COL866: Foundations of Data Science

Ragesh Jaiswal, IITD

Matrix Algorithm using Sampling

Sketching

- The data can be stored in the memory but we would like to avoid working directly with the data (it may be in a slower memory) and create a sketch of the data so that:
  - The sketch retains the important properties of the data with respect to the computational task we want to perform on the data.
  - The sketch takes much smaller (faster) memory.
- Example: Matrix multiplication where the task is to multiply two matrices A and B. We would like to create sketches of the matrices that take much smaller space so that AB can be approximated using just the sketches.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:, k)$ denote the k^{th} column of A and $A(k, :)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:, k)B(k, :)$. Note that $A(:, k)B(k, :)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Question: What is $\mathbf{E}[X]$?

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:, k)$ denote the k^{th} column of A and $A(k, :)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:, k)B(k, :)$. Note that $A(:, k)B(k, :)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Claim: $\mathbf{E}[X] \stackrel{?}{=} AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:,k)$ denote the k^{th} column of A and $A(k,:)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:,k)B(k,:)$. Note that $A(:,k)B(k,:)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Claim: $\mathbf{E}[X] \stackrel{?}{=} AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.

Calculations

$$\begin{aligned}
Var[X] &= \sum_{i=1}^{m}\sum_{j=1}^{p} Var[X_{ij}] = \sum_{i,j}(\mathbf{E}[X_{ij}^2] - \mathbf{E}[X_{ij}]^2) \\
&= \sum_{i,j}\sum_{k} p_k \frac{A_{ik}^2 B_{kj}^2}{p_k^2} - ||AB||_F^2 \\
&= \sum_{k} \frac{1}{p_k}\left(\sum_{i} A_{ik}^2\right)\left(\sum_{i} B_{kj}^2\right) - ||AB||_F^2 \\
&= \sum_{k} \frac{1}{p_k}||A(:,k)||^2||B(k,:)||^2 - ||AB||_F^2
\end{aligned}$$

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:,k)$ denote the k^{th} column of A and $A(k,:)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:,k)B(k,:)$. Note that $A(:,k)B(k,:)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Claim: $\mathbf{E}[X] \stackrel{?}{=} AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.

Calculations

$$Var[X] = \sum_k \frac{1}{p_k} ||A(:,k)||^2 ||B(k,:)||^2 - ||AB||_F^2$$

- The RHS is minimized when p_k's are proportional to $||A(:,k)|| \cdot ||B(k,:)||$.
- For ease of calculations let us use $p_k = ||A(:,k)||^2$. This gives $Var[X] \leq ||A||_F^2 \sum_k ||B(k,:)||^2 = ||A||_F^2 \cdot ||B||_F^2$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:, k)$ denote the k^{th} column of A and $A(k, :)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:, k)B(k, :)$. Note that $A(:, k)B(k, :)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Claim: $\mathbf{E}[X] = AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.
- For ease of calculations let us use $p_k = ||A(:, k)||^2$. This gives $Var[X] \leq ||A||_F^2 \sum_k ||B(k, :)||^2 = ||A||_F^2 \cdot ||B||_F^2$.
- In order to obtain an X with smaller variance, we can do s independent trials to obtain matrices $X_1, ..., X_s$ and take an average. That is $X = \frac{X_1 + ... + X_s}{s}$.
- Claim: For such an X, $Var[X] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:, k)$ denote the k^{th} column of A and $A(k, :)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:, k)B(k, :)$. Note that $A(:, k)B(k, :)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- <u>Claim</u>: $\mathbf{E}[X] = AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.
- For ease of calculations let us use $p_k = ||A(:, k)||^2$. This gives $Var[X] \leq ||A||_F^2 \sum_k ||B(k, :)||^2 = ||A||_F^2 \cdot ||B||_F^2$.
- In order to obtain an X with smaller variance, we can do s independent trials to obtain matrices $X_1, ..., X_s$ and take an average. That is $X = \frac{X_1 + ... + X_s}{s}$.
- <u>Claim</u>: For such an X, $Var[X] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}$.
- Let $k_1, ..., k_s$ denote the k's chosen in each trial. Then $X = \frac{1}{s}\left(\frac{A(:,k_1)B(k_1,:)}{p_{k_1}} + ... + \frac{A(:,k_1)B(k_1,:)}{p_{k_1}}\right)$

