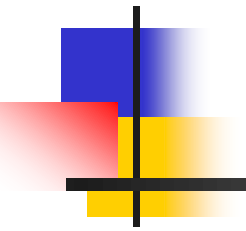


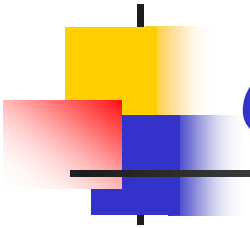
Leverage scores

Petros Drineas

Rensselaer Polytechnic Institute
Computer Science Department

To access my web page:

Google drineas



Overview

- **Least Squares problems**
  - Formulation and background
  - A sampling based approach: the leverage scores
- **The Column Subset Selection Problem (CSSP)**
  - Motivation, formulation, and the CX factorization
  - A sampling based approach: the leverage scores
- **Leverage scores and Effective Resistances**
  - Leverage scores vs effective resistances
  - Solving systems of linear equations on Laplacian matrices
- **Conclusions**

L_2 regression problems

$$\mathcal{Z}_2 = \min_{x \in \mathbb{R}^d} \|b - Ax\|_2 = \|b - A\hat{x}\|_2 \Rightarrow \underset{n \times d \ , \ n \gg d}{\begin{pmatrix} A \end{pmatrix}} \begin{pmatrix} \hat{x} \end{pmatrix} \approx \begin{pmatrix} b \end{pmatrix}$$

We are interested in over-constrained L_2 regression problems, $n \gg d$.

(Under-constrained problem, n << d, can also be handled in a similar manner.)

Typically, there is no x such that $Ax = b$.

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

Exact solution to L_2 regression

Cholesky Decomposition:

If A is full rank and well-conditioned,

decompose $A^TA = R^TR$, where R is upper triangular, and

solve the normal equations: $R^TRx = A^Tb$.

QR Decomposition:

Slower but numerically stable, esp. if A is rank-deficient.

Write $A = QR$, and solve $Rx = Q^Tb$.

Projection of b on the subspace spanned by the columns of A

$$\begin{aligned} \mathcal{Z}_2^2 &= ||b||_2^2 - ||AA^+b||_2^2 \\ \hat{x} &= A^+b \end{aligned}$$

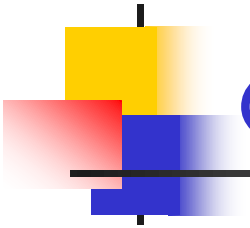
Pseudoinverse of A

Singular Value Decomposition:

Most expensive, but best if A is very ill-conditioned.

Write $A = U\Sigma V^T$, in which case: $\mathbf{x_{OPT} = A^+b = V\Sigma^{-1}U^Tb}$.

Complexity is $O(nd^2)$, but constant factors differ.



Questions …

$$\mathcal{Z}_2 = \min_{x \in \mathbb{R}^d} \|b - Ax\|_2 = \|b - A\hat{x}\|_2$$

Approximation algorithms:

Can we approximately solve L_2 regression faster than "exact" methods?

(Sarlos FOCS 2006, Drineas, Mahoney, Muthukrishnan, & Sarlos NumMath 2011)

This talk: Core-sets (or induced sub-problems):

Can we find a small set of constraints such that solving the L_2 regression on those constraints gives an approximation to the original problem?

If we can find those constraints efficiently, then we also get faster algorithms for L_2 regression problems.

Algorithm: Sampling for L_2 regression

(Drineas, Mahoney, & Muthukrishnan SODA 2006)

$$Z_2 = \min_{x \in \mathbb{R}^d} \|b - Ax\|_2 = \|b - A\hat{x}\|_2$$

$$\begin{pmatrix} \\ A \\ \\ \end{pmatrix} \begin{pmatrix} \\ \hat{x} \\ \\ \end{pmatrix} \approx \begin{pmatrix} \\ b \\ \\ \end{pmatrix}$$

$n \times d, \quad n >> d$

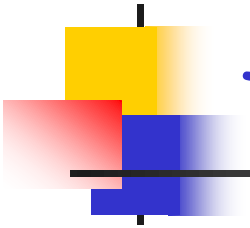
Algorithm

1. Fix a set of probabilities p_i, i=1…n, summing up to 1.
2. Pick the i-th row of A and the i-th element of b with probability

   $$\min \{1, rp_i\},$$

   and rescale both by $(1/\min\{1,rp_i\})^{1/2}$.
3. Solve the induced problem.

Note: in expectation, at most r rows of A and r elements of b are kept.



The result

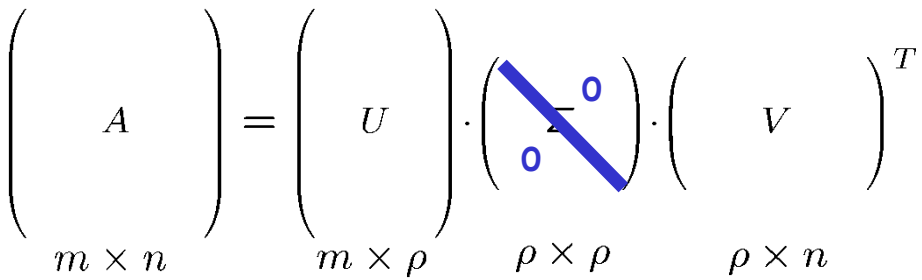
If the p_i **satisfy a condition**, then with probability at least 1-δ,

$$\mathcal{Z}_2 \leq \|A\hat{x}_s - b\|_2 \leq (1+\epsilon)\,\mathcal{Z}_2$$

The sampling complexity is

$$r = O(d\log(d)\log(1/\delta)/\epsilon^2)$$

SVD: formal definition



ρ: rank of A

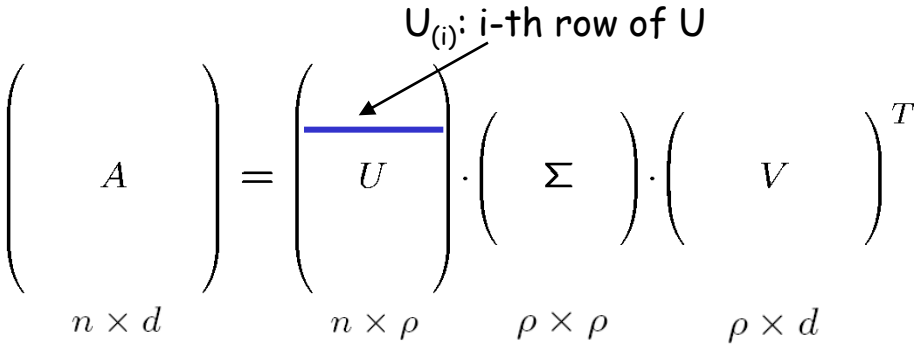
U (V): orthogonal matrix containing the left (right) singular vectors of A.

Σ: diagonal matrix containing the singular values of A.

Let σ_1 , σ_2 , … , σ_ρ be the entries of Σ.

Standard methods for the SVD take $O(\min\{mn^2 , m^2n\})$ time.

Notation



ρ: rank of A (at most d, since we assume n > d)

U: orthogonal matrix containing the left singular vectors of A.

Leverage scores

The condition that the p_i must satisfy is, for some $\beta \in (0,1]$:

lengths of **rows** of matrix of left singular vectors of A

$$p_i \geq \frac{\beta \left\| U_{(i)} \right\|_2^2}{\sum_{i=1}^{n} \left\| U_{(i)} \right\|_2^2} = \frac{\beta \left\| U_{(i)} \right\|_2^2}{d}$$

Notes:

- *$O(nd^2)$ time suffices* (to compute probabilities and to construct a core-set).

Leverage scores

The condition that the p_i must satisfy is, for some $\beta \in (0,1]$:

lengths of **rows** of matrix of left singular vectors of A

$$p_i \geq \frac{\beta \left\| U_{(i)} \right\|_2^2}{\sum_{i=1}^{n} \left\| U_{(i)} \right\|_2^2} = \frac{\beta \left\| U_{(i)} \right\|_2^2}{d}$$

Leverage scores

(useful in statistics for outlier detection)

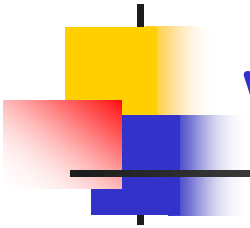
Notes:

- *$O(nd^2)$ time suffices* (to compute probabilities and to construct a core-set).
- Important question:

  Is $O(nd^2)$ necessary? Can we compute the p_i's, or construct a core-set, faster?

