Linear Algebra Review

ORIE 4741

September 24, 2020

Table of Contents

Linear Independence and Dependence

Linear Independence

The sequence of vectors $v_1, \ldots, v_n \in \mathbb{R}^p$ is linear independent if the only linear combination of them that gives zero vector have all coefficients as zero. Specifically, for any $l_1, \ldots, l_n \in \mathbb{R}$,

$$l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0.$$

- Solve $l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0$ (p equations and n variables) and there is a unique solution which is a zero vector.
- v_i cannot be represented by linear combination of $v_1, \ldots, v_{i-1}, v_{i+1}, v_n$.

Linear Independence and Dependence

Linear Independence

The sequence of vectors $v_1, \ldots, v_n \in \mathbb{R}^p$ is linear independent if the only linear combination of them that gives zero vector have all coefficients as zero. Specifically, for any $l_1, \ldots, l_n \in \mathbb{R}$,

$$l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0.$$

- Solve $l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0$ (p equations and n variables) and there is a unique solution which is a zero vector.
- v_i cannot be represented by linear combination of $v_1, \ldots, v_{i-1}, v_{i+1}, v_n$.

Examples

The vectors $[1, 0], [0, 1]$ are linearly independent.

Linear Independence and Dependence

Linear Dependence

The sequence of vectors $v_1, \ldots, v_n \in \mathbb{R}^p$ is linear dependent if there exists a linear combination of them that gives zero vector which has at least one non-zero coefficient. Specifically, there exists a sequence $l_1, \ldots, l_n \in \mathbb{R}$ such that

$$l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \text{ and } l_i \neq 0 \text{ for some } i = 1, \ldots, n.$$

- Solve $l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0$ (p equations and n variables) and there is at least one non-zero solution.
- $v_i = \sum_{j \neq i} -\frac{l_j}{l_i} v_j$.

Linear Independence and Dependence

Linear Dependence

The sequence of vectors $v_1, \ldots, v_n \in \mathbb{R}^p$ is linear dependent if there exists a linear combination of them that gives zero vector which has at least one non-zero coefficient. Specifically, there exists a sequence $l_1, \ldots, l_n \in \mathbb{R}$ such that

$$l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \text{ and } l_i \neq 0 \text{ for some } i = 1, \ldots, n.$$

- Solve $l_1 v_1 + \ldots + l_n v_n = \mathbf{0} \implies l_1, \ldots, l_n = 0$ (p equations and n variables) and there is at least one non-zero solution.
- $v_i = \sum_{j \neq i} -\frac{l_j}{l_i} v_j$.

Examples

The vectors $[1, 0], [0, 1], [1, 1]$ are linearly dependent because $[1, 0] + [0, 1] - [1, 1] = 0$.

Matrix Rank

Now we have a matrix $A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1p} \\ a_{21} & a_{22} & \cdots & a_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{np} \end{bmatrix}$

- A collection of n row vectors $a_1, \dots, a_n \in \mathbb{R}^p$.
- A collection of p column vectors $A_1, \dots, A_p \in \mathbb{R}^n$.

Row Rank and Column Rank

The row rank of a matrix is the maximum number of linearly independent rows in a matrix.
The column rank of a matrix is the maximum number of linearly independent columns in a matrix.

Matrix Rank: Exercise

Compute the row rank and the column rank of the following matrices:

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 3 & 5 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 3 & -1 \\ 0 & 0 & -2 \end{bmatrix}$$

Matrix Rank: Exercise

Compute the row rank and the column rank of the following matrices:

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 3 & 5 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 3 & -1 \\ 0 & 0 & -2 \end{bmatrix}$$

- The row rank of A is 2.
- The column rank of A is also 2.
- The row rank of B is 3.
- The column rank of B is also 3.

Matrix Rank

- For every matrix, its' row rank is equal to its' column rank.

Matrix Rank

- For every matrix, its' row rank is equal to its' column rank.
- The rank of a matrix is defined as its' row rank or column rank.

