Streaming SVD

Ahmer Nadeem Khan

Florida State University

February 18, 2026

Outline

Outline

Incremental SVD

We want an SVD of the augmented matrix

$$\widetilde{A} := [A \;\; C] \in \mathbb{R}^{m\times(n+c)}$$

without recomputing from scratch.

Incremental SVD

Project C onto the current left singular subspace $\mathrm{span}(U)$:

$$L := U^\top C \in \mathbb{R}^{r \times c}.$$

Split C into "in-subspace" plus orthogonal remainder:

$$C = UL + H, \qquad H := (I - UU^\top)C \in \mathbb{R}^{m \times c}.$$

By construction,

$$U^\top H = U^\top (I - UU^\top)C = 0.$$

So columns of H are orthogonal to columns of U.

Incremental SVD

Compute a thin QR factorization of H:

$$H = JK,$$

where

$$J \in \mathbb{R}^{m \times \rho}, \quad J^\top J = I_\rho, \qquad K \in \mathbb{R}^{\rho \times c} \text{ upper triangular,}$$

and $\rho = \mathrm{rank}(H) \leq c$.
Because $U^\top H = 0$, we also have

$$U^\top J = 0 \quad \Longrightarrow \quad [U \;\; J]^\top [U \;\; J] = \begin{bmatrix} I_r & 0 \\ 0 & I_\rho \end{bmatrix}.$$

Hence $[U \;\; J]$ is an orthonormal basis for $\mathrm{span}(U) \oplus \mathrm{span}(J)$.

Incremental SVD

Start with

$$\widetilde{A} = [A \;\; C] = [U\Sigma V^\top \;\; C].$$

Insert the decomposition $C = UL + JK$:

$$\widetilde{A} = [U\Sigma V^\top \;\; UL + JK].$$

Factor out the left orthonormal block $[U \;\; J]$:

$$[U\Sigma V^\top \;\; UL + JK] = [U \;\; J] \begin{bmatrix} \Sigma V^\top & L \\ 0 & K \end{bmatrix}.$$

So far this is exact.

Incremental SVD

Now separate the right-side orthonormal pieces by writing

$$\begin{bmatrix} \Sigma V^\top & L \\ 0 & K \end{bmatrix} = \underbrace{\begin{bmatrix} \Sigma & L \\ 0 & K \end{bmatrix}}_{=:Q\in\mathbb{R}^{(r+\rho)\times(r+c)}} \underbrace{\begin{bmatrix} V^\top & 0 \\ 0 & I_c \end{bmatrix}}_{=:W^\top}.$$

Therefore,

$$\widetilde{A} = [U \;\; J]\; Q\; W^\top.$$

Here $[U \;\; J]$ has orthonormal columns, and W has orthonormal columns as well.

Key Idea: Reduce the Update to an SVD of Q

We have expressed $\widetilde{A}$ as

$$\widetilde{A} = \underbrace{[U \;\; J]}_{\text{orthonormal}} \; \underbrace{Q}_{\text{small}} \; \underbrace{W^\top}_{\text{orthonormal}} .$$

Thus, an SVD of $\widetilde{A}$ can be obtained by computing an SVD of the *small core* matrix Q.
Compute (thin) SVD:

$$Q = \widehat{U}\,\widehat{\Sigma}\,\widehat{V}^\top,$$

where

$$\widehat{U} \in \mathbb{R}^{(r+\rho)\times(r+\rho)}, \quad \widehat{\Sigma} \in \mathbb{R}^{(r+\rho)\times(r+\rho)}, \quad \widehat{V} \in \mathbb{R}^{(r+c)\times(r+\rho)}.$$

(Any consistent thin/full convention is fine)

Incremental SVD

Substitute $Q = \widehat{U}\widehat{\Sigma}\widehat{V}^\top$ into $\widetilde{A}$:

$$\widetilde{A} = [U \;\; J]\, \widehat{U}\, \widehat{\Sigma}\, \widehat{V}^\top\, W^\top.$$

Define updated factors:

$$U_{\text{new}} := [U \;\; J]\widehat{U}, \qquad \Sigma_{\text{new}} := \widehat{\Sigma}, \qquad V_{\text{new}} := W\widehat{V}.$$

Then

$$\widetilde{A} = U_{\text{new}}\, \Sigma_{\text{new}}\, V_{\text{new}}^\top.$$

Orthonormality:

$$U_{\text{new}}^\top U_{\text{new}} = \widehat{U}^\top [U \;\; J]^\top [U \;\; J]\widehat{U} = \widehat{U}^\top I\, \widehat{U} = I,$$

and similarly $V_{\text{new}}^\top V_{\text{new}} = I$.