Better constructions (smaller coresets) exist, not using leverage scores.

(With C. Boutsidis and M. Magdon-Ismail, building upon Boutsidis, Drineas, & Magdon-Ismail FOCS 2011)



Why leverage scores

An old question:

Given an orthogonal matrix, sample a subset of its rows and argue that the resulting matrix is almost orthogonal.

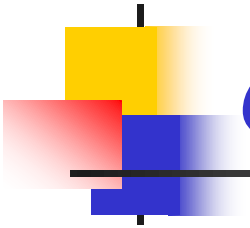
What if we are allowed to sample rows of an orthogonal matrix (scaled appropriately) **with respect to leverage scores**?

Then, in our case (n >> d), we can prove that:

$$\left\| U_A^T S^T S U_A - I \right\|_2 \leq \epsilon \qquad r = O(d \log d / \epsilon^2)$$

(see Drineas and Kannan FOCS 2001, Drineas, Kannan, and Mahoney 2006, Rudelson and Virshynin JACM 2006)

Current state of the art: matrix Chernoff/Bernstein bounds; for an empirical and theoretical evaluation see Ipsen & Wentworth ArXiv 2012.



Overview

- Least Squares problems
  - Formulation and background
  - A sampling based approach: the leverage scores
- **The Column Subset Selection Problem (CSSP)**
  - Motivation, formulation, and the CX factorization
  - A sampling based approach: the leverage scores
- Leverage scores and Effective Resistances
  - Leverage scores vs effective resistances
  - Solving systems of linear equations on Laplacian matrices
- Conclusions

SVD decomposes a matrix as…

$$\underset{m \times n}{\begin{pmatrix} \\ A \\ \\ \end{pmatrix}} \approx \underset{m \times k}{\begin{pmatrix} \\ U_k \\ \\ \end{pmatrix}} \underset{k \times n}{\begin{pmatrix} \\ X \\ \\ \end{pmatrix}}$$

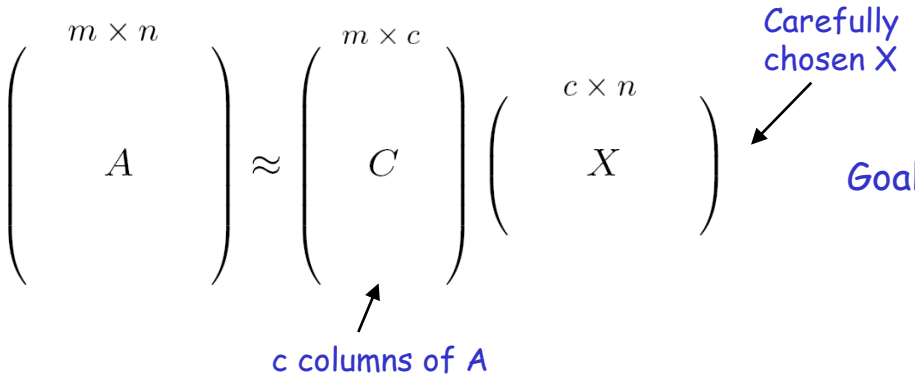
Top k left singular vectors

The SVD has strong optimality properties.

- It is easy to see that $X = U_k^T A$.
- SVD has strong optimality properties.
- The columns of U_k are linear combinations of up to all columns of A.

The CX decomposition

Drineas, Mahoney, & Muthukrishnan (2008) SIAM J Mat Anal Appl
Mahoney & Drineas (2009) PNAS



Goal: make (some norm) of A-CX small.

Why?

If A is a data matrix with rows corresponding to objects and columns to features, then selecting representative columns is equivalent to selecting representative features to capture the same structure as the top eigenvectors.

We want c as small as possible!

CX decomposition

$$\underset{m \times n}{\begin{pmatrix} \\ A \\ \\ \end{pmatrix}} \approx \underset{m \times c}{\begin{pmatrix} \\ C \\ \\ \end{pmatrix}} \underset{c \times n}{\begin{pmatrix} \\ X \\ \\ \end{pmatrix}}$$

c columns of *A*

Easy to prove that optimal X = C^+A. (C^+ is the Moore-Penrose pseudoinverse of *C*.)

Thus, the challenging part is to find good columns of *A* to include in *C*.

From a mathematical perspective, this is a hard combinatorial problem, closely related to the so-called Column Subset Selection Problem (CSSP).

The CSSP has been heavily studied in Numerical Linear Algebra.

Relative-error Frobenius norm bounds

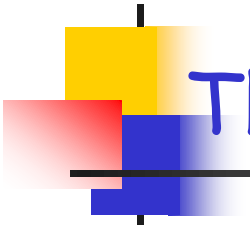
Drineas, Mahoney, & Muthukrishnan (2008) SIAM J Mat Anal Appl

Given an m-by-n matrix A, there exists an $O(mn^2)$ algorithm that picks

at most $O((k/\varepsilon^2) \log (k/\varepsilon))$ columns of A

such that with probability at least .9

$$\left\| A - CC^{\dagger}A \right\|_F \leq (1+\epsilon) \left\| A - A_k \right\|_F$$



The algorithm

Input: m-by-n matrix A,

$0 < \varepsilon < .5$, the desired accuracy

Output: C, the matrix consisting of the selected columns

Sampling algorithm

- Compute probabilities p_j summing to 1.
- Let $c = O((k/\varepsilon^2) \log (k/\varepsilon))$.
- In c i.i.d. trials pick columns of A, where in each trial the j-th column of A is picked with probability p_j.
- Let C be the matrix consisting of the chosen columns.

Subspace sampling (Frobenius norm)

$$\underset{m \times n}{\begin{pmatrix} \\ A_k \\ \\ \end{pmatrix}} = \underset{m \times k}{\begin{pmatrix} \\ U_k \\ \\ \end{pmatrix}} \cdot \underset{k \times k}{\begin{pmatrix} \\ \Sigma_k \\ \\ \end{pmatrix}} \cdot \underset{k \times n}{\begin{pmatrix} \\ V_k^T \\ \\ \end{pmatrix}}$$

V_k: orthogonal matrix containing the top k right singular vectors of A.

Σ_k: diagonal matrix containing the top k singular values of A.

Remark: The rows of V_k^T are orthonormal vectors, but its columns $(V_k^T)^{(i)}$ are not.

Subspace sampling (Frobenius norm)

$$\underset{m \times n}{\begin{pmatrix} \\ A_k \\ \\ \end{pmatrix}} = \underset{m \times k}{\begin{pmatrix} \\ U_k \\ \\ \end{pmatrix}} \cdot \underset{k \times k}{\begin{pmatrix} \\ \Sigma_k \\ \\ \end{pmatrix}} \cdot \underset{k \times n}{\begin{pmatrix} \\ V_k^T \\ \\ \end{pmatrix}}$$

V_k: orthogonal matrix containing the top k right singular vectors of A.

Σ_k: diagonal matrix containing the top k singular values of A.

Remark: The rows of V_k^T are orthonormal vectors, but its columns $(V_k^T)^{(i)}$ are not.

Subspace sampling in $O(mn^2)$ time

Leverage scores

(useful in statistics for outlier detection)

$$p_j = \frac{\left\| \left(V_k^T\right)^{(j)} \right\|_2^2}{k}$$

Normalization s.t. the p_j sum up to 1

Leverage scores: human genetics data

Single Nucleotide Polymorphisms: the most common type of genetic variation in the genome across different individuals.

They are known locations at the human genome where two alternate nucleotide bases (alleles) are observed (out of A, C, G, T).

