Time Complexity

Nipun Batra

February 12, 2024

IIT Gandhinagar

Time Complexity: Normal Equation for Linear Regression

Normal Equation

- Consider $X \in \mathcal{R}^{N \times D}$

Normal Equation

- Consider $X \in \mathcal{R}^{N \times D}$
- N examples and D dimensions

Normal Equation

- Consider $X \in \mathcal{R}^{N \times D}$
- N examples and D dimensions
- What is the time complexity of solving the normal equation $\hat{\theta} = (X^T X)^{-1} X^T y$?

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$
- Inversion of $X^T X$ is an inversion of a $D \times D$ matrix, which is $\mathcal{O}(D^3)$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$
- Inversion of $X^T X$ is an inversion of a $D \times D$ matrix, which is $\mathcal{O}(D^3)$
- $X^T y$ is a matrix vector product of size $D \times N$ and $N \times 1$, which is $\mathcal{O}(DN)$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$
- Inversion of $X^T X$ is an inversion of a $D \times D$ matrix, which is $\mathcal{O}(D^3)$
- $X^T y$ is a matrix vector product of size $D \times N$ and $N \times 1$, which is $\mathcal{O}(DN)$
- $(X^T X)^{-1} X^T y$ is a matrix product of a $D \times D$ matrix and $D \times 1$ matrix, which is $\mathcal{O}(D^2)$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$
- Inversion of $X^T X$ is an inversion of a $D \times D$ matrix, which is $\mathcal{O}(D^3)$
- $X^T y$ is a matrix vector product of size $D \times N$ and $N \times 1$, which is $\mathcal{O}(DN)$
- $(X^T X)^{-1} X^T y$ is a matrix product of a $D \times D$ matrix and $D \times 1$ matrix, which is $\mathcal{O}(D^2)$
- Overall complexity: $\mathcal{O}(D^2 N) + \mathcal{O}(D^3) + \mathcal{O}(DN) + \mathcal{O}(D^2)$ $= \mathcal{O}(D^2 N) + \mathcal{O}(D^3)$

Normal Equation

- X has dimensions $N \times D$, X^T has dimensions $D \times N$
- $X^T X$ is a matrix product of matrices of size: $D \times N$ and $N \times D$, which is $\mathcal{O}(D^2 N)$
- Inversion of $X^T X$ is an inversion of a $D \times D$ matrix, which is $\mathcal{O}(D^3)$
- $X^T y$ is a matrix vector product of size $D \times N$ and $N \times 1$, which is $\mathcal{O}(DN)$
- $(X^T X)^{-1} X^T y$ is a matrix product of a $D \times D$ matrix and $D \times 1$ matrix, which is $\mathcal{O}(D^2)$
- Overall complexity: $\mathcal{O}(D^2 N) + \mathcal{O}(D^3) + \mathcal{O}(DN) + \mathcal{O}(D^2)$ $= \mathcal{O}(D^2 N) + \mathcal{O}(D^3)$
- Scales cubic in the number of columns/features of X

Time Complexity: Gradient Descent for Liner Regression

Gradient Descent

Start with random values of θ_0 and θ_1

Till convergence

- $\theta_0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0}(\sum \epsilon_i^2)$

Gradient Descent

Start with random values of θ_0 and θ_1

Till convergence

- $\theta_0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0}(\sum \epsilon_i^2)$
- $\theta_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1}(\sum \epsilon_i^2)$

Gradient Descent

Start with random values of θ_0 and θ_1

Till convergence

- $\theta_0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0}(\sum \epsilon_i^2)$
- $\theta_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1}(\sum \epsilon_i^2)$
- Question: Can you write the above for D dimensional data in vectorised form?

Gradient Descent

Start with random values of θ_0 and θ_1

Till convergence

- $\theta_0 = \theta_0 - \alpha \dfrac{\partial}{\partial \theta_0}(\sum \epsilon_i^2)$
- $\theta_1 = \theta_1 - \alpha \dfrac{\partial}{\partial \theta_1}(\sum \epsilon_i^2)$
- Question: Can you write the above for D dimensional data in vectorised form?
- $\theta_0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0} (y - X\theta)^\top (y - X\theta)$
  $\theta_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} (y - X\theta)^\top (y - X\theta)$
  $\vdots$
  $\theta_D = \theta_D - \alpha \frac{\partial}{\partial \theta_D} (y - X\theta)^\top (y - X\theta)$

Gradient Descent

Start with random values of θ_0 and θ_1

Till convergence

- $\theta_0 = \theta_0 - \alpha \dfrac{\partial}{\partial \theta_0}(\sum \epsilon_i^2)$
- $\theta_1 = \theta_1 - \alpha \dfrac{\partial}{\partial \theta_1}(\sum \epsilon_i^2)$
- Question: Can you write the above for D dimensional data in vectorised form?
- $$\begin{aligned} \theta_0 &= \theta_0 - \alpha \tfrac{\partial}{\partial \theta_0} (y - X\theta)^\top (y - X\theta) \\ \theta_1 &= \theta_1 - \alpha \tfrac{\partial}{\partial \theta_1} (y - X\theta)^\top (y - X\theta) \\ &\vdots \\ \theta_D &= \theta_D - \alpha \tfrac{\partial}{\partial \theta_D} (y - X\theta)^\top (y - X\theta) \end{aligned}$$
- $\theta = \theta - \alpha \frac{\partial}{\partial \theta} (y - X\theta)^\top (y - X\theta)$

