EE270
Large scale matrix computation, optimization and learning

Instructor : Mert Pilanci
Stanford University

Thursday, Jan 23 2020
Randomized Linear Algebra
Lecture 7: Least Squares Optimization and Random Projections
Recap: Johnson Lindenstrauss Lemma

Let $\epsilon \in (0, \frac{1}{2})$. Given any set of points $\{x_1, \ldots, x_n\}$ in $\mathbb{R}^d$, there exists a map $S : \mathbb{R}^n \to \mathbb{R}^m$ with $m = \frac{9 \log(n)}{\epsilon^2 - \epsilon^3}$ such that

$$1 - \epsilon \leq \frac{\|Sx_i - Sx_j\|_2^2}{\|x_i - x_j\|_2^2} \leq 1 + \epsilon$$
Recap: Johnson Lindenstrauss Lemma

Let $\epsilon \in (0, \frac{1}{2})$. Given any set of points $\{x_1, \ldots, x_n\}$ in $\mathbb{R}^d$, there exists a map $S : \mathbb{R}^n \rightarrow \mathbb{R}^m$ with $m = \frac{9 \log(n)}{\epsilon^2 - \epsilon^3}$ such that

$$1 - \epsilon \leq \frac{\|Sx_i - Sx_j\|_2^2}{\|x_i - x_j\|_2^2} \leq 1 + \epsilon$$

Note that the target dimension $m$ is independent of the original dimension $d$, and depends only on the number of points $n$ and the accuracy parameter.
Recap: Johnson Lindenstrauss Lemma

- Let $\epsilon \in (0, \frac{1}{2})$. Given any set of points $\{x_1, ..., x_n\}$ in $\mathbb{R}^d$, there exists a map $S : \mathbb{R}^n \rightarrow \mathbb{R}^m$ with $m = \frac{9 \log(n)}{\epsilon^2 - \epsilon^3}$ such that

$$1 - \epsilon \leq \frac{\|Sx_i - Sx_j\|_2^2}{\|x_i - x_j\|_2^2} \leq 1 + \epsilon$$

- Note that the target dimension $m$ is independent of the original dimension $d$, and depends only on the number of points $n$ and the accuracy parameter.

- more surprises: picking an $m \times d$ random matrix $S = \frac{1}{\sqrt{m}} G$ with $G_{ij} \sim N(0, 1)$ standard normal works with high probability!
True ‘projections’: random subspaces also work

- Pick $S_{(i)}$ uniformly random on the unit sphere
- Pick $S_{(i+1)}$ uniformly random on the unit sphere and $\perp S_{(i)}, \ldots, S_{(1)}$
- $S$ is a projection matrix, which projects onto a uniformly random subspace

\[ \mathbb{P} \left\{ \left| \|Su\|_2 - \sqrt{\frac{m}{d}} \right| > t \right\} \leq 2e^{-\frac{t^2d}{2}} \]

- Applying union bound for all points $i, j = 1, \ldots, d$ gives a similar result
- Random i.i.d. $S$ matrices are easier to generate and approximately orthogonal: $\mathbb{E}S^T S = I$
Computationally cheaper random matrices

- **Gaussian** \( S_{ij} = \frac{1}{\sqrt{m}} N(0, 1) \)

- **Rademacher**

  \[
  S_{ij} = \begin{cases} 
  + \frac{1}{m} & \text{with probability } \frac{1}{\sqrt{m}} \\ 
  - \frac{1}{\sqrt{m}} & \text{with probability } \frac{1}{2} 
  \end{cases}
  \tag{1}
  \]

- **Bernoulli-Rademacher**

  \[
  S_{ij} = \begin{cases} 
  + \frac{\sqrt{3}}{\sqrt{m}} & \text{with probability } \frac{1}{2} \\ 
  0 & \text{with probability } \frac{2}{3} \\ 
  - \frac{\sqrt{3}}{\sqrt{m}} & \text{with probability } \frac{1}{2} 
  \end{cases}
  \tag{2}
  \]

- **other sparse matrices** (e.g. one non-zero per column)
- **Fourier transform based matrices**
Random projection for Approximate Matrix Multiplication

- Let the approximate product of $AB$ be $C = AS^T SB$

\[ \mathbb{P}[\|AB - C\|_F > 3\epsilon\|A\|_F\|B\|_F] \leq \delta \]

- Follows from JL Moment property
- \( S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \) \times \text{random i.i.d. sub-Gaussian, e.g., } \pm 1, \text{ or } \mathcal{N}(0, 1) \text{ with } m = \frac{c_1}{\epsilon^2} \log \frac{1}{\delta} \)
- \( S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \) \times \text{CountSketch matrix (one nonzero per column, which is } \pm 1 \text{ at a uniformly random location) with } m = \frac{c_2}{\epsilon^2 \delta} \)
- \( S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \) \times \text{Fast JL Transform with } m = \frac{c_3}{\epsilon} \log \frac{1}{\delta} \)

Sparse JL and Fast JL are more efficient
- advantages: doesn't require any knowledge about matrices \( A \) and \( B \) (oblivious)
- optimal sampling probabilities depend on the column/row norms of \( A \) and \( B \)
Random projection for Approximate Matrix Multiplication

- Let the approximate product of $AB$ be $C = A S^T S B$

  \[ \mathbb{P} [ \|AB - C\|_F > 3\epsilon \|A\|_F \|B\|_F ] \leq \delta \]

- Follows from JL Moment property

- $S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \times \text{random i.i.d. sub-Gaussian, e.g., } \pm 1, \text{ or } N(0, 1) \text{ with } m = \frac{c_1}{\epsilon^2} \log \frac{1}{\delta}$

- $S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \times \text{CountSketch matrix (one nonzero per column, which is } \pm 1 \text{ at a uniformly random location) with } m = \frac{c_2}{\epsilon^2 \delta}$

- $S \in \mathbb{R}^{m \times n} \sim \frac{1}{\sqrt{m}} \times \text{Fast JL Transform with } m = \frac{c_3}{\epsilon} \log \frac{1}{\delta}$

- Sparse JL and Fast JL are more efficient

- advantages: doesn’t require any knowledge about matrices $A$ and $B$ (oblivious)

- optimal sampling probabilities depend on the column/row norms of $A$ and $B$
Least Squares Regression

- Predict the value of a continuous target variable $y$ 
  $(a_1, b_1), \ldots, (a_n, b_n)$ 
  $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$

- Linear regression $f(a) = x^T a + x_0$
Least Squares Regression

- Predict the value of a continuous target variable $y$
  $(a_1, b_1), \ldots, (a_n, b_n)$
  $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$
- Linear regression $f(a) = x^T a + x_0$
- Performance measure: minimum sum of squares

$$\min_{x, x_0} \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2$$
Least Squares Regression

- Predict the value of a continuous target variable $y$ 
  $(a_1, b_1), \ldots, (a_n, b_n)$
  $a_i \in \mathbb{R}^d$ and $b_i \in \mathbb{R}$

- Linear regression $f(a) = x^T a + x_0$

- Performance measure: minimum sum of squares

  $$\min_{x, x_0} \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2$$

- We can add a regularization term $\lambda \|x\|_2^2$

  $$\min_{x, x_0} \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2 + \lambda \|x\|_2^2$$
Least Squares Regression

- Loss function:
  \[ L(x, x_0) = \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2 + \lambda \|x\|^2 \]

- \[ \frac{\partial}{\partial x_0} L(x, x_0) = \]
  optimal \[ x_0^* = \frac{1}{n} \sum_{i=1}^{n} (y_i - x^T a_i) = \bar{b} - x^T \bar{a} \]
  where \( \bar{a} = \sum_{i=1}^{n} a_i \) and \( \bar{b} = \sum_{i=1}^{n} b_i \)

- plugging \( x_0^* \) in \( L(x, x_0) \)
  \[ L(x, x_0^*) = \frac{1}{n} \sum_{i=1}^{n} (b_i - \bar{b} - x^T (a_i - \bar{a}))^2 + \lambda \|x\|^2 \]
Least Squares Regression

- Loss function:
  \[ L(x, x_0) = \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2 + \lambda \|x\|_2^2 \]

- \[ \frac{\partial}{\partial x_0} L(x, x_0) = \]
  optimal \( x_0^* = \frac{1}{n} \sum_{i=1}^{n} (y_i - x^T a_i) = \bar{b} - x^T \bar{a} \)
  where \( \bar{a} = \sum_{i=1}^{n} a_i \) and \( \bar{b} = \sum_{i=1}^{n} b_i \)

- plugging \( x_0^* \) in \( L(x, x_0) \)
  \[ L(x, x_0^*) = \frac{1}{n} \sum_{i=1}^{n} (b_i - \bar{b} - x^T (a_i - \bar{a}))^2 + \lambda \|x\|_2^2 \]
  define centered data \( \tilde{a}_i = a_i - \bar{a} \) and \( \tilde{b}_i = b_i - \bar{b} \)

  \[ \min_x \|\tilde{A}x - \tilde{b}\|_2^2 + n\lambda \|x\|_2^2 \]
Least Squares Regression

- Loss function:
  \[ L(x, x_0) = \frac{1}{n} \sum_{i=1}^{n} (b_i - x^T a_i - x_0)^2 + \lambda \|x\|^2 \]

- \[ \frac{\partial}{\partial x_0} L(x, x_0) = \]
  optimal \( x_0^* = \frac{1}{n} \sum_{i=1}^{n} (y_i - x^T a_i) = \bar{b} - x^T \bar{a} \)
  where \( \bar{a} = \sum_{i=1}^{n} a_i \) and \( \bar{b} = \sum_{i=1}^{n} b_i \)