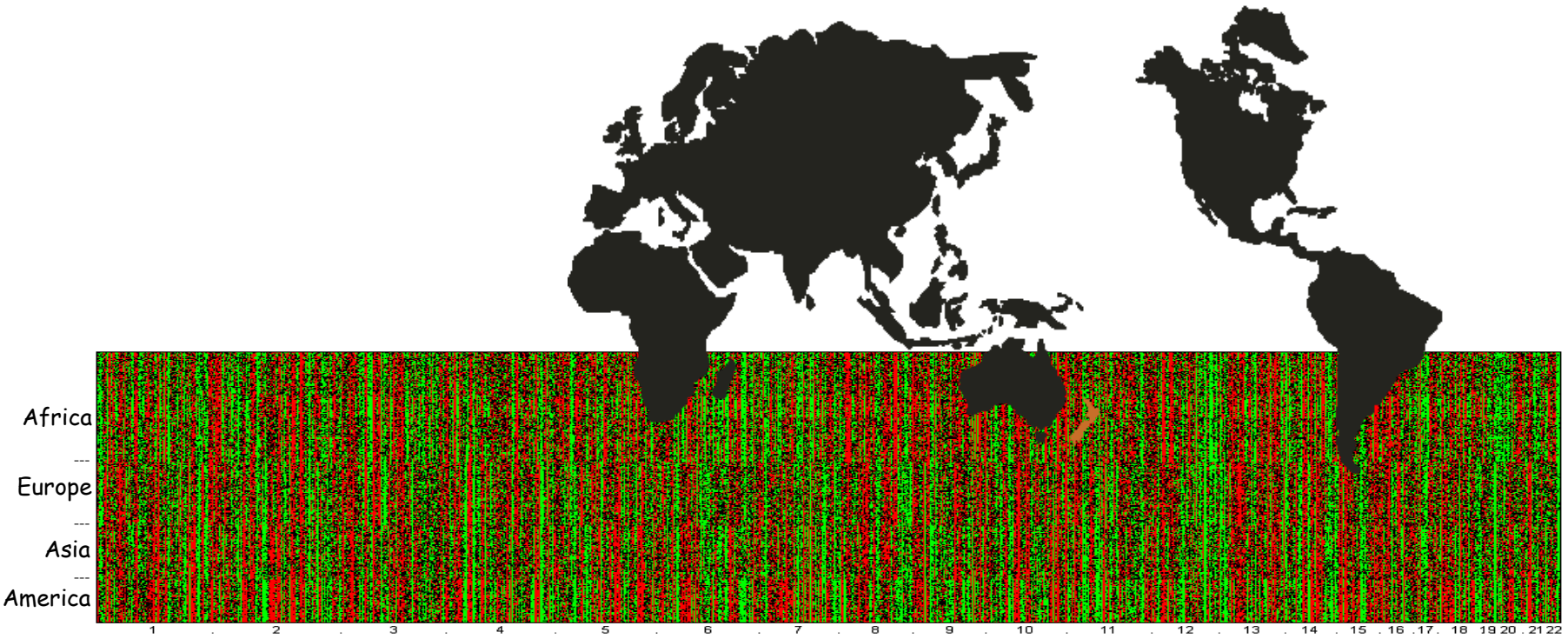
Matrices including thousands of individuals and hundreds of thousands if SNPs are available.

Worldwide data

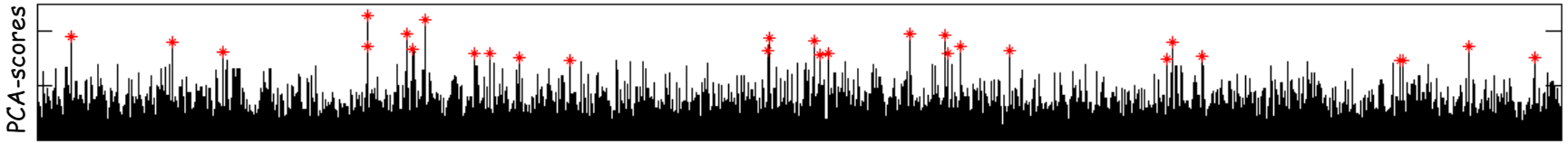
274 individuals, 12 populations, ~10,000 SNPs

