Least-squares in RandNLA

Michael W. Mahoney[1]

Complexity of Matrix Computations Seminar
September 1, 2021

[1]ICSI and Dept of Statistics, UC Berkeley

Least-squares and solving least-squares

Consider an $n \times d$ least squares problem $(\mathbf{A}, \mathbf{b})$, where $n \gg d$:

$$L(\mathbf{w}) = \sum_{i=1}^{n} (\mathbf{a}_i^\top \mathbf{w} - b_i)^2 = \|\mathbf{A}\mathbf{w} - \mathbf{b}\|^2$$

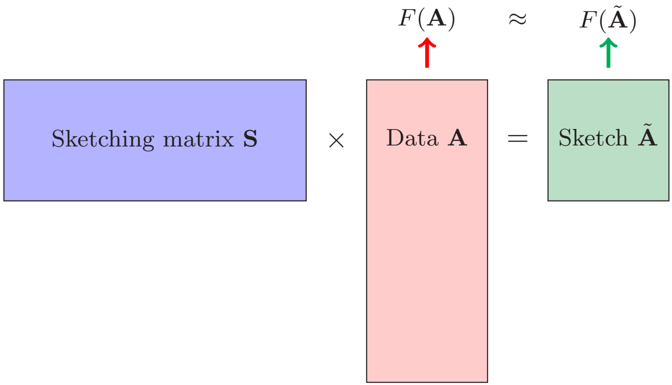
Goal: Find (exactly or approximately) the optimum solution:

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} L(\mathbf{w}) = \mathbf{A}^\dagger \mathbf{b}.$$

Many ways to solve this:

- Direct methods: normal equations; via QR; via the SVD
- Iterative methods: LSQR, Chebyshev semi-iterative, etc.
- Randomized Numerical Linear Algebra sketching methods

RandNLA Sketching



- Q1: How to construct the sketch?
- Q2: How to use the sketch to solve the problem?

Q1: How to construct the sketch?

Data oblivious methods.

- Random orthogonal matrix
- Entries i.i.d. Gaussian (*)
- Hadamard/Fourier-like construction (*)
- Entries i.i.d. Rademacher / sub-Gaussian (*)
- Sparse CountSketch and extensions (*)

Data aware methods.

- Approximate leverage scores (*)

  i-th leverage score $= (\mathbf{P})_{ii}$, where $\mathbf{P} = \mathrm{proj}(\mathrm{span}(\mathbf{A}))$

  "condition number" for sampling algorithms

- Leverage-like sketches

Data oblivious + data aware methods.

- Sparse LESS embeddings (*)

Q2: How to use the sketch to solve the problem?

- **Sketch-and-solve.**
  - Get a sketch; solve subproblem; return answer
- **Sketch-and-precondition.**
  - Get a sketch; construct a preconditioner; call traditional iterative algorithm
- **Sketch-and-regularize.**
  - Get a sketch; solve regularized subproblem; process and return answer

Sketch-and-solve.

- Sketch with any of the sketching methods.
  - Slightly "oversample" with any data-oblivious projection or data-aware leverage sampling method.
- Solve the LS problem on the sketched problem.
  - With any black box solver.
- Return the solution.
  - Get "relative error" on objective and solution.

Setup

- Original LS problem:

$$
\begin{aligned}
\mathbf{x}_{opt} &= \text{argmin}_{\mathbf{x}} ||\mathbf{A}\mathbf{x} - \mathbf{b}||_2 && (1) \\
&= (\mathbf{A})^{\dagger}\mathbf{b} && (2)
\end{aligned}
$$

  Thus, $\hat{\mathbf{b}} = \mathbf{P}\mathbf{b}$, where $\mathbf{P} = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$.