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:,k)$ denote the k^{th} column of A and $A(k,:)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:,k)B(k,:)$. Note that $A(:,k)B(k,:)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- <u>Claim</u>: $\mathbf{E}[X] = AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.
- For ease of calculations let us use $p_k = ||A(:,k)||^2$. This gives $Var[X] \leq ||A||_F^2 \sum_k ||B(k,:)||^2 = ||A||_F^2 \cdot ||B||_F^2$.
- In order to obtain an X with smaller variance, we can do s independent trials to obtain matrices $X_1, ..., X_s$ and take an average. That is $X = \frac{X_1 + ... + X_s}{s}$.
- <u>Claim</u>: For such an X, $Var[X] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}$.
- Let $k_1, ..., k_s$ denote the k's chosen in each trial. Then $X = \frac{1}{s}\left(\frac{A(:,k_1)B(k_1,:)}{p_{k_1}} + ... + \frac{A(:,k_1)B(k_1,:)}{p_{k_1}}\right)$.
- Let C be the matrix with columns $\frac{A(:,k_1)}{\sqrt{sp_{k_1}}}, ..., \frac{A(:,k_s)}{\sqrt{sp_{k_s}}}$ and R be matrix with rows $\frac{B(k_1,:)}{\sqrt{sp_{k_1}}}, ..., \frac{B(k_s,:)}{\sqrt{sp_{k_s}}}$. Then $X = CR$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Let $A(:, k)$ denote the k^{th} column of A and $A(k, :)$ denote the k^{th} row.
- We can write the product AB as $AB = \sum_{k=1}^{n} A(:, k)B(k, :)$. Note that $A(:, k)B(k, :)$ is an $m \times p$ matrix for any k.
- Consider a random variable z that takes value in the set $\{1, ..., n\}$ and let $p_k = \mathbf{Pr}[z = k]$.
- Let $X = \frac{A(:,z)B(z,:)}{p_z}$.
- Claim: $\mathbf{E}[X] = AB$.
- We are interested in the quantity $\mathbf{E}[||AB - X||_F^2]$ which may be interpreted as the sum of variances of entries of X. Let us call this $Var[X]$.
- For ease of calculations let us use $p_k = ||A(:, k)||^2$. This gives $Var[X] \leq ||A||_F^2 \sum_k ||B(k, :)||^2 = ||A||_F^2 \cdot ||B||_F^2$.
- In order to obtain an X with smaller variance, we can do s independent trials to obtain matrices $X_1, ..., X_s$ and take an average. That is $X = \frac{X_1 + ... + X_s}{s}$.
- Claim: For such an X, $Var[X] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}$.
- Let $k_1, ..., k_s$ denote the k's chosen in each trial. Then $X = \frac{1}{s}\left(\frac{A(:,k_1)B(k_1,:)}{p_{k_1}} + ... + \frac{A(:,k_1)B(k_1,:)}{p_{k_1}}\right)$.
- Let C be the matrix with columns $\frac{A(:,k_1)}{\sqrt{sp_{k_1}}}, ..., \frac{A(:,k_s)}{\sqrt{sp_{k_s}}}$ and R be matrix with rows $\frac{B(k_1,:)}{\sqrt{sp_{k_1}}}, ..., \frac{B(k_s,:)}{\sqrt{sp_{k_s}}}$. Then $X = CR$.
- Claim: $\mathbf{E}[CC^T] = AA^T$ and $\mathbf{E}[R^T R] = B^T B$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

- Here is a nice summary of the entire discussion in terms of a usable theorem.

