IIT KHARAGPUR

IIT GANDHINAGAR

NPTEL ONLINE CERTIFICATION COURSES

NPTEL

Scalable Data Science

Lecture 15b: Modified QB + Linear Regression

Anirban Dasgupta

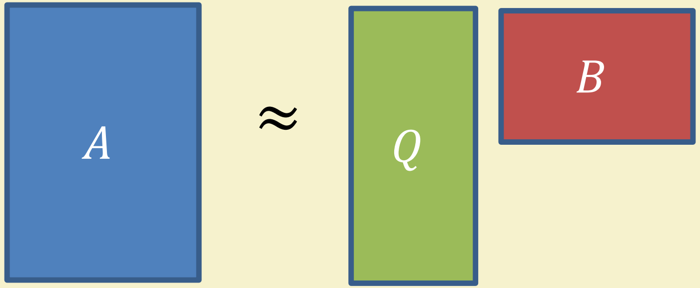
Computer Science and Engineering

IIT GANDHINAGAR

IIT Gandhinagar

Indian Institute of Technology Gandhinagar

QB decomposition



Can we find orthogonal $Q \in \Re^{n\times(k+p)}$ and B such that

$|A - QB|_F \approx |A - A_k|_F$ and/or $|A - QB|_F \approx |A - A_k|_2$

Also want $k + p = O(k)$

QB: prototype algorithm

Find $\Omega \in \Re^{d\times(k+p)}$ a random matrix

$Y = A\,\Omega$

Find Q = orthogonal basis for col(Y)

Return $(Q, Q^t A)$

Modified Algorithm

In reality, the bound $|A - QB|_F$ depends on the singular values $\{\sigma_{k+1}, \ldots, \}$

Bound improves if $\sum_{i \leq k} \sigma_i^2 \gg \sum_{i > k} \sigma_i^2$

Power scheme

In reality, the bound $|A - QB|_F$ depends on the singular values $\{\sigma_{k+1}, \dots, \}$

Bound improves if $\sum_{i \leq k} \sigma_i^2 \gg \sum_{i > k} \sigma_i^2$

To achieve this, we use $(AA^t)^P A = U\, \Sigma^{2P+1} V^t$

Modified Algorithm [HMK10]

function $[\mathbf{Q}, \mathbf{B}] = \texttt{randQB\_p}(\mathbf{A}, \ell, P)$

$\mathbf{\Omega} = \texttt{randn}(n, \ell)$.
$\mathbf{Q} = \texttt{orth}(\mathbf{A}\mathbf{\Omega})$.
for $j = 1 : P$
  $\mathbf{Q} = \texttt{orth}(\mathbf{A}^*\mathbf{Q})$.
  $\mathbf{Q} = \texttt{orth}(\mathbf{A}\mathbf{Q})$.
end for
$\mathbf{B} = \mathbf{Q}^*\mathbf{A}$

Total runtime = $O(nd\ell + n\ell^2)$

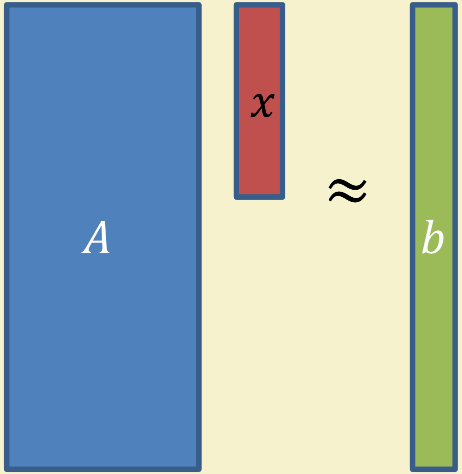
$\ell = k + p \approx k + 5$

$P = 1 \; or \; 2$ is sufficient

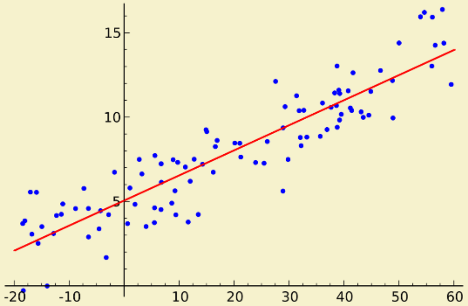
IIT Gandhinagar
Indian Institute of Technology Gandhinagar

NPTEL ONLINE CERTIFICATION COURSES

Linear Regression



Solve $\min_x |Ax - b|_2$



Finding the "best linear" relation that explains the target

Linear Regression

Overconstrained setting, $n \gg d$

$\rightarrow$ there is no x that satisfies $Ax = b$

Want to find "best" x such that $Ax \approx b$

Statistical interpretation: Find the best "unbiased estimator"

Geometric interpretation: orthogonal projection of b onto span(A)

Solving Linear Regression

Cholesky decomposition:

- find out upper triangular R such that $R^t R = A^t A$
- solve normal equations $R^t R x = A^t b$

QR decomposition:

- $A = QR$. Solve $Rx = Q^t b$

SVD:

