• DISCLAIMER: These notes are not necessarily an accurate representation of what I said during the class. They are mostly what I intend to say, and have not been carefully edited.

• Summary
  – Sampling matrices via leverage scores.
  – Uniform sampling.
  – Coherence reducing reweighting.

• Leverage score sampling.
  – Given \( n \times d \) matrix \( A \), find \( \tilde{A} \) so that \( \|Ax\|_2 \approx \|\tilde{A}x\|_2 \) for all \( x \).
  – Matrix concentration (Lectures 7 and 11):
    * Let rows of \( A \) be \( a_1^T \ldots a_n^T \).
    * Leverage score: \( \tau_i = a_i^T(A^TA)^{-1}a_i \).
    * Matrix concentration sampling rows w.p. \( \min\{1, \tau_i \log d\} \) gives \( \tilde{A} \) with \( O(d \log d) \) rows s.t. \( A \approx_{O(1)} \tilde{A} \).
  – Application: overconstrained regression: \( \min_x \|Ax - b\|_2 \).

• Computing leverage scores:
  – Lecture 9: solve \( O(\log n) \) linear systems in \( A^TA \).
  – \( A^TA \): \( d \times d \) matrix, but need time \( nd^2 \) to compute.
  – Need a good approximation, \( \tilde{A} \approx A \).
  – Chicken-and-egg problem.

• Solutions:
  – Matrix sketches (Lecture 12): \( \tilde{A} = GA \), can mix up structure of data.
  – (tail) recursive scheme: sample half of \( A \) to obtain \( \hat{A} \), compute leverage scores w.r.t. (a good approximation of) \( \hat{A} \).
  – Performance: \( O(\log(n/d)) \) steps.
Key property of leverage score sampling: can correct for a factor 2 error in leverage score estimation by taking twice as many samples.

- Generalized leverage scores:
  - Issue: \( \hat{A} \) can be lower rank, e.g. \( A = I \).
  - Alternate definition: add \( a_i \) to the space as well.

\[
\tau_{\hat{A}}(a_i) = a_i^T \left( \hat{A}^T \hat{A} + a_i^T a_i \right)^\dagger a_i.
\]

- Computable by doing ‘is in null space’ check.

- Main result:
  - If we sample \( \alpha \) of the rows, \( E_{\hat{A}} \left[ \sum_i \tau_{\hat{A}}(a_i) \right] = O(\alpha^{-1}d) \).
  - Use: build a sequence \( A^{(0)}, A^{(1)}, A^{(2)} \ldots \) each a random half of the previous.
  - Then build backwards a sequence of approximations, \( \tilde{A}^{(i)} \approx A^{(i)} \).
  - Structural theorem gives that leverage scores of \( A^{(i)} \) lead to \( O(d \log d) \) sized approximation to \( A^{(i-1)} \).

- Proof of main result:
  - Current view: pick \( i \), then pick \( \hat{A} \).
  - Equivalent process: pick \( \hat{A} \cup \{ i \} \), then pick \( i \) from it.
  - \( \hat{A} \) has \( \alpha n + 1 \) rows, total leverage score: \( d \).
  - Conditioned on a fixed \( \hat{A} \), Expected leverage score of a random row: \( \frac{d}{\alpha n + 1} \).
  - Total across all rows: \( d \frac{n}{\alpha n + 1} \leq \alpha^{-1}d \).

- Coherence reducing reweighting:
  - Maximum leverage score also referred to as coherence.
  - Can rescale \( O(\alpha^{-1}d) \) rows so that all leverage scores are \( \leq \alpha \).
  - Main idea: whack-a-mole: decrease weights of any row whose leverage score is more than \( \alpha \).
  - Let these rows be \( \overline{A} \).
  - Probabilities of \( \alpha \) dominates leverage scores of \( A + \alpha^{-1} \overline{A} \).
  - Sample results in \( \hat{A} \) s.t. \( A^T A \preceq \hat{A}^T \hat{A} \preceq A^T A^T + \alpha^{-1} \hat{A}^T \overline{A} \).