Shriver et al. (2005) Hum Genom

Leverage scores of the columns of the 274-by-10,000 SNP matrix



* top 30 PCA-correlated SNPs



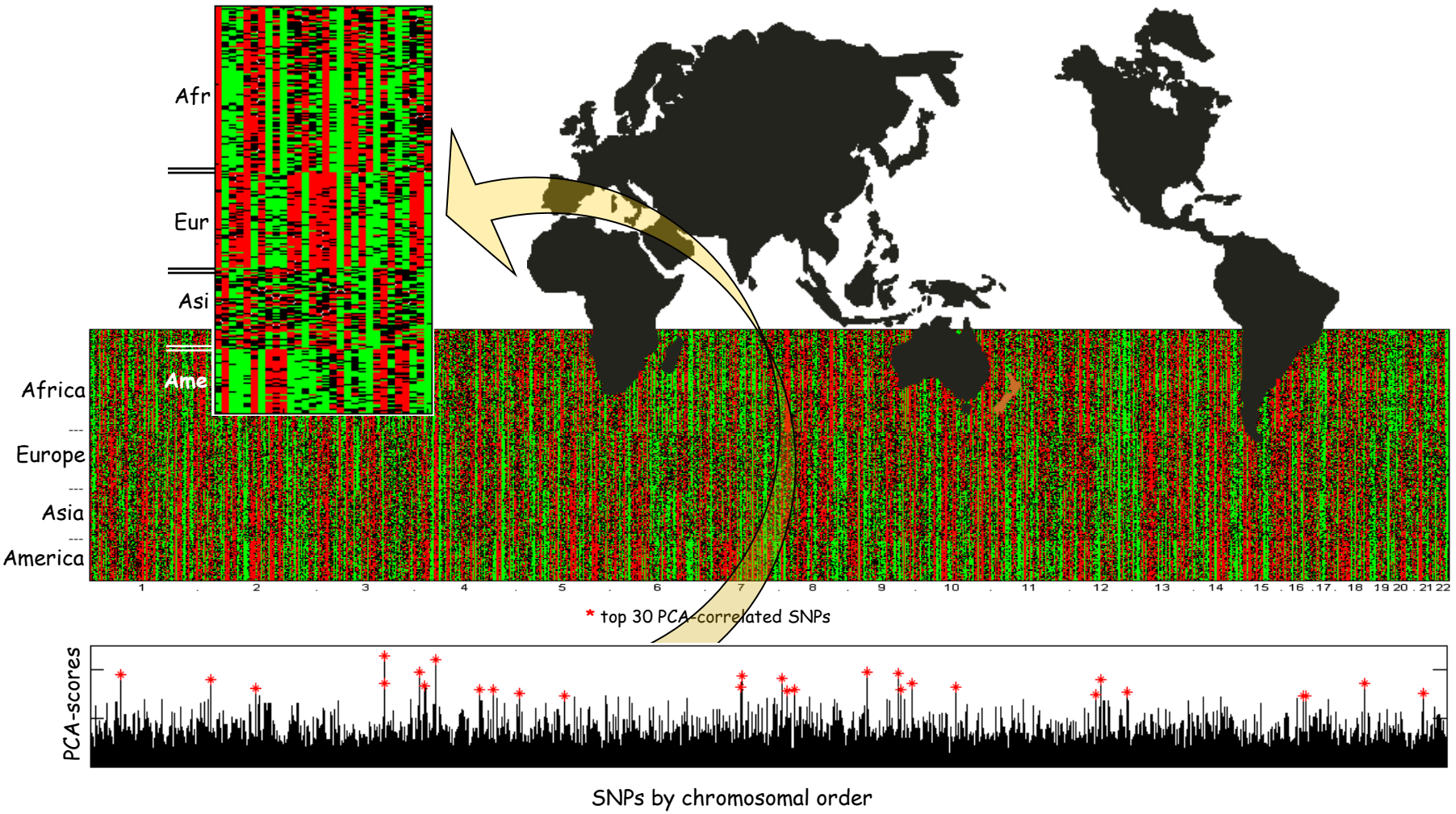
SNPs by chromosomal order

Paschou et al (2007; 2008) PLoS Genetics

Paschou et al (2010) J Med Genet

Drineas et al (2010) PLoS One

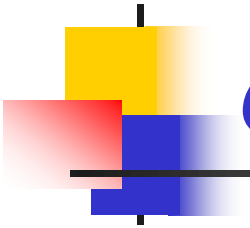
Selecting ancestry informative SNPs for individual assignment to four continents (Africa, Europe, Asia, America)