Matrix Rank

- For every matrix, its' row rank is equal to its' column rank.
- The rank of a matrix is defined as its' row rank or column rank.
- A matrix of size $n \times p$ has full rank if $\text{rank}(A) = \min(n, p)$.

Matrix Rank

- For every matrix, its' row rank is equal to its' column rank.
- The rank of a matrix is defined as its' row rank or column rank.
- A matrix of size $n \times p$ has full rank if $\text{rank}(A) = \min(n, p)$.

$$A = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 3 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 3 & -1 \\ 0 & 0 & -2 \end{bmatrix}$$

- Both A and B have full rank.

Matrix Product

Suppose we have matrix $A, B \in \mathbb{R}^{n\times p}(n \neq p)$ and $C, D \in \mathbb{R}^{n\times n}$.

Dimension match AB is not defined, while $AB^\top$ and $B^\top A$ are defined.

Matrix Product

Suppose we have matrix $A, B \in \mathbb{R}^{n\times p} (n \neq p)$ and $C, D \in \mathbb{R}^{n\times n}$.

Dimension match AB is not defined, while $AB^\top$ and $B^\top A$ are defined.

- The (i, j)-th element of matrix $AB^\top$ is the inner product between i-th row of A and j-th column of $B^\top$.

Matrix Product

Suppose we have matrix $A, B \in \mathbb{R}^{n \times p} (n \neq p)$ and $C, D \in \mathbb{R}^{n \times n}$.

Dimension match AB is not defined, while $AB^\top$ and $B^\top A$ are defined.

- The (i, j)-th element of matrix $AB^\top$ is the inner product between i-th row of A and j-th column of $B^\top$.

Order matters $CD \neq DC$ in general. Actually $AB^\top \in \mathbb{R}^{n \times n}$ while $B^\top A \in \mathbb{R}^{p \times p}$.

Matrix Product

Suppose we have matrix $A, B \in \mathbb{R}^{n\times p} (n \neq p)$ and $C, D \in \mathbb{R}^{n\times n}$.

Dimension match AB is not defined, while $AB^\top$ and $B^\top A$ are defined.

- The (i,j)-th element of matrix $AB^\top$ is the inner product between i-th row of A and j-th column of $B^\top$.

Order matters $CD \neq DC$ in general. Actually $AB^\top \in \mathbb{R}^{n\times n}$ while $B^\top A \in \mathbb{R}^{p\times p}$.

- $(CD)^\top = D^\top C^\top$.
- $(C+D)A = CA + DA$.
- $CDA = (CD)A = C(DA)$.

Matrix Product Example

$$A= \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \; B= \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

$$AB = \begin{bmatrix} (1*5)+(2*7) & (1*6)+2*8 \\ (3*5)+(4*7) & (3*6)+4*8 \end{bmatrix} = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

Identity Matrix

Identity Matrix

The identity matrix, I_p, is an $p \times p$ matrix with 1s on the diagonal and 0s everywhere else.

$$\mathrm{I}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- For any vector x of size $p \times 1$, $\mathrm{I}_p x = x$
- For any matrix C of size $n \times n$, $C\mathrm{I}_p = \mathrm{I}_p C = C$.
- For any matrix A of size $n \times p$, $A\mathrm{I}_p = \mathrm{I}_n A = A$.

Matrix Inverse

Invertible Matrix

The matrix C of size $n \times n$ is invertible if there exists one matrix D such that $DC = CD = \mathrm{I}_n$.

Matrix Inverse

Invertible Matrix

The matrix C of size $n \times n$ is invertible if there exists one matrix D such that $DC = CD = \mathrm{I}_n$.

- Only square matrices can be invertible.
- C is invertible is equivalent to that C is square and has full rank.

Matrix Inverse

Invertible Matrix

The matrix C of size $n \times n$ is invertible if there exists one matrix D such that $DC = CD = \mathrm{I}_n$.