- Sketched LS problem:

$$
\begin{aligned}
\tilde{\mathbf{x}}_{opt} &= \text{argmin}_{\mathbf{x}} ||\mathbf{Z}(\mathbf{A}\mathbf{x} - \mathbf{b})||_2 && (3) \\
&= (\mathbf{Z}\mathbf{A})^{\dagger}\,\mathbf{Z}\mathbf{b} && (4)
\end{aligned}
$$

  I.e., premultiply $\mathbf{A}$ and $\mathbf{b}$ with some arbitrary matrix $\mathbf{Z}$ (e.g., random sketch $\mathbf{S}$, left singular vectors $\mathbf{U}_{\mathbf{A}}$, etc.).

Basic Structural Theorem

Theorem

If $\mathbf{Z}$ *satisfies certain structural conditions, then*

$$
\begin{aligned}
||\mathbf{b} - \mathbf{A}\tilde{\mathbf{x}}_{opt}||_2 &\leq (1+\epsilon)||\mathbf{b} - \mathbf{A}\mathbf{x}_{opt}||_2 \quad \textit{and} \quad (5) \\
||\mathbf{x}_{opt} - \tilde{\mathbf{x}}_{opt}||_2 &\leq \sqrt{\epsilon}\left(\kappa(\mathbf{A})\sqrt{\gamma^{-2}-1}\right)||\mathbf{x}_{opt}||_2, \quad (6)
\end{aligned}
$$

where $\kappa(\mathbf{A})$ *is the condition number and* $\gamma = ||\mathbf{Pb}||_2/||\mathbf{b}||_2$.

For this result, $\mathbf{Z}$ can be deterministic or randomized.

All the sketching methods construct $\mathbf{S}$ to satisfy these structural conditions, if you set parameters right.[2]

[2] All the papers you read focus on the details of this.

Subspace embeddings

Structural (subspace embedding) conditions.

$$\left\|\mathbf{I} - (\mathbf{Z}\mathbf{U_A})^T(\mathbf{Z}\mathbf{U_A})\right\|_2^2 \leq 1/2 \tag{7}$$

$$||(\mathbf{Z}\mathbf{U_A})^T\mathbf{Z}\mathbf{b}^\perp - \mathbf{U_A}\mathbf{b}^\perp||_2^2 \leq \frac{\epsilon}{2}||\mathbf{A}\mathbf{x}_{opt} - \mathbf{b}||_2^2 \tag{8}$$

- First used by [DMM06] with sampling;
  then used with projections by [Sar06, DMMS11];
  then popularized and extended by [Woo14].
- Acute perturbations.
- "Johnson-Lindenstrauss in a Euclidean space."
- "Morally necessary and sufficient" for worst-case analysis.[3]
- Neither necessary nor sufficient for NLA, statistics, etc.

[3] but see [CI18, CI20]

Sketch-and-precondition.

- Sketch with any of the sketching methods.[4]
- Use the sketch to construct a preconditioner.
- Call a traditional iterative algorithm.

[4] Sketch in the same way as with Sketch-and-solve.

Comments on Sketch-and-precondition

Idea: If the sketch is reasonable, it can be used to construct a preconditioner; e.g., do QR of $\mathbf{SA}$ rather than $\mathbf{A}$

- Introduced by [RT08];
  Blendenpik beat LAPACK [AMT10];
  LSRN in parallel/distributed [MSM14]
- Subspace embedding is overkill: a low-rank perturbation of a good preconditioner is still a good preconditioner
- Convergence guarantees follow from spectral control and the iterative method (better w.r.t. error parameter ϵ)

Statistical issues

Conditional expectation/variance for Sketch-and-solve:

$$\mathbf{E}_{\text{data}}\left[\tilde{\mathbf{w}}_{\mathcal{S}}|\mathbf{b}\right] = \hat{\mathbf{w}}_{ols} + \mathbf{E}_{\text{data}}\left[R_{\mathcal{S}}\right];$$

$$\mathbf{Var}_{\text{data}}\left[\tilde{\mathbf{w}}_{\mathcal{S}}|\mathbf{b}\right] = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\left[Diag\left\{\hat{\mathbf{e}}\right\} Diag\left\{\frac{1}{r\boldsymbol{\pi}}\right\} Diag\left\{\hat{\mathbf{e}}\right\}\right]\mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1} + \mathbf{Var}_{\text{data}}\left[R_{\mathcal{S}}\right],$$

where $\hat{\boldsymbol{e}} = \mathbf{b} - \mathbf{A}\hat{\mathbf{w}}_{ols}$, R_W is the remainder, and $\mathcal{S}$ specifies the sampling probability distribution.

- MSE, AMSE, EAMSE [MMY15, MZX+20]
- Must control small (not large) leverage scores
- Other notions of optimal sampling
- Random projections tends to uniformize all of these
- OK to not be a subspace embedding—that just introduces some bias.

Inversion Bias

Task: Estimate $F\big((\mathbf{A}^\top\mathbf{A})^{-1}\big)$, where F is a linear function

- $(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{b}$ — least squares, second-order optimization
- $\mathbf{x}^\top(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{x}$ — statistical leverage scores
- $\mathrm{tr}\,\mathbf{C}(\mathbf{A}^\top\mathbf{A})^{-1}$ — uncertainty quantification, optimal design

Inversion bias: $\mathbb{E}\big[(\tilde{\mathbf{A}}^\top\tilde{\mathbf{A}})^{-1}\big] \neq (\mathbf{A}^\top\mathbf{A})^{-1}$

Correcting the inversion bias

Simple correction for a Gaussian sketch $\tilde{\mathbf{A}}$ of size $m \times d$:

$$\mathbb{E}\big[(\gamma \tilde{\mathbf{A}}^\top \tilde{\mathbf{A}})^{-1}\big] = (\mathbf{A}^\top \mathbf{A})^{-1} \quad \text{for} \quad \gamma = \frac{m}{m-d-1}$$

Dense Gaussian sketch:

- unbiased Newton step;
- strong problem-independent convergence;
- etc.

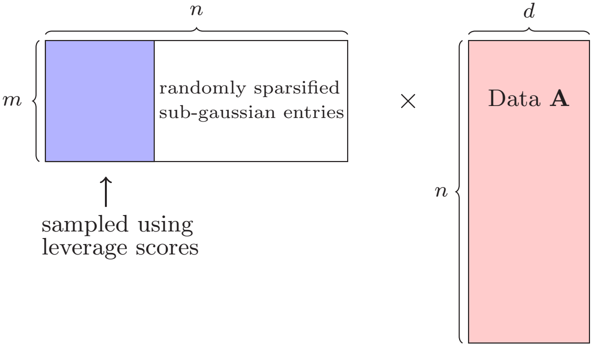
- *Not true for other sketching methods!*

- *What if we slightly relax the notion of unbiasedness?*

LESS Embeddings: Fast Gaussian-like Sketches

Leverage Score Sparsified (LESS) Embeddings [DLDM20]:

Leverage Score Sampling + *Sparse Embedding Matrices*



- Easy to make sub-Gaussian embeddings (ϵ, δ)-unbiased.
- LESS makes very sparse embeddings (ϵ, δ)-unbiased.

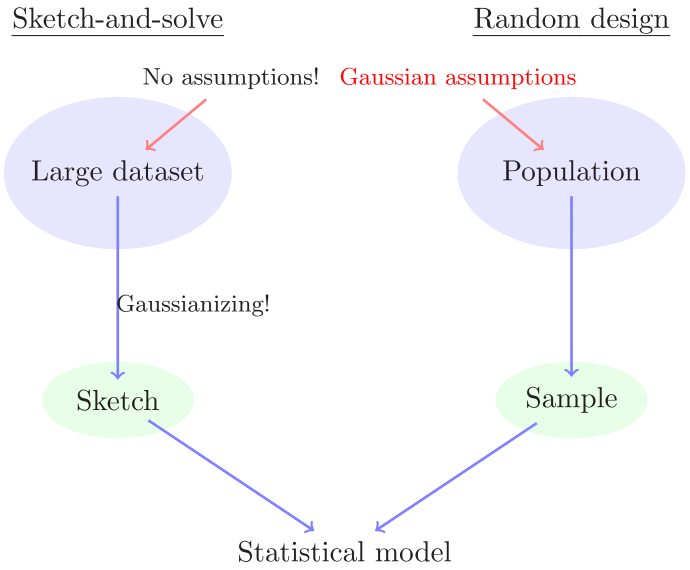
Newton-LESS: Sparsification without Trade-offs

Second order optimization with LESS sketches [DLPM21]

- Random sketching: trade-off between cost of sketching and convergence rate
- LESS sketches: dramatically sparsify without affecting convergence (versus dense Gaussian)
- Corollary: SOTA convergence for iterative LS solver[5]

[5] in theory; but that's what other sketching methods were 15 years ago ...

Randomized Sketching as a Computational Model for Statistical Inference



Sketch-and-regularize.

- Sketch with any of the sketching methods.
- Solve the regularized LS problem on the sketched problem.

$$
\begin{aligned}
\widehat{\mathbf{w}}_\lambda &= \operatorname*{argmin}_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \lambda\|\mathbf{w}\|^2 \\
&= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y} \\
&= (\mathbf{A}^\top\mathbf{S}^\top\mathbf{S}\mathbf{A} + \lambda\mathbf{I})^{-1}\mathbf{A}^\top\mathbf{S}^\top\mathbf{S}\mathbf{b}.
\end{aligned}
$$

- Return the solution.
  - Based on statistical intuition, we expect that

$$
L(\widehat{\mathbf{w}}_\lambda) < L(\widehat{\mathbf{w}}) \quad \text{for some} \quad \lambda > 0,
$$

  where $L(\mathbf{w}) = \|\mathbf{A}\mathbf{w} - \mathbf{b}\|^2$ is the unregularized objective.

Stay tuned ...

Low-rank matrix approximation: structural result

Lemma

Let $\mathbf{V}_k \in \mathbb{R}^{n\times k}$ be the top k right singular vectors of $\mathbf{A} \in \mathbb{R}^{m\times n}$. Let $\mathbf{Z} \in \mathbb{R}^{n\times r}$ $(r \geq k)$ be any matrix such that $\mathbf{V}_k^T\mathbf{Z}$ has full rank. Then, for any unitarily invariant norm ξ,

$$\|\mathbf{A} - P_{\mathbf{AZ}}\mathbf{A}\|_\xi \leq \|\mathbf{A} - \mathbf{A}_k\|_\xi + \left\|\Sigma_{k,\perp}\left(\mathbf{V}_{k,\perp}^T\mathbf{Z}\right)\left(\mathbf{V}_k^T\mathbf{Z}\right)^\dagger\right\|_\xi .$$

- Used by [BMD09] for Column Subset Selection.
- Used by [HMT11] for low-rank approximation.
- See [MD16] for discussion.
- As with LS, various extensions and refinements.

RandNLA Reviews

- [Mah11]: general overview, ML perspective.
- [HMT11]: framewok for low-rank approximation.
- [Woo14]: sketching, especially TCS perspective.
- [DM16]: ACM review/overview.
- [DM18]: PCMI lecture notes chapter.
- [KV17]:
- [MT20]:
- [DM21]:
- ...

References I

H. Avron, P. Maymounkov, and S. Toledo.
Blendenpik: Supercharging LAPACK's least-squares solver.
SIAM Journal on Scientific Computing, 32:1217–1236, 2010.

C. Boutsidis, M. W. Mahoney, and P. Drineas.
An improved approximation algorithm for the column subset selection problem.
In Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms, pages 968–977, 2009.