Paschou et al (2007; 2008) PLoS Genetics

Paschou et al (2010) J Med Genet

Drineas et al (2010) PLoS One



Overview

- Least Squares problems
  - Formulation and background
  - A sampling based approach: the leverage scores
- The Column Subset Selection Problem (CSSP)
  - Motivation, formulation, and the CX factorization
  - A sampling based approach: the leverage scores
- **Leverage scores and Effective Resistances**
  - Leverage scores vs effective resistances
  - Solving systems of linear equations on Laplacian matrices
- Conclusions

Leverage scores & effective resistances

Consider a weighted (positive weights only!) undirected graph G and let L be the Laplacian matrix of G.

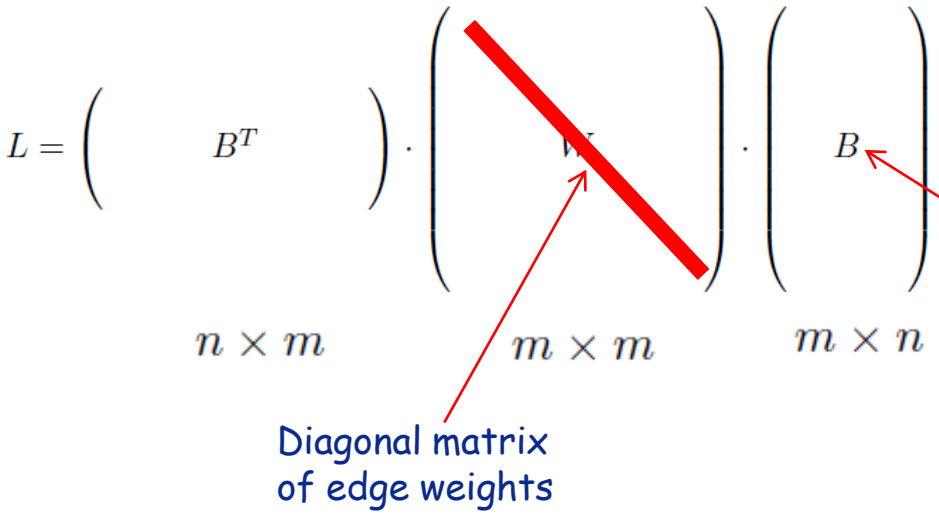
Assuming n vertices and $m > n$ edges, L is an n-by-n matrix, defined as follows:

$$L = \underset{n \times m}{\begin{pmatrix} & B^T & \end{pmatrix}} \cdot \underset{m \times m}{\begin{pmatrix} & W & \end{pmatrix}} \cdot \underset{m \times n}{\begin{pmatrix} & B & \end{pmatrix}}$$

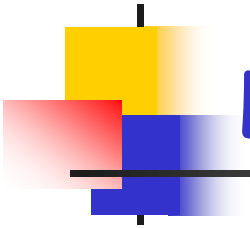
Leverage scores & effective resistances

Consider a weighted (positive weights only!) undirected graph G and let L be the Laplacian matrix of G.

Assuming n vertices and $m > n$ edges, L is an n-by-n matrix, defined as follows:



Clearly, $L = (B^T W^{1/2})(W^{1/2} B) = (B^T W^{1/2})(B^T W^{1/2})^T$.



Leverage scores & effective resistances

Effective resistances:

Let G denote an electrical network, in which each edge e corresponds to a resistor of resistance $1/w_e$.

The effective resistance R_e between two vertices is equal to the potential difference induced between the two vertices when a unit of current is injected at one vertex and extracted at the other vertex.

