8] F. T. Liu, K. M. Ting, and Z. H. Zhou. Isolation forest. *IEEE ICDM*, pages 413–422, 2008.

[9] J. Misra and D. Gries. Finding repeated elements. *Science of computer programming*, 2(2):143–152, 1982.

[10] S. Papadimitriou, J. Sun, and C. Faloutsos. Streaming pattern discovery in multiple time-series. In *VLDB*, 2005.

[11] K. M. Ting, G. T. Zhou, F. T. Liu, and J. S. Tan. Mass estimation and its applications. *ACM SIGKDD*, 2010.