Theorem

Suppose A is an $m \times n$ matrix and B is an $n \times p$ matrix. The product AB can be estimated by CR, where C is an $m \times s$ matrix consisting of s columns of A picked according to length-squared distribution and scaled to satisfy $\mathbf{E}[CC^T] = AA^T$ and R is the $s \times p$ matrix consisting of the corresponding rows of B scaled to satisfy $\mathbf{E}[R^T R] = B^T B$. The error is bounded by:

$$\mathbf{E}[||AB - CR||_F^2] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}.$$

Thus to ensure $\mathbf{E}[||AB - CR||_F^2] \leq \varepsilon^2 ||A||_F^2 \cdot ||B||_F^2$, it suffices to make $s \geq \frac{1}{\varepsilon^2}$.

- Note that if $\varepsilon = \Omega(1)$, so $s \in O(1)$, then the multiplication CR can be performed in time $O(mp)$.

Problem

Given an $m \times n$ matrix A and an $n \times p$ matrix B, design an algorithm to compute AB.

Theorem

Suppose A is an $m \times n$ matrix and B is an $n \times p$ matrix. The product AB can be estimated by CR, where C is an $m \times s$ matrix consisting of s columns of A picked according to length-squared distribution and scaled to satisfy $\mathbf{E}[CC^T] = AA^T$ and R is the $s \times p$ matrix consisting of the corresponding rows of B scaled to satisfy $\mathbf{E}[R^T R] = B^T B$. The error is bounded by:

$$\mathbf{E}[||AB - CR||_F^2] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}.$$

Thus to ensure $\mathbf{E}[||AB - CR||_F^2] \leq \varepsilon^2 ||A||_F^2 \cdot ||B||_F^2$, it suffices to make $s \geq \frac{1}{\varepsilon^2}$.

- Note that if $\varepsilon = \Omega(1)$, so $s \in O(1)$, then the multiplication CR can be performed in time $O(mp)$.
- Let us analyse the circumstances under which the above theorem may be useful (not useful).

Theorem

Suppose A is an $m \times n$ matrix and B is an $n \times p$ matrix. The product AB can be estimated by CR, where C is an $m \times s$ matrix consisting of s columns of A picked according to length-squared distribution and scaled to satisfy $\mathbf{E}[CC^T] = AA^T$ and R is the $s \times p$ matrix consisting of the corresponding rows of B scaled to satisfy $\mathbf{E}[R^TR] = B^TB$. The error is bounded by:

$$\mathbf{E}[||AB - CR||_F^2] \leq \frac{||A||_F^2 \cdot ||B||_F^2}{s}.$$

Thus to ensure $\mathbf{E}[||AB - CR||_F^2] \leq \varepsilon^2 ||A||_F^2 \cdot ||B||_F^2$, it suffices to make $s \geq \frac{1}{\varepsilon^2}$.

- Note that if $\varepsilon = \Omega(1)$, so $s \in O(1)$, then the multiplication CR can be performed in time $O(mp)$.
- Let us analyse the circumstances under which the above theorem may be useful (not useful).
- Let $A = I$ and $B = A^T$. So, $||AA^T||_F^2 = n$ and $\frac{||A||_F^2 \cdot ||B||_F^2}{s} = \frac{n^2}{s}$.
- What this means is that s needs to be greater than n in order to give better approximation than the trivial zero matrix.
- In general, it will be useful exercise to examine the situations under which the sampling algorithm provides better approximation than the trivial zero matrix whose error is $||AA^T||_F^2$.

- Let us analyse the circumstances under which the above theorem may be useful (not useful).
- Let $A = I$ and $B = A^T$. So, $||AA^T||_F^2 = n$ and $\frac{||A||_F^2 \cdot ||B||_F^2}{s} = \frac{n^2}{s}$.
- What this means is that s needs to be greater than n in order to give better approximation than the trivial zero matrix.
- In general, it will be useful exercise to examine the situations under which the sampling algorithm provides better approximation than the trivial zero matrix whose error is $||AA^T||_F^2$.
  - Claim 1: $||AA^T||_F^2 = \sum_t \sigma_t^4$.
  - Claim 2: $||A||_F^2 = \sum_t \sigma_t^2$.
  - So, $\mathbf{E}[||AA^T - CR||_F^2] \leq ||AA^T||_F^2$ provided $s \geq \frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4}$.