- plugging \( x_0^* \) in \( L(x, x_0) \)
  \[ L(x, x_0^*) = \frac{1}{n} \sum_{i=1}^{n} (b_i - \bar{b} - x^T (a_i - \bar{a}))^2 + \lambda \|x\|^2 \]
  define centered data \( \tilde{a}_i = a_i - \bar{a} \) and \( \tilde{b}_i = b_i - \bar{b} \)
  \[ \min_x \| \tilde{A}x - \tilde{b} \|^2 + n\lambda \|x\|^2 \]

- \[ \frac{\partial}{\partial x} L(x, x_0^*) = 2\tilde{A}^T (\tilde{A}x^* - \tilde{b}) + 2n\lambda x^* = 0 \]
  optimal solution \( x^* = (\tilde{A}^T \tilde{A} + n\lambda I)^{-1} \tilde{A}^T \tilde{b} \)
Autoregressive Models

\[ b[n] = a[n + 1] \approx \sum_{k} x_k a[n - k] \]

- **AR(2) model**: two non-zero filter coefficients

  \[ a[n + 1] = -x_0 a[n] - x_1 a[n - 1] \]

  and error term \( e_n = 0 \)

- **Example**: Sine wave \( a[n] = \sin(\alpha n) \) satisfies AR(2) model
Autoregressive models

- We can predict future values using

\[ b[n] = \sum_{k} a[n - k] x_k \]
Given $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^d$

find the best linear fit $Ax \approx b$ according to

$$\min_{x \in \mathbb{R}^d} ||Ax - b||_2^2$$

no regularization, i.e., $\lambda = 0$

If $A$ is full column rank then

$$x_{LS} = (A^T A)^{-1} A^T b$$
Geometry

\[ \min_{x \in \mathbb{R}^d} \| Ax - b \|_2^2 \]
Singular Value Decomposition

Every $A \in \mathbb{R}^{n \times n}$ has a singular value decomposition

$$A = U\Sigma V^T$$

where $U \in \mathbb{R}^{n \times n}$ has orthonormal columns
$\Sigma$ is diagonal with non-increasing non-negative entries
$V^T$ has orthonormal rows

Pseudoinverse $A^\dagger = V \Sigma^{-1} U^T$

Least Square solution

$$x_{LS} = (A^T A)^{-1} A^T b = A^\dagger b = V \Sigma^{-1} U^T b$$
Classical Methods for Least Squares

- **Direct methods**
  - Cholesky decomposition: Form $A^T A$ and decompose $A^T A = R^T R$ where $R$ is upper triangular. Solve normal equations $(A^T A)^{-1} = (R^T R)^{-1} A^T b$
  - QR decomposition: $A = QR$, solve $Rx = Q^T b$
  - Singular Value Decomposition: $x_{LS} = V \Sigma^{-1} U^T b$
  
  Direct methods have typically $O(nd^2)$ complexity

- **Indirect methods**
  - Gradient descent with momentum (Chebyshev iteration)
  - Conjugate Gradient
  - Other iterative methods

  Indirect methods have typically $O(\sqrt{\kappa} n d)$ complexity, where $\kappa$ is the condition number
Faster Least Squares Optimization: Random Projection

- **Left-sketching**
  Form $SA$ and $Sb$ where $S \in \mathbb{R}^{m \times n}$ is a random projection matrix

- Solve the smaller problem
  \[
  \min_{x \in \mathbb{R}^d} \|SAx - Sb\|_2^2
  \]

- using any classical method.
  Direct method complexity $md^2$
Faster Least Squares Optimization: Random Projection

- **Left-sketching**
  Form $SA$ and $Sb$ where $S \in \mathbb{R}^{m \times n}$ is a random projection matrix
- Solve the smaller problem
  \[
  \min_{x \in \mathbb{R}^d} \|SAx - Sb\|_2^2
  \]
- using any classical method.
  Direct method complexity $md^2$
Approximation Result

- Let \( S \in \mathbb{R}^{m \times d} \) be a Johnson-Lindenstrauss Embedding

\[
x_{LS} = \arg \min_{x \in \mathbb{R}^d} \left\{ \left\| Ax - b \right\|_2^2 \right\} f(x)
\]

\[
\tilde{x} = \arg \min_{x \in \mathbb{R}^d} \left\| SAx - Sb \right\|_2^2
\]

- If \( m \geq \text{constant} \times \frac{\text{rank}(A)}{\epsilon^2} \) then,
  - \( f(x_{LS}) \leq f(\tilde{x}) \leq (1 + \epsilon^2) f(x_{LS}) \)
  - \( \left\| A(x_{LS} - \tilde{x}) \right\|_2^2 \leq \epsilon^2 \) with high probability
Gaussian Sketch

Let $S$ be $\frac{1}{m} \times$ i.i.d. Gaussian. $\mathbb{E}[S^T S] = I$

$$\tilde{x} = \arg\min_{x \in \mathbb{R}^d} \|SAx - Sb\|_2^2$$

Is $\mathbb{E}[\tilde{x}]$ equal to $x_{LS}$?
Questions?