Sketching as a Tool for Numerical Linear Algebra

- David Woodruff
- IBM Almaden

Massive data sets

Examples

- Internet traffic logs
- Financial data
- etc.

Algorithms

- Want nearly linear time or less
- Usually at the cost of a randomized approximation

Regression analysis

Regression

- Statistical method to study dependencies between variables in the presence of noise.

Regression analysis

Linear Regression

- Statistical method to study linear dependencies between variables in the presence of noise.

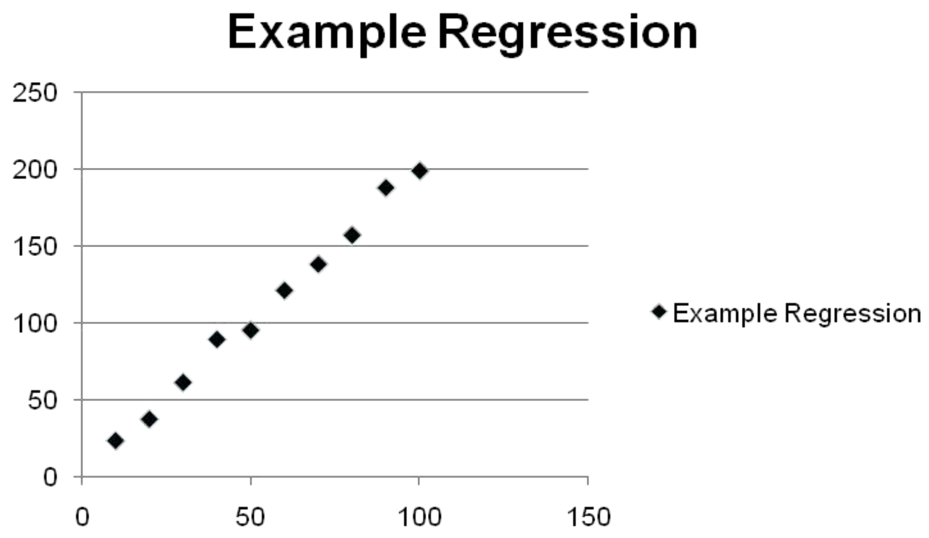
Regression analysis

Linear Regression

- Statistical method to study linear dependencies between variables in the presence of noise.

Example

- Ohm's law V = R · I



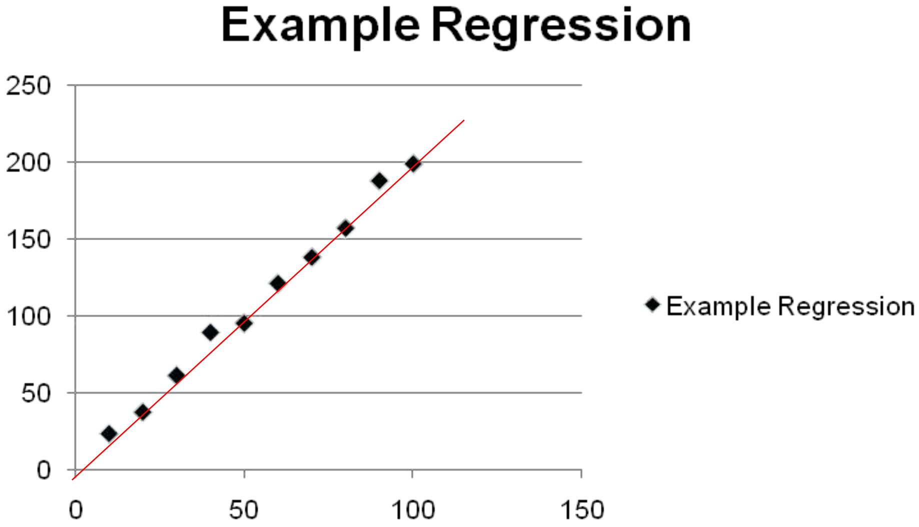
Regression analysis

Linear Regression

- Statistical method to study linear dependencies between variables in the presence of noise.

Example

- Ohm's law V = R · I
- Find linear function that best fits the data



Regression analysis

Linear Regression

- Statistical method to study linear dependencies between variables in the presence of noise.