- Let us analyse the circumstances under which the above theorem may be useful (not useful).
- Let $A = I$ and $B = A^T$. So, $||AA^T||_F^2 = n$ and $\frac{||A||_F^2 \cdot ||B||_F^2}{s} = \frac{n^2}{s}$.
- What this means is that s needs to be greater than n in order to give better approximation than the trivial zero matrix.
- In general, it will be useful exercise to examine the situations under which the sampling algorithm provides better approximation than the trivial zero matrix whose error is $||AA^T||_F^2$.
  - Claim 1: $||AA^T||_F^2 = \sum_t \sigma_t^4$.
  - Claim 2: $||A||_F^2 = \sum_t \sigma_t^2$.
  - So, $\mathbf{E}[||AA^T - CR||_F^2] \leq ||AA^T||_F^2$ provided $s \geq \frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4}$.
  - Claim 3: If $rank(A) = r$, then $\frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4} \leq r$ and s needs to be at least r.
    - This means that if A is full rank, then sampling will not gain us anything.

- Let us analyse the circumstances under which the above theorem may be useful (not useful).
- Let $A = I$ and $B = A^T$. So, $||AA^T||_F^2 = n$ and $\frac{||A||_F^2 \cdot ||B||_F^2}{s} = \frac{n^2}{s}$.
- What this means is that s needs to be greater than n in order to give better approximation than the trivial zero matrix.
- In general, it will be useful exercise to examine the situations under which the sampling algorithm provides better approximation than the trivial zero matrix whose error is $||AA^T||_F^2$.
  - Claim 1: $||AA^T||_F^2 = \sum_t \sigma_t^4$.
  - Claim 2: $||A||_F^2 = \sum_t \sigma_t^2$.
  - So, $\mathbf{E}[||AA^T - CR||_F^2] \leq ||AA^T||_F^2$ provided $s \geq \frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4}$.
  - Claim 3: If $rank(A) = r$, then $\frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4} \leq r$ and s needs to be at least r.
    - This means that if A is full rank, then sampling will not gain us anything.
  - Claim 4: If there are small constants c and p such that $\sum_{t=1}^{p} \sigma_t^2 \geq \frac{\sum_t \sigma_t^2}{c}$, then $\frac{(\sum_t \sigma_t^2)^2}{\sum_t \sigma_t^4} \leq c^2 p$.

Sketching: CUR decomposition

- Goal: Create a sketch of a given large $m \times n$ matrix A with respect to the 2-norm.
- We already talked about this while discussing SVD. So, why are we addressing this question again?

- Goal: Create a sketch of a given large $m \times n$ matrix A with respect to the 2-norm.
- We already talked about this while discussing SVD. So, why are we addressing this question again?
  - The SVD computation was in the batch setting. In the current low-space context, we want algorithms that are space efficient.
  - Interpolative approximation: The sketch involves a subset of (scaled) rows and columns of the original matrix A. This is useful in many contexts where the rows and columns have specific interpretation and preserving them is important.

- Goal: Create a sketch of a given large $m \times n$ matrix A with respect to the 2-norm.
- We already talked about this while discussing SVD. So, why are we addressing this question again?
  - The SVD computation was in the batch setting. In the current low-space context, we want algorithms that are space efficient.
  - Interpolative approximation: The sketch involves a subset of (scaled) rows and columns of the original matrix A. This is useful in many contexts where the rows and columns have specific interpretation and preserving them is important.
- Here is what we plan to do:
  - Sample s columns of A as per length squared distribution and each column is scaled so that if a column k is picked, then it is scaled by $\frac{1}{\sqrt{sp_k}}$. Let C be the $m \times s$ matrix of such (scaled) columns.
  - Similarly, sample r rows of A as per length squared distribution and each row is scaled so that if a row k is picked, then it is scaled by $\frac{1}{\sqrt{rp_k}}$. Let R be the $r \times n$ matrix of such (scaled) rows.
  - From C and R find an $s \times r$ matrix U such that $A \approx CUR$.