- Only square matrices can be invertible.
- C is invertible is equivalent to that C is square and has full rank.
- If C is invertible, there exists one and only one D such that $CD = DC = \mathrm{I}_n$. We write such D as C^{-1}.

Invertible Matrix Example

Let $A = \begin{bmatrix} 4 & 3 \\ 1 & 1 \end{bmatrix}$.

Then, the inverse of A is: $A^{-1} = \begin{bmatrix} 1 & -3 \\ 1 & 4 \end{bmatrix}$.

Check that A^{-1} satisfies the definition of an inverse:

$$AA^{-1} = \begin{bmatrix} 4 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & -3 \\ 1 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$A^{-1}A = \begin{bmatrix} 1 & -3 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} 4 & 3 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ Therefore, A^{-1} is the inverse of A.

Norm

For a vector $v = [v_1, \ldots, v_p]$, we commonly use the following norms:

- ℓ_2 norm: $||v||_2 = \sqrt{v_1^2 + \cdots + v_p^2}$.
- ℓ_1 norm: $||v||_1 = |v_1| + \cdots + |v_p|$.
- ℓ_∞ norm: $||v||_\infty = \max_{i=1,\ldots,p} |v_i|$.

Norm

For a vector $v = [v_1, \dots, v_p]$, we commonly use the following norms:

- ℓ_2 norm: $||v||_2 = \sqrt{v_1^2 + \cdots + v_p^2}$.
- ℓ_1 norm: $||v||_1 = |v_1| + \cdots + |v_p|$.
- ℓ_∞ norm: $||v||_\infty = \max_{i=1,\dots,p} |v_i|$.

For a matrix A of size $n \times p$, we commonly use the following norms:

- Frobeinus norm: $||A||_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^p a_{ij}^2}$.

Multivariate Derivatives

Gradients

Let $f(x_1, x_2, ..., x_p)$ be a function from $\mathbb{R}^p$ to $\mathbb{R}$. The gradient of f, $\nabla f \in \mathbb{R}^p$, has each entry corresponds to a partial derivative of f with respect to a variable of the function, $\nabla f_j = \frac{\partial f}{\partial x_j}$ for $j = 1, \ldots, p$.

- $\nabla f = (\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_p})$ is a function from $\mathbb{R}^p$ to $\mathbb{R}^p$.

Hessian Matrix

Hessian Matrix

Let $f(x_1, x_2, ..., x_p)$ be a function from $\mathbb{R}^p$ to $\mathbb{R}$. The Hessian matrix of f, $H \in \mathbb{R}^{p \times p}$, has each entry corresponds to a second order partial derivative of f, $H(i,j) = \frac{\partial^2 f}{\partial x_i \partial x_j}$.

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_p} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_p \partial x_1} & \frac{\partial^2 f}{\partial x_p \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_p \partial x_p} \end{bmatrix}$$

- The i-th row (or column) of H is the gradient of $\frac{\partial f}{\partial x_i}$, i.e. $\nabla \frac{\partial f}{\partial x_i}$.

Example

Let $f(x, y, z) = 5x + 2x^2 + xy + 3xy^2z - 4y^3 + 2yz + 4z^2 - x^4yz^2$, then

$$\begin{aligned}
\frac{\partial f}{\partial x} &= 5 + 4x + y + 3y^2z - 4x^3yz^2, \\
\frac{\partial f}{\partial y} &= x + 6xyz - 12y^2 + 2z - x^4z^2, \\
\frac{\partial f}{\partial z} &= 3xy^2 + 2y + 8z - 2x^4yz.
\end{aligned}$$

- The gradient $\nabla f = [\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}]$, and the Hessian matrix is:

$$\begin{bmatrix}
4 - 12x^2yz^2 & 1 + 6yz - 4x^3z^2 & 3y^2 - 8x^3yz \\
1 + 6yz - 4x^3z^2 & 6xz - 24y & 6xy + 2 - 2x^4z \\
3y^2 - 8x^3yz & 6xy + 2 - 2x^4z & 8 - 2x^4y
\end{bmatrix}$$

Example

Let $f(x, y, z) = 5x + 2x^2 + xy + 3xy^2z - 4y^3 + 2yz + 4z^2 - x^4yz^2$, then

$$\frac{\partial f}{\partial x} = 5 + 4x + y + 3y^2z - 4x^3yz^2,$$
$$\frac{\partial f}{\partial y} = x + 6xyz - 12y^2 + 2z - x^4z^2,$$
$$\frac{\partial f}{\partial z} = 3xy^2 + 2y + 8z - 2x^4yz.$$

Example

Let $f(x, y, z) = 5x + 2x^2 + xy + 3xy^2z - 4y^3 + 2yz + 4z^2 - x^4yz^2$, then

$$\frac{\partial f}{\partial x} = 5 + 4x + y + 3y^2z - 4x^3yz^2,$$
$$\frac{\partial f}{\partial y} = x + 6xyz - 12y^2 + 2z - x^4z^2,$$
$$\frac{\partial f}{\partial z} = 3xy^2 + 2y + 8z - 2x^4yz.$$

- The gradient $\nabla f = [\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}]$, and the Hessian matrix is:

$$\begin{bmatrix} 4 - 12x^2yz^2 & 1 + 6yz - 4x^3z^2 & 3y^2 - 8x^3yz \\ 1 + 6yz - 4x^3z^2 & 6xz - 24y & 6xy + 2 - 2x^4z \\ 3y^2 - 8x^3yz & 6xy + 2 - 2x^4z & 8 - 2x^4y \end{bmatrix}$$

Matrix Derivatives

For a vector of variable $x = [x_1, \ldots, x_p] \in \mathbb{R}^p$ and a constant symmetirc matrix A of size $p \times p$, what is the gradients of $x^\top Ax$?

Matrix Derivatives

For a vector of variable $x = [x_1, \ldots, x_p] \in \mathbb{R}^p$ and a constant symmetirc matrix A of size $p \times p$, what is the gradients of $x^\top Ax$?

- $x^\top Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij}x_ix_j = \sum_{i=1}^n a_{ii}x_i^2 + 2\sum_{1\leq i<j\leq n} a_{ij}x_ix_j$.

Matrix Derivatives

For a vector of variable $x = [x_1, \ldots, x_p] \in \mathbb{R}^p$ and a constant symmetirc matrix A of size $p \times p$, what is the gradients of $x^\top Ax$?

- $x^\top Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij}x_ix_j = \sum_{i=1}^n a_{ii}x_i^2 + 2\sum_{1\leq i<j\leq n} a_{ij}x_ix_j$.

$$\frac{\partial x^\top Ax}{\partial x_i} = 2a_{ii}x_i + 2\sum_{j\neq i} a_{ij}x_j = 2\sum_{1\leq j\leq n} a_{ij}x_j = 2x^\top a_i.$$

where $a_i \in \mathbb{R}^n$ stores the i-th row of A.

Matrix Derivatives

For a vector of variable $x = [x_1, \ldots, x_p] \in \mathbb{R}^p$ and a constant symmetirc matrix A of size $p \times p$, what is the gradients of $x^\top Ax$?

- $x^\top Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij}x_ix_j = \sum_{i=1}^n a_{ii}x_i^2 + 2\sum_{1\leq i<j\leq n} a_{ij}x_ix_j$.

$$\frac{\partial x^\top Ax}{\partial x_i} = 2a_{ii}x_i + 2\sum_{j\neq i} a_{ij}x_j = 2\sum_{1\leq j\leq n} a_{ij}x_j = 2x^\top a_i.$$

where $a_i \in \mathbb{R}^n$ stores the i-th row of A.

$$\text{Gradients:} \quad \nabla\partial x^\top Ax = 2x^\top A.$$

Matrix Derivatives