Standard Setting

- One measured variable b
- A set of predictor variables $a_1, \ldots, a_d$
- Assumption:

$$b = x_0 + a_1 x_1 + \ldots + a_d x_d + \varepsilon$$

- ε is assumed to be noise and the x_i are model parameters we want to learn
- Can assume $x_0 = 0$
- Now consider n observations of b

Regression analysis

Matrix form

Input: n×d-matrix A and a vector $b=(b_1,\ldots, b_n)$
n is the number of observations; d is the number of predictor variables

Output: x^* so that Ax^* and b are close

Consider the over-constrained case, when n À d

- Can assume that A has full column rank

Regression analysis

Least Squares Method

- Find x^* that minimizes $|Ax-b|_2^2 = \Sigma\ (b_i - \langle A_{i*}, x\rangle)^2$
- A_{i*} is i-th row of A
- Certain desirable statistical properties

Method of least absolute deviation (l_1 -regression)

- Find x^* that minimizes $|Ax-b|_1 = \Sigma\ |b_i - \langle A_{i*}, x\rangle|$
- Cost is less sensitive to outliers than least squares

Regression analysis

Geometry of regression

- We want to find an x that minimizes $|Ax-b|_p$
- The product Ax can be written as

$$A_{*1}x_1 + A_{*2}x_2 + \ldots + A_{*d}x_d$$

where A_{*i} is the i-th column of A

- This is a linear d-dimensional subspace
- The problem is equivalent to computing the point of the column space of A nearest to b in l_p-norm

Regression analysis

Solving least squares regression via the normal equations

- How to find the solution x to $\min_x |Ax-b|_2$?
- Normal Equations: $A^TAx = A^Tb$
- $x = (A^TA)^{-1} A^T b$

Regression analysis

Solving l_1 -regression via linear programming

- Minimize (1,…,1) · ($\alpha_+ + \alpha_-$)
- Subject to:

$$A x + \alpha_+ - \alpha_- = b$$

$$\alpha_+, \alpha_- \geq 0$$

- Generic linear programming gives poly(nd) time

Talk Outline

Sketching to speed up Least Squares Regression

Sketching to speed up Least Absolute Deviation (l_1) Regression

Sketching to speed up Low Rank Approximation

Sketching to solve least squares regression

How to find an approximate solution x to $\min_x |Ax-b|_2$?

Goal: output x‘ for which $|Ax‘-b|_2 \cdot (1+\varepsilon) \min_x |Ax-b|_2$ with high probability

Draw S from a k x n random family of matrices, for a value k << n

Compute S*A and S*b

Output the solution x‘ to $\min_{x‘} |(SA)x-(Sb)|_2$

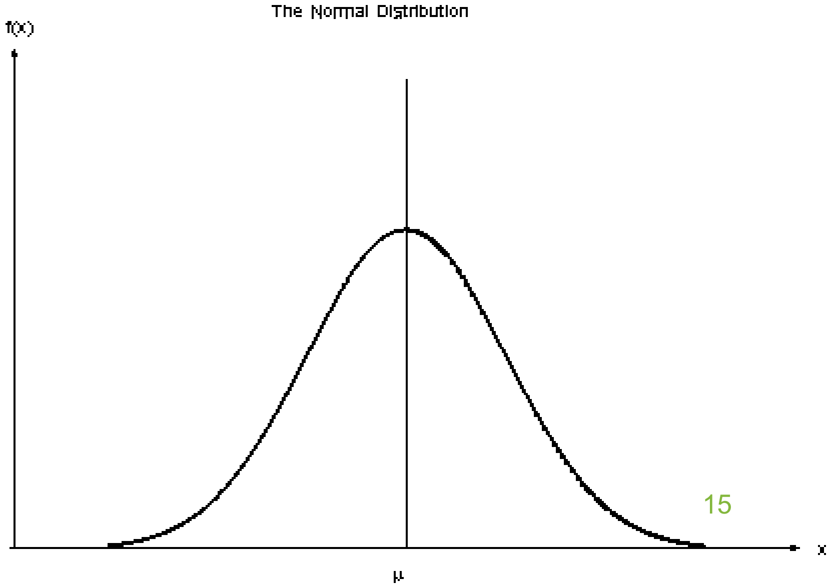
How to choose the right sketching matrix S?

Recall: output the solution x' to $\min_{x'} |(SA)x-(Sb)|_2$

Lots of matrices work

S is d/ε^2 x n matrix of i.i.d. Normal random variables

Computing S*A may be slow…



How to choose the right sketching matrix S? [S]

S is a Johnson Lindenstrauss Transform

S = P*H*D

D is a diagonal matrix with +1, -1 on diagonals

H is the Hadamard transform

P just chooses a random (small) subset of rows of H*D

S*A can be computed much faster

Even faster sketching matrices [CW,MM,NN]

CountSketch matrix

Define k x n matrix S, for k = d^2/ε^2

S is really sparse: single randomly chosen non-zero entry per column

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Surprisingly,

this works!

Talk Outline

Sketching to speed up Least Squares Regression

Sketching to speed up Least Absolute Deviation (l_1) Regression

Sketching to speed up Low Rank Approximation

Sketching to solve l_1-regression

How to find an approximate solution x to $\min_x |Ax-b|_1$?

Goal: output x‘ for which $|Ax‘-b|_1 \cdot (1+\varepsilon) \min_x |Ax-b|_1$ with high probability

Natural attempt: Draw S from a k x n random family of matrices, for a value k << n

Compute S*A and S*b

Output the solution x‘ to $\min_{x‘} |(SA)x-(Sb)|_1$

Turns out this does not work!

Sketching to solve l_1-regression [SW]

Why doesn’t outputting the solution x‘ to $\min_{x‘}$ $|(SA)x-(Sb)|_1$ work?

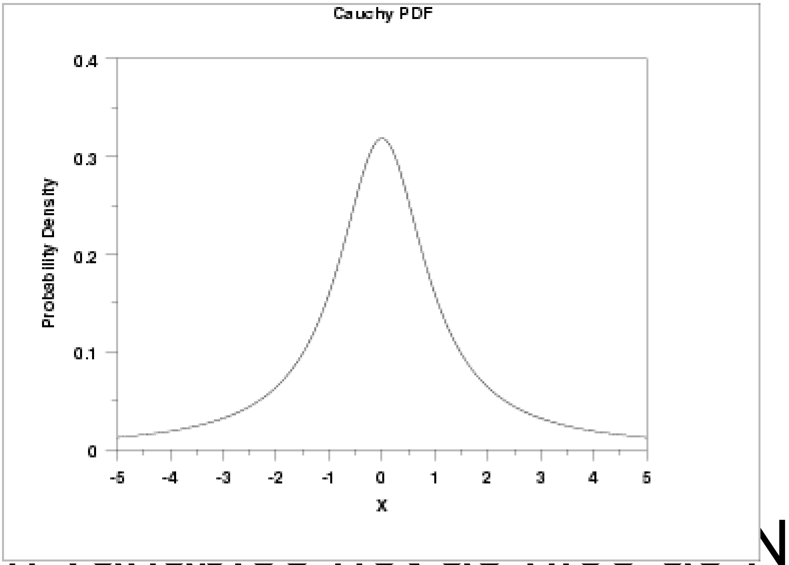
Don‘t know of k x n matrices S with small k for which if x‘ is solution to $\min_x |(SA)x-(Sb)|_1$ then
$|Ax‘-b|_1 \cdot (1+\varepsilon) \min_x |Ax-b|_1$
with high probability

Instead: can find an S so that

$$|Ax‘-b|_1 \cdot (d \log d) \min_x |Ax-b|_1$$

S is a matrix of i.i.d. Cauchy random variables

Cauchy random variables



Cauchy random variables not as nice as Normal (Gaussian) random variables

They don’t have a mean and have infinite variance

Ratio of two independent Normal random variables is Cauchy

Sketching to solve l_1-regression

How to find an approximate solution x to $\min_x |Ax-b|_1$?

Want x‘ for which if x‘ is solution to $\min_x |(SA)x-(Sb)|_1$, then $|Ax‘-b|_1 \cdot (1+\varepsilon) \min_x |Ax-b|_1$ with high probability

For d log d x n matrix S of Cauchy random variables:

$$|Ax‘-b|_1 \cdot (d \log d) \min_x |Ax-b|_1$$

For this “poor” solution x’, let b’ = Ax’-b

Might as well solve regression problem with A and b’

Sketching to solve l_1-regression

Main Idea: Compute a QR-factorization of S*A

Q has orthonormal columns and Q*R = S*A

A*R^{-1} turns out to be a “well-conditioning” of original matrix A

Compute A*R^{-1} and sample $d^{3.5}/\varepsilon^2$ rows of [A*R^{-1} , b’] where the i-th row is sampled proportional to its 1-norm

Solve regression problem on the (reweighted) samples

Sketching to solve l_1-regression [MM]

Most expensive operation is computing S*A where S is the matrix of i.i.d. Cauchy random variables

All other operations are in the "smaller space"

Can speed this up by choosing S as follows:

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{¢} \begin{bmatrix} C_1 & & & & \\ & C_2 & & & \\ & & C_3 & & \\ & & & \cdots & \\ & & & & C_n \end{bmatrix}$$

Further sketching improvements [WZ]

Can show you need a fewer number of sampled rows in later steps if instead choose S as follows

Instead of diagonal of Cauchy random variables, choose diagonal of reciprocals of exponential random variables

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{¢} \begin{bmatrix} 1/E_1 & & & & \\ & 1/E_2 & & & \\ & & 1/E_3 & & \\ & & & \cdots & \\ & & & & 1/E_n \end{bmatrix}$$

Talk Outline

Sketching to speed up Least Squares Regression

Sketching to speed up Least Absolute Deviation (l_1) Regression

Sketching to speed up Low Rank Approximation

Low rank approximation

A is an n x n matrix

Typically well-approximated by low rank matrix

E.g., only high rank because of noise

Want to output a rank k matrix A', so that

$$|A-A'|_F \cdot (1+\varepsilon) |A-A_k|_F,$$

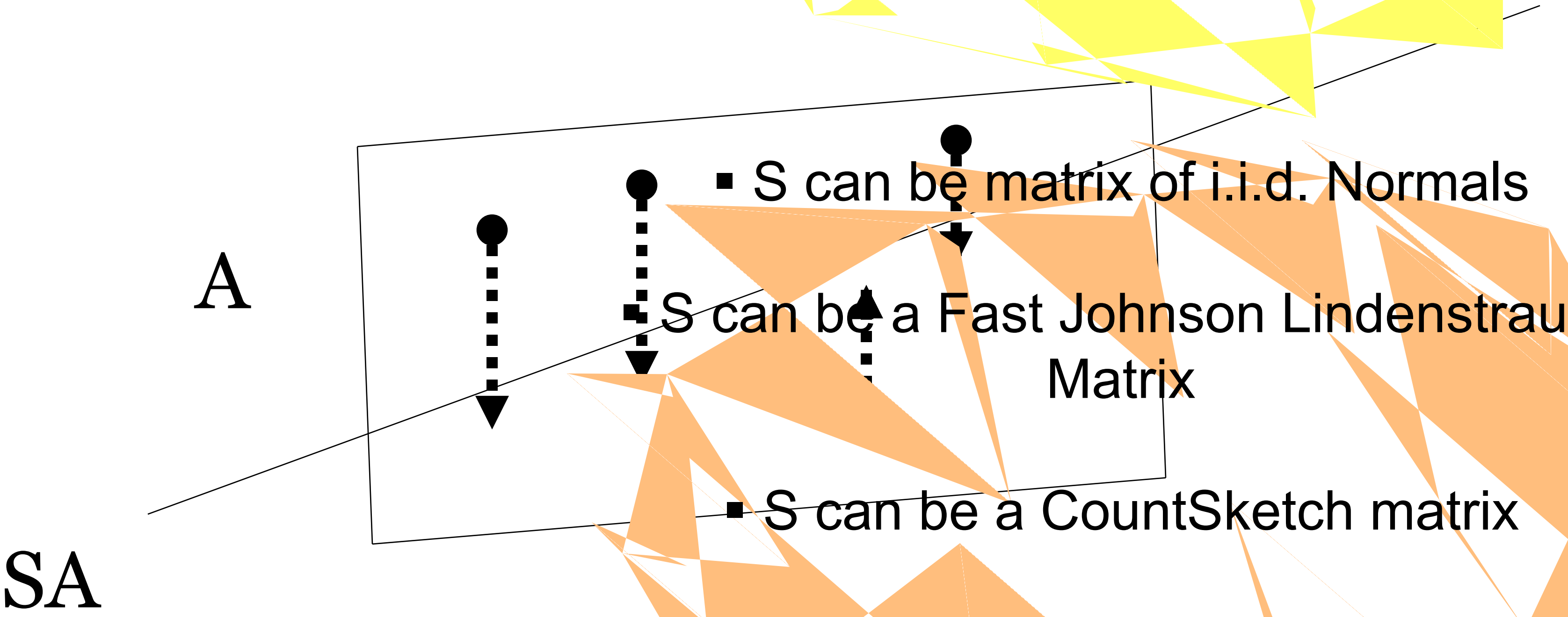
w.h.p., where $A_k = \text{argmin}_{\text{rank k matrices B}} |A-B|_F$

For matrix C, $|C|_F = (\Sigma_{i,j} C_{i,j}^2)^{1/2}$

Solution to low-rank approximation [S]

- Given n x n input matrix A
- Compute S*A using a sketching matrix S with k << n rows. S*A takes random linear combinations of rows of A

Most time-consuming step is computing S*A



- S can be matrix of i.i.d. Normals
- S can be a Fast Johnson Lindenstrauss Matrix
- S can be a CountSketch matrix

Project rows of A onto SA, then find best rank-k approximation to points inside of SA.

Caveat: projecting the points onto SA is slow [CW]

Current algorithm:

1. Compute S*A (easy)
2. Project each of the rows onto S*A
3. Find best rank-k approximation of projected points inside of rowspace of S*A (easy)

Bottleneck is step 2

Turns out if you compute $(AR)(S*A*R)^{-}(SA)$, this is a good low-rank approximation

Uses generalized regression: $\min_X |X(SA)-A|_F^2$

Conclusion

Gave fast sketching-based algorithms for numerical linear algebra problems

Least Squares Regression

Least Absolute Deviation (l_1) Regression

Low Rank Approximation

Sketching also provides “dimensionality reduction”

Communication-efficient solutions for these problems