Optional: Truncation Back to Rank r

Often we want to maintain rank r.
After computing the SVD of Q, keep only the top r singular values:

$$Q \approx \widehat{U}_r \widehat{\Sigma}_r \widehat{V}_r^\top,$$

where $\widehat{U}_r \in \mathbb{R}^{(r+\rho)\times r}$, $\widehat{\Sigma}_r \in \mathbb{R}^{r\times r}$, $\widehat{V}_r \in \mathbb{R}^{(r+c)\times r}$.
Then the *rank-r* updated approximation is

$$\widetilde{A} \approx \underbrace{([U \;\; J]\widehat{U}_r)}_{U_{\text{new},r}} \; \widehat{\Sigma}_r \; \underbrace{(W\widehat{V}_r)}_{V_{\text{new},r}}^{\top}.$$

Key Idea

The update is:

$$[A \;\; C] = [U \;\; J] \underbrace{\begin{bmatrix} \Sigma & U^\top C \\ 0 & K \end{bmatrix}}_{\text{small}} \begin{bmatrix} V^\top & 0 \\ 0 & I_c \end{bmatrix}, \quad \text{with } (I - UU^\top)C = JK.$$

So you only need an SVD of the small middle matrix.

Other Related Ideas

1. **Incremental SVD**
   - Update an existing SVD when new data arrives.
   - Maintains a continuously evolving factorization.
   - Efficient but introduces dependency across updates.
2. **Subspace Tracking**
   - Maintain one low-rank subspace over time.
   - Used in signal processing (PAST, Oja).
3. **Block / Hierarchical Methods**
   - Partition matrix into spatial or temporal blocks.
   - Use hierarchical low-rank structure (H-matrices, HSS).

Outline

Streaming SVD

Problem Setting
We observe a sequence of snapshots

$$S_1, S_2, \ldots, S_T, \qquad S_t \in \mathbb{R}^{m \times n}$$

Goal:

- Compute a low-rank SVD of each snapshot.
- Exploit structural similarity between consecutive snapshots.
- Maintain independent compressed storage per snapshot.

Streaming SVD

Key Observation

Assume

$$S_t = S_{t-1} + E_t$$

If $\|E_t\|$ is small and a spectral gap exists, then

$$\text{range}(U_t) \approx \text{range}(U_{t-1})$$

by classical perturbation theory (Davis–Kahan, Wedin).
Therefore, the dominant singular subspaces drift slowly.

Streaming SVD

Wedin's Theorem (Perturbation of Singular Subspaces)
Let $M, \widetilde{M} \in \mathbb{R}^{m\times n}$ with

$$M = [U_1 \ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}, \quad \widetilde{M} = M + \Delta = [\widetilde{U}_1 \ \widetilde{U}_2] \begin{bmatrix} \widetilde{\Sigma}_1 & 0 \\ 0 & \widetilde{\Sigma}_2 \end{bmatrix} \begin{bmatrix} \widetilde{V}_1^T \\ \widetilde{V}_2^T \end{bmatrix}$$

Define the spectral gap

$$\delta = \min \left\{ \min_{\substack{1\le i\le r \\ r+1\le j\le n}} |\sigma_i - \widetilde{\sigma}_j|, \ \min_{1\le i\le r} \sigma_i \right\} > 0.$$

Then

$$\|\sin\Theta(\widetilde{U}_1, U_1)\|_F^2 + \|\sin\Theta(\widetilde{V}_1, V_1)\|_F^2 \ \le \ \frac{\|U_1^T\Delta\|_F^2 + \|\Delta V_1\|_F^2}{\delta^2}.$$

Streaming SVD

Interpretation of Wedin's Theorem

Assume consecutive snapshots satisfy:

$$S_t = S_{t-1} + E_t.$$

Wedin's theorem implies:

- If $\|E_t\|$ is small,
- and a spectral gap δ exists,

then

$$\sin \Theta(U_t, U_{t-1}) \lesssim \frac{\|E_t\|}{\delta}.$$

Implication for Streaming SVD:

- Dominant singular subspaces drift slowly.
- U_{t-1} is already a high-quality approximation to U_t.
- Warm-starting randomized SVD drastically reduces required iterations.

Small perturbation $\Rightarrow$ small subspace rotation $\Rightarrow$ fast convergence.

Streaming SVD

Standard Randomized SVD

For a matrix A:

$$Y = A\Omega$$

where Ω is Gaussian.

Then:

1. $Q = \text{orth}(Y)$
2. Compute SVD of $Q^T A$
3. Form truncated approximation

Streaming SVD

Warm-Started
Instead of a purely random probe, use the previous left singular vectors:

$$Y = S_t U_{t-1}$$

This immediately captures dominant energy if subspace drift is small.

Streaming SVD

Residual Correction (Necessary for Stability)

To capture new emerging directions:

$$Y = S_t [\, U_{t-1} \;\; \Omega \,]$$

where:

- U_{t-1} = smart probe
- Ω = small Gaussian block

Streaming SVD

Algorithm Structure

For snapshot S_t:

1. Form $Y = S_t[U_{t-1}, \Omega]$
2. $Q = \text{orth}(Y)$
3. Compute small SVD of $Q^T S_t$
4. Recover approximate SVD of S_t
5. Store (U_t, Σ_t, V_t) independently

Important: U_{t-1} is used only during solving.

Streaming SVD

Compression vs Decompression
Compression (Dependent)

- Uses U_{t-1} to accelerate convergence.

Decompression (Independent)

- Each (U_t, Σ_t, V_t) stored as standalone block.
- No dependency on S_{t-1}.

Streaming SVD

Error Control (?)

After computing Q, estimate residual:

$$\|(I - QQ^T)S_t\|$$

Adaptive strategy:

- Add random columns until residual $\leq$ tolerance.

This preserves strict approximation guarantees.

Outline

Streaming SVD

Assumed Structure in Snapshot Data

We observe a sequence:

$$S_1, S_2, \ldots, S_T$$

Streaming acceleration is possible only if snapshots exhibit structure:

- Small perturbations between steps
- Slowly drifting dominant subspaces
- Low-rank updates
- Localized spatial or temporal changes

Without structure, we must recompute from scratch.

Streaming SVD

Model 1: Small Perturbation Model

Assume

$$S_t = S_{t-1} + E_t, \quad \|E_t\| \text{ small.}$$

By Wedin/Davis–Kahan:

$$\sin\Theta(U_t, U_{t-1}) \lesssim \frac{\|E_t\|}{\delta}.$$

Implication:

- Dominant subspaces drift slowly.

Streaming SVD

Model 2: Low-Rank Update Structure

Assume

$$S_t = S_{t-1} + A_t B_t^T, \quad \mathrm{rank}(A_t B_t^T) \ll n.$$

Properties:

- Only a few new directions introduced.
- Residual energy concentrated in small subspace.
- Random correction block can be very small.

Common in scientific simulations and video data.

Streaming SVD

Model 3: Slowly Rotating Subspace

Assume

$$S_t = U_t \Sigma_t V_t^T, \quad U_t = U_{t-1} R_t,$$

where R_t is a small rotation.

Then:

$$\|U_t - U_{t-1}\| \text{ small.}$$

Implication:

- Principal components evolve smoothly.
- Warm-start approximates one power iteration step.

Streaming SVD

Model 4: Block / Localized Changes

Assume partitioned structure:

$$S_t = \begin{bmatrix} S_{t-1}^{(1)} & * \\ * & S_{t-1}^{(2)} \end{bmatrix} + \text{localized update.}$$

Properties:

- Changes affect subset of rows/columns.
- Hierarchical/block methods may help.

Streaming SVD

When Streaming Fails

If snapshots are statistically independent:

$$S_t \perp S_{t-1},$$

then:

- Subspaces unrelated.
- Full randomized SVD required each time.

Next Steps

Construct synthetic snapshots:

$$S_t = S_{t-1} + E_t,$$

where:

- $S_1 = U\Sigma V^T$ is low-rank,
- E_t is a small random perturbation,
- $\|E_t\|$ is controlled.

Drift parameter:

$$\eta_t = \|E_t\|.$$

Next Steps

Experimental Comparison

For each snapshot, compare:

- Standard randomized SVD
- Warm-started randomized SVD

Measure:

- Reconstruction error:

$$\|S_t - U_t \Sigma_t V_t^T\|$$

- Required oversampling size
- Number of matrix–vector products
- Runtime

Next Steps

Expected Outcome

From Wedin/Davis–Kahan:

$$\sin\Theta(U_t, U_{t-1}) \lesssim \frac{\|E_t\|}{\delta}.$$

Thus:

$\|E_t\|$ small $\quad\Rightarrow\quad$ Subspace drift small $\quad\Rightarrow\quad$ Warm-start effective.