- Goal: Create a sketch of a given large $m \times n$ matrix A with respect to the 2-norm.
- We already talked about this while discussing SVD. So, why are we addressing this question again?
  - The SVD computation was in the batch setting. In the current low-space context, we want algorithms that are space efficient.
  - Interpolative approximation: The sketch involves a subset of (scaled) rows and columns of the original matrix A. This is useful in many contexts where the rows and columns have specific interpretation and preserving them is important.
- Here is what we plan to do:
  - Sample s columns of A as per length squared distribution and each column is scaled so that if a column k is picked, then it is scaled by $\frac{1}{\sqrt{sp_k}}$. Let C be the $m \times s$ matrix of such (scaled) columns.
  - Similarly, sample r rows of A as per length squared distribution and each row is scaled so that if a row k is picked, then it is scaled by $\frac{1}{\sqrt{rp_k}}$. Let R be the $r \times n$ matrix of such (scaled) rows.
  - From C and R find an $s \times r$ matrix U such that $A \approx CUR$.
- The notion of similarity ($\approx$) that we are interested in is the 2-norm since in many cases we would want to create a sketch for multiplying A with unit vectors. In case $A \approx CUR$, then the vector multiplication costs $O(ms + sr + rn)$ which is small is r and s are small.

- Here is what we plan to do:
  - Sample s columns of A as per length squared distribution and each column is scaled so that if a column k is picked, then it is scaled by $\frac{1}{\sqrt{sp_k}}$. Let C be the $m \times s$ matrix of such (scaled) columns.
  - Similarly, sample r rows of A as per length squared distribution and each row is scaled so that if a row k is picked, then it is scaled by $\frac{1}{\sqrt{rp_k}}$. Let R be the $r \times n$ matrix of such (scaled) rows.
  - From C and R find an $s \times r$ matrix U such that $A \approx CUR$.
- We will define a matrix P (that depends on matrix R) using which we will define U.

Defining matrix P

$$P = \begin{cases} R^T(RR^T)^{-1}R, & \text{if } RR^T \text{ is invertible} \\ R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R, & rank(RR^T) = \ell \ \& \ R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T \end{cases}$$

Here $R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T$ is the SVD of R.

- We will define a matrix P (that depends on matrix R) using which we will define U.

Defining matrix P

$$P = \begin{cases} R^T(RR^T)^{-1}R, & \text{if } RR^T \text{ is invertible} \\ R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R, & rank(RR^T) = \ell \ \&\ R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T \end{cases}$$

Here $R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T$ is the SVD of R.

Lemma

The matrix P defined above satisfies the following properties:

1. *For every vector $\mathbf{x}$ of the form $\mathbf{x} = R^T\mathbf{y}$, $P\mathbf{x} = \mathbf{x}$. That is, it acts like an identity matrix on the row space of R.*
2. *For every $\mathbf{x}$ that is orthogonal to the row space of R, $P\mathbf{x} = 0$.*

- We will define a matrix P (that depends on matrix R) using which we will define U.

Defining matrix P

$$P = \begin{cases} R^T(RR^T)^{-1}R, & \text{if } RR^T \text{ is invertible} \\ R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R, & rank(RR^T) = \ell \text{ \& } R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T \end{cases}$$

Here $R = \sum_{t=1}^{\ell} \sigma_t \mathbf{u_t}\mathbf{v}_t^T$ is the SVD of R.

Lemma

The matrix P defined above satisfies the following properties:

1. *For every vector $\mathbf{x}$ of the form $\mathbf{x} = R^T\mathbf{y}$, $P\mathbf{x} = \mathbf{x}$. That is, it acts like an identity matrix on the row space of R.*
2. *For every $\mathbf{x}$ that is orthogonal to the row space of R, $P\mathbf{x} = 0$.*

Proof sketch

- Case 1: RR^T is invertible:
  - For any $\mathbf{x} = R^T\mathbf{y}$, $P\mathbf{x} = R^T(RR^T)^{-1}R\mathbf{x} = R^T(RR^T)^{-1}RR^T\mathbf{y} = R^T\mathbf{y} = \mathbf{x}$.
  - For $\mathbf{x}$ orthogonal to every row of R, we have $R\mathbf{x} = 0$ and hence $P\mathbf{x} = 0$.
- Case 2: $rank(RR^T) = \ell < r$
  - $R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R = \sum_{t=1}^{\ell} \mathbf{v}_t\mathbf{v}_t^T$.