- $A = U \Sigma V^t$, solve $x = V \Sigma^{-1} U^t b$

Solving Linear Regression

Normal equations:
$(A^t A)x = A^t b$

Cholesky decomposition:
- find out upper triangular R such that $R^t R = A^t A$
- solve normal equations $R^t R x = A^t b$

Time taken = $O(nd^2)$

QR decomposition:
- $A = QR$. Solve $Rx = Q^t b$

SVD:
- $A = U\Sigma V^t$, solve $x = V\Sigma^{-1} U^t b$

IIT Gandhinagar
Indian Institute of Technology Gandhinagar

NPTEL ONLINE CERTIFICATION COURSES

NPTEL

Prototype Randomized Algorithm

Input: $A \in \Re^{n \times d}$, $b \in \Re^{d}$

Create $\Omega \in \Re^{s \times n}$, random

Solve $\tilde{x} = \min_{\mathrm{x}} |\Omega A x \ - \Omega b|_2$

Prototype Randomized Algorithm

Input: $A \in \Re^{n \times d}, b \in \Re^{d}$

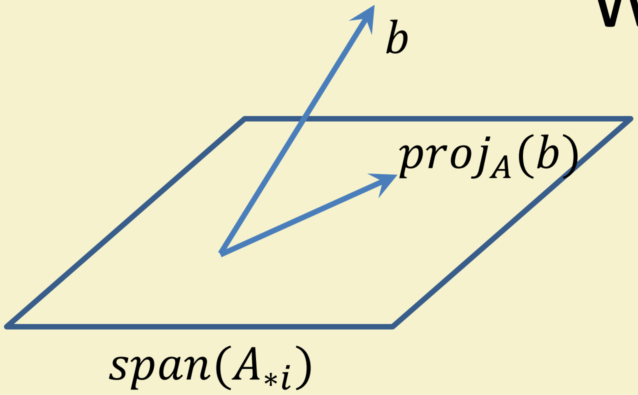
Create $\Omega \in \Re^{s \times n}$, random

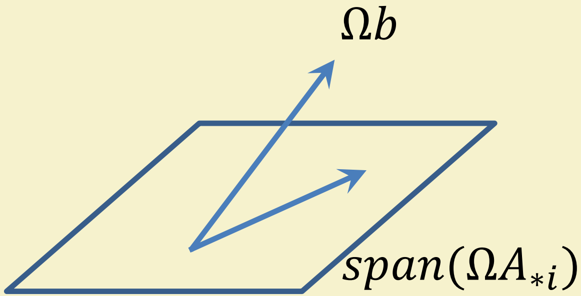
Solve $\tilde{x} = \min_{\mathrm{x}} |\Omega A x - \Omega b|_2$

Tentative claim: $|A\tilde{x} - b| \leq (1+\epsilon)|Ax^* - b|$

Running time: If $\Omega \sim N(0,1)$, then $O(nd^2)$

Why would it work?





Geometrically, solution depends on projection of b on the $span(A)$

$x = (A^t A)^{-1} A^t b = A^+ b$

Similarly, the optimal solution for the sampled problem

$\tilde{x} = (A^t \Omega^t \Omega A)^{-1} A^t \Omega^t \Omega b = (\Omega A)^+ S \Omega b$

Guarantee

At its core, uses a matrix concentration type inequality to show that

$|U(I \ - \Omega\Omega^t)U^t| \leq \epsilon \ , U =$ left singular vectors of A

Holds for S satisfying the JL lemma with (ϵ, δ). Not always more efficient.

Randomized Hadamard based projection

$\Omega = PHD \in \Re^{s\times n}$

$H = n \times n$ Hadamard matrix

$D =$ diagonal random ± 1

$P =$ can be just a sampling matrix now, i.e.

For each $t \in [1, s]$

$$P_{t*} = e_i \cdot \sqrt{n/s} \text{ with prob } 1/n$$

Running time

To ensure $|A\tilde{x} - b| \leq (1+\epsilon)|Ax^* - b|$, need $s \geq Cd \log(d)/\epsilon$

Time for projection = $O(nd \log n + sn)$

Time to solve = $O(sd^2)$

Overall time = $O\left(nd \log n + \frac{d^3 \log(n)}{\epsilon}\right)$, ignoring lower terms

Guarantees

1. $|A\tilde{x} \ - b| \le (1+\epsilon)|Ax^* \ - b|$

2. $|\tilde{x} \ - x^*| \le \sqrt{\epsilon}\,\kappa\left(\frac{1}{\gamma^2} - 1\right)^{\frac{1}{2}}|x|$

   $\kappa =$ condition number of A

   $\gamma = \frac{|UU^t b|}{|b|} =$ captures how much of b is present in span(A)

Extensions

- Can be extended using sparse JL so that theoretical runtime depends on sparsity
  - sorting out the stability issues is still an open question
- L2 regularized versions
  - Stated results work as it is, but we should be able to do better, recent results on sharper bounds
- BLENDENPIK : an implementation of randomized regression that beats decades old and optimized LAPACK routines for a thin rectangular matrices.

[Avron, Maymounkov, Toledo]

References:

- Primary reference
    - Lecture notes on Randomized Numerical Linear Algebra by Petros Drineas and Michael Mahoney, https://arxiv.org/abs/1712.08880

Thank You!!