Gradient Descent

- $\frac{\partial A\theta}{\partial \theta} = A^\top$
- $\frac{\partial \theta^\top A}{\partial \theta} = A$
- $\frac{\partial \theta^\top A^\top A\theta}{\partial \theta} = 2A^\top A\theta$

Gradient Descent

$$
\begin{aligned}
&\frac{\partial}{\partial\theta}(y - X\theta)^\top(y - X\theta) \\
&= \frac{\partial}{\partial\theta}\left(y^\top - \theta^\top X^\top\right)(y - X\theta) \\
&= \frac{\partial}{\partial\theta}\left(y^\top y - \theta^\top X^\top y - y^\top X\theta + \theta^\top X^\top X\theta\right) \\
&= -2X^\top y + 2X^\top X\theta \\
&= 2X^\top(X\theta - y)
\end{aligned}
$$

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\theta = \theta - \alpha X^\top X\theta + \alpha X^\top y$

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\theta = \theta - \alpha X^\top X\theta + \alpha X^\top y$

Complexity of computing $X^\top y$ is $\mathcal{O}(DN)$

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\theta = \theta - \alpha X^\top X\theta + \alpha X^\top y$

Complexity of computing $X^\top y$ is $\mathcal{O}(DN)$

Complexity of computing $\alpha X^\top y$ once we have $X^\top y$ is $\mathcal{O}(D)$ since $X^\top y$ has D entries

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\theta = \theta - \alpha X^\top X\theta + \alpha X^\top y$

Complexity of computing $X^\top y$ is $\mathcal{O}(DN)$

Complexity of computing $\alpha X^\top y$ once we have $X^\top y$ is $\mathcal{O}(D)$ since $X^\top y$ has D entries

Complexity of computing $X^\top X$ is $\mathcal{O}(D^2 N)$ and then multiplying with α is $\mathcal{O}(D^2)$

Gradient Descent

We can write the vectorised update equation as follows, for each iteration

$\theta = \theta - \alpha X^\top (X\theta - y)$

For t iterations, what is the computational complexity of our gradient descent solution?

Hint, rewrite the above as: $\theta = \theta - \alpha X^\top X\theta + \alpha X^\top y$

Complexity of computing $X^\top y$ is $\mathcal{O}(DN)$

Complexity of computing $\alpha X^\top y$ once we have $X^\top y$ is $\mathcal{O}(D)$ since $X^\top y$ has D entries

Complexity of computing $X^\top X$ is $\mathcal{O}(D^2 N)$ and then multiplying with α is $\mathcal{O}(D^2)$

All of the above need only be calculated once!

Gradient Descent

Gradient Descent

For each of the t iterations, we now need to first multiply $\alpha X^\top X$ with θ which is matrix multiplication of a $D \times D$ matrix with a $D \times 1$, which is $\mathcal{O}(D^2)$

Gradient Descent

For each of the t iterations, we now need to first multiply $\alpha X^\top X$ with θ which is matrix multiplication of a $D \times D$ matrix with a $D \times 1$, which is $\mathcal{O}(D^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(D)$ for each iteration.

Gradient Descent

For each of the t iterations, we now need to first multiply $\alpha X^\top X$ with θ which is matrix multiplication of a $D \times D$ matrix with a $D \times 1$, which is $\mathcal{O}(D^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(D)$ for each iteration.

What is overall computational complexity?

Gradient Descent

For each of the t iterations, we now need to first multiply $\alpha X^\top X$ with θ which is matrix multiplication of a $D \times D$ matrix with a $D \times 1$, which is $\mathcal{O}(D^2)$

The remaining subtraction/addition can be done in $\mathcal{O}(D)$ for each iteration.

What is overall computational complexity?

$\mathcal{O}(tD^2) + \mathcal{O}(D^2 N) = \mathcal{O}((t + N)D^2)$

Gradient Descent (Alternative)

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^\top$ is $\mathcal{O}(ND)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^{\top}(X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^{\top}$ is $\mathcal{O}(ND)$
- Computing $\alpha X^{\top}(X\theta - y)$ is $\mathcal{O}(ND)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^\top$ is $\mathcal{O}(ND)$
- Computing $\alpha X^\top (X\theta - y)$ is $\mathcal{O}(ND)$
- Computing $\theta = \theta - \alpha X^\top (X\theta - y)$ is $\mathcal{O}(D)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^\top$ is $\mathcal{O}(ND)$
- Computing $\alpha X^\top (X\theta - y)$ is $\mathcal{O}(ND)$
- Computing $\theta = \theta - \alpha X^\top (X\theta - y)$ is $\mathcal{O}(D)$

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^\top$ is $\mathcal{O}(ND)$
- Computing $\alpha X^\top (X\theta - y)$ is $\mathcal{O}(ND)$
- Computing $\theta = \theta - \alpha X^\top (X\theta - y)$ is $\mathcal{O}(D)$

What is overall computational complexity?

Gradient Descent (Alternative)

If we do not rewrite the expression $\theta = \theta - \alpha X^\top (X\theta - y)$

For each iteration, we have:

- Computing $X\theta$ is $\mathcal{O}(ND)$
- Computing $X\theta - y$ is $\mathcal{O}(N)$
- Computing $\alpha X^\top$ is $\mathcal{O}(ND)$
- Computing $\alpha X^\top (X\theta - y)$ is $\mathcal{O}(ND)$
- Computing $\theta = \theta - \alpha X^\top (X\theta - y)$ is $\mathcal{O}(D)$

What is overall computational complexity?

$\mathcal{O}(NDt)$