J. T. Chi and I. C. F. Ipsen.
A projector-based approach to quantifying total and excess uncertainties for sketched linear regression.
Technical Report Preprint: arXiv:1808.05924, 2018.

J. T. Chi and I. C. F. Ipsen.
Multiplicative perturbation bounds for multivariate multiple linear regression in schatten p-norms.
Technical Report Preprint: arXiv:2007.06099, 2020.

M. Dereziński, Z. Liao, E. Dobriban, and M. W. Mahoney.
Sparse sketches with small inversion bias.
Technical Report Preprint: arXiv:2011.10695, 2020.

M. Dereziński, J. Lacotte, M. Pilanci, and M. W. Mahoney.
Newton-LESS: Sparsification without trade-offs for the sketched Newton update,.
Technical Report Preprint: arXiv:2107.07480, 2021.

References II

P. Drineas and M. W. Mahoney.
RandNLA: Randomized numerical linear algebra.
Communications of the ACM, 59:80–90, 2016.

P. Drineas and M. W. Mahoney.
Lectures on randomized numerical linear algebra.
In M. W. Mahoney, J. C. Duchi, and A. C. Gilbert, editors, The Mathematics of Data, IAS/Park City Mathematics Series, pages 1–48. AMS/IAS/SIAM, 2018.

M. Derezinski and M. W. Mahoney.
Determinantal point processes in randomized numerical linear algebra.
Notices of the AMS, 68(1):34–45, 2021.

P. Drineas, M. W. Mahoney, and S. Muthukrishnan.
Sampling algorithms for ℓ_2 regression and applications.
In Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms, pages 1127–1136, 2006.

P. Drineas, M. W. Mahoney, S. Muthukrishnan, and T. Sarlós.
Faster least squares approximation.
Numerische Mathematik, 117(2):219–249, 2011.

References III

N. Halko, P.-G. Martinsson, and J. A. Tropp.
Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions.
SIAM Review, 53(2):217–288, 2011.

R. Kannan and S. Vempala.
Randomized algorithms in numerical linear algebra.
Acta Mathematica, 26:95–135, 2017.

M. W. Mahoney.
Randomized algorithms for matrices and data.
Foundations and Trends in Machine Learning. NOW Publishers, Boston, 2011.

M. W. Mahoney and P. Drineas.
Structural properties underlying high-quality randomized numerical linear algebra algorithms.
In P. Bühlmann, P. Drineas, M. Kane, and M. van de Laan, editors, Handbook of Big Data, pages 137–154. CRC Press, 2016.

P. Ma, M. W. Mahoney, and B. Yu.
A statistical perspective on algorithmic leveraging.
Journal of Machine Learning Research, 16:861–911, 2015.

References IV

X. Meng, M. A. Saunders, and M. W. Mahoney.
LSRN: A parallel iterative solver for strongly over- or under-determined systems.
SIAM Journal on Scientific Computing, 36(2):C95–C118, 2014.

P.-G. Martinsson and J. A. Tropp.
Randomized numerical linear algebra: Foundations and algorithms.
Acta Numerica, 29:403–572, 2020.

P. Ma, X. Zhang, X. Xing, J. Ma, and M. W. Mahoney.
Asymptotic analysis of sampling estimators for randomized numerical linear algebra algorithms.
Technical Report Preprint: arXiv:2002.10526, 2020.

V. Rokhlin and M. Tygert.
A fast randomized algorithm for overdetermined linear least-squares regression.
Proc. Natl. Acad. Sci. USA, 105(36):13212–13217, 2008.

T. Sarlós.
Improved approximation algorithms for large matrices via random projections.
In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, pages 143–152, 2006.

References V

D. P. Woodruff.
Sketching as a Tool for Numerical Linear Algebra.
Foundations and Trends in Theoretical Computer Science. NOW Publishers, Boston, 2014.