Leverage scores & effective resistances

Effective resistances:

Let G denote an electrical network, in which each edge e corresponds to a resistor of resistance $1/w_e$.

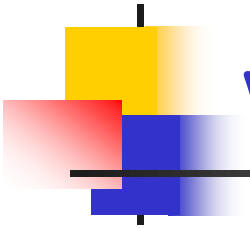
The effective resistance R_e between two vertices is equal to the potential difference induced between the two vertices when a unit of current is injected at one vertex and extracted at the other vertex.

Formally, the effective resistances are the diagonal entries of the m-by-m matrix:

$$\mathbf{R} = \mathbf{B}\mathbf{L}^{+}\mathbf{B}^{T} = \mathbf{B}(\mathbf{B}^{T}\mathbf{W}\mathbf{B})^{+}\mathbf{B}^{T}$$

Lemma: *The leverage scores of the m-by-n matrix $W^{1/2}B$ are equal (up to a simple rescaling) to the effective resistances of the edges of G.*

(Drineas & Mahoney, ArXiv 2011)



Why effective resistances?

Effective resistances are very important!

Very useful in graph sparsification (Spielman & Srivastava STOC 2008).

Graph sparsification is a critical step in solvers for Symmetric Diagonally Dominant (SDD) systems of linear equations (seminal work by Spielman and Teng).

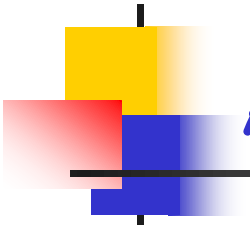
Approximating effective resistances (Spielman & Srivastava STOC 2008)

They can be approximated using the SDD solver of Spielman and Teng.

Breakthrough by Koutis, Miller, & Peng (FOCS 2010, FOCS 2011):

Low-stretch spanning trees provide a means to approximate effective resistances!

This observation (and a new, improved algorithm to approximate low-stretch spanning trees) led to almost optimal algorithms for solving SDD systems of linear equations.



Approximating leverage scores

Are leverage scores a viable alternative to approximate effective resistances?

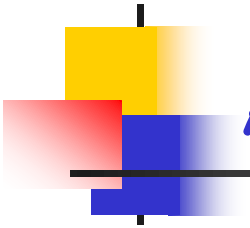
Not yet! But, we now know the following:

Theorem: Given any m-by-n matrix A with $m > n$, we can approximate its leverage scores with relative error accuracy in

$$O(mn \text{ polylog}(m)) \text{ time},$$

as opposed to the – trivial – $O(mn^2)$ time.

(Clarkson, Drineas, Mahoney, Magdon-Ismail, & Woodruff ICML 2012, ArXiv 2012)



Approximating leverage scores

Are leverage scores a viable alternative to approximate effective resistances?

Not yet! But, we now know the following:

Theorem: Given any m-by-n matrix A with $m > n$, we can approximate its leverage scores with relative error accuracy in

$$O(mn \text{ polylog}(m)) \text{ time},$$

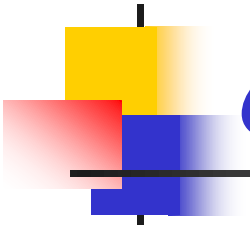
as opposed to the – trivial – $O(mn^2)$ time.

(Clarkson, Drineas, Mahoney, Magdon-Ismail, & Woodruff ICML 2012, ArXiv 2012)

Not good enough for $W^{1/2}B$!

This matrix is very sparse (2m non-zero entries). We must take advantage of the sparsity and approximate the leverage scores/effective resistances in $O(m \text{ polylog}(m))$ time.

Our algorithm will probably not do the trick, since it depends on random projections that "densify" the input matrix.



Conclusions

- **Leverage scores:** a statistic on rows/columns of matrices that reveals the most influential rows/columns of a matrix.

- **Leverage scores:** equivalent to effective resistances.

- **Additional Fact:** Leverage scores can be "uniformized" by preprocessing the matrix via random projection-type matrices.
  (E.g., random sign matrices, Gaussian matrices, or Fast JL-type transforms.)

- **Open (?) question:** how fast can we approximate the leverage scores for **sparse** matrices?