- We will define a matrix P (that depends on matrix R) using which we will define U.

Defining matrix P

$$P = \begin{cases} R^T(RR^T)^{-1}R, & \text{if } RR^T \text{ is invertible} \\ R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R, & rank(RR^T) = \ell \ \&\ R = \sum_{t=1}^{\ell}\sigma_t \mathbf{u_t}\mathbf{v}_t^T \end{cases}$$

Here $R = \sum_{t=1}^{\ell}\sigma_t \mathbf{u_t}\mathbf{v}_t^T$ is the SVD of R.

Lemma

The matrix P defined above satisfies the following properties:

1. *For every vector $\mathbf{x}$ of the form $\mathbf{x} = R^T\mathbf{y}$, $P\mathbf{x} = \mathbf{x}$. That is, it acts like an identity matrix on the row space of R.*
2. *For every $\mathbf{x}$ that is orthogonal to the row space of R, $P\mathbf{x} = 0$.*

- Claim 1: $\mathbf{E}[||A - AP||_2^2] \leq \frac{||A_F||^2}{\sqrt{r}}$.

- We will define a matrix P (that depends on matrix R) using which we will define U.

Defining matrix P

$$P = \begin{cases} R^T(RR^T)^{-1}R, & \text{if } RR^T \text{ is invertible} \\ R^T\left(\sum_{t=1}^{\ell} \frac{1}{\sigma_t^2}\mathbf{u_t}\mathbf{u_t^T}\right)R, & rank(RR^T) = \ell \;\&\; R = \sum_{t=1}^{\ell}\sigma_t \mathbf{u_t}\mathbf{v}_t^T \end{cases}$$

Here $R = \sum_{t=1}^{\ell}\sigma_t \mathbf{u_t}\mathbf{v}_t^T$ is the SVD of R.

Lemma

The matrix P defined above satisfies the following properties:

1. *For every vector $\mathbf{x}$ of the form $\mathbf{x} = R^T\mathbf{y}$, $P\mathbf{x} = \mathbf{x}$. That is, it acts like an identity matrix on the row space of R.*
2. *For every $\mathbf{x}$ that is orthogonal to the row space of R, $P\mathbf{x} = 0$.*

- Claim 1: $\mathbf{E}[||A - AP||_2^2] \leq \frac{||A_F||^2}{\sqrt{r}}$.
- Sampling s columns from A and taking the same rows from P, leads to an expression of the form CUR. Using our multiplication result, we get:

$$\mathbf{E}[||AP - CUR||_2^2] \leq \mathbf{E}[||AP - CUR||_F^2] \leq \frac{||A||_F^2 \cdot ||P||_F^2}{s} \leq \frac{r}{s}||A||_F^2.$$

- Finally, using the triangle inequality we get that:

$$\mathbf{E}[||A - CUR||_2^2] \leq ||A||_F^2\left(\frac{2}{\sqrt{r}} + \frac{2r}{s}\right).$$

- The entire discussion is summarized in the following theorem.

Theorem

Let A be an $n \times m$ matrix and r and s be positive integers. Let C be an $m \times s$ matrix of s columns of A picked according to length squared sampling and let R be a matrix of r rows of A picked according to length squared sampling. Then, we can find from C and R an $s \times r$ matrix U so that

$$\mathbf{E}[||A - CUR||_2^2] \leq ||A||_F^2 \left(\frac{2}{\sqrt{r}} + \frac{2r}{s} \right).$$

- Using $r = \Theta(1/\varepsilon^2)$ and $s = \Theta(1/\varepsilon^3)$, we get that the LHS is at most $O(\varepsilon)||A||_F^2$.

End