Finding structure with randomness - Probabilistic Algorithms for Constructing Approximate Matrix Decomposition

Oussama Ennafii

Ecole Normale Superieur de Cachan

December 26, 2014

Problematic:

Issue to solve:

Traditional low rank approximation algorithms such as the QR decomposition and SVD are not addapted to very huge or inaccurate matrices.

We need to find a framework to apply to these kinds of reccurent problems.

Two stage approximation framework:

Two stage approximation framework

- Stage A: Compute an orthornormal low rank basis $\mathbf{Q}$ such that, $\mathbf{A} \approx \mathbf{Q}\mathbf{Q}^{*}\mathbf{A}$
- Stage B: Compute the matrix factorisation on $\mathbf{B} = \mathbf{Q}^{T}\mathbf{A}$

The randomization, in Stage A, permits to span the range of A very efficiently.

Problem formulation:

The fixed precision problem

Given $\mathbf{A}$ and ϵ, find $\mathbf{Q}$ s.t.

$$\|\mathbf{A} - \mathbf{Q}\mathbf{Q}^T\mathbf{A}\| \leq \epsilon$$

The fixed rank problem

Given $\mathbf{A}$ and k, seek $\mathbf{B}$ s. t.

$$\|\mathbf{A} - \mathbf{Q}\mathbf{Q}^T\mathbf{A}\| \approx \min_{rank(\mathbf{X}) \leq k} \|\mathbf{A} - \mathbf{X}\|$$

Stage A:

The proto-algorithm:

The proto-algorithm

1. Draw an n x (k + p) random matrix **O**
2. Form the matrix **Y** = **AO**
3. Construct a matrix **Q** whose columns form an othonormal basis for the range of **Y**

Stage A:

Randomized Range Finder:

Stage A - Randomized Range Finder

1. Draw an n x l standard Gaussian matrix **O**
2. Form the m x l matrix **Y** = **AO**
3. Construct the m x l matrix **Q** using the QR factorization of: **Y** = **QR**

The number of flops necessary for this algorithm is:

$$lnT_{rand} + lT_{mult} + l^2 m$$

Stage A:

Randomized Power Iteration:

Randomized Power Iteration:

1. Draw an n x l standard Gaussian matrix $\mathbf{O}$
2. For the m x l matrix $\mathbf{Y} = (\mathbf{A}\mathbf{A}^*)^q \mathbf{A}\mathbf{O}$ via alternative application of $\mathbf{A}$ and $\mathbf{A}^q$
3. Construct the m x l matrix $\mathbf{Q}$ using the QR decomposition: $\mathbf{Y} = \mathbf{Q}\mathbf{R}$

Stage A:

Fast Randomized Range Finder:

Fast Randomized Range Finder

1. Draw an n x l SRFT matrix **O**
2. Form the m x l matrix **Y** = **AO**
3. Construct the m x l matrix **Q** using the QR factorization of: **Y** = **QR**

An SFRT is :

$$\mathbf{O} = \sqrt{\frac{n}{l}}\mathbf{DFR}$$

where: **D** is a n x n diagonal matrix whose entries are random variables, distributed on the complex unit circle **F** is the n x n unitary discrete Fourier transform and **R** is an n x l matrix whose l column are drawn from the n x n identity matrix.

Stage B:

Direct SVD:

Direct SVD

1. Form the matrix $\mathbf{B} = \mathbf{Q}^*\mathbf{A}$
2. Compute the SVD of the matrix $\mathbf{B} = \tilde{\mathbf{U}}\mathbf{S}\mathbf{V}^*$
3. Form the orthonormal matrix $\mathbf{U} = \mathbf{Q}\tilde{\mathbf{U}}$

Theory:

Proto-Algorithm Error bound:

We use the notations:

$$A = U \begin{pmatrix} \Sigma_1 & \\ & \Sigma_2 \end{pmatrix} \begin{pmatrix} V_1^* \\ V_2^* \end{pmatrix}$$

and

$$\Omega_1 = V_1^* \Omega \; and \; \Omega_2 = V_2^* \Omega$$

Theorem:

Assuming that Ω_1 has full row rank,

$$||(I - P_Y)A||^2 \leq ||\Sigma_2||^2 + ||\Sigma_2 \Omega_2 \Omega_1^{\dagger}|| \tag{1}$$

Theory:

Error bound on the randomized range finder:

Theorem:

We keep the same notations as before. $k, p \geq 2$ and $k + p \leq min(m, n)$:

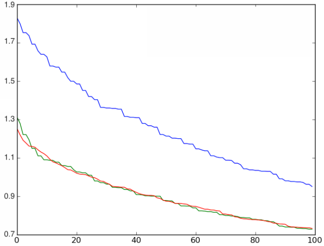
$$\mathbb{E}\|(I - P_Y)A\|_F \leq (1 + \frac{k}{p-1})^{1/2}(\sum_{j>k} \sigma_j^2)^{1/2} \tag{2}$$

and for the spectral norm :

$$\mathbb{E}\|(I - P_Y)A\| \leq (1 + \frac{k}{p-1})\sigma_{k+1} + \frac{e\sqrt{k+p}}{p}(\sum_{j>k} \sigma_j^2)^{1/2} \tag{3}$$

Numericals:

Image processing application:



- The random range finder(blue) runs for 4.5 sec.
- The random power iteration(green) runs for 9.8 sec.
- The fast random range finder(red) runs for 11 sec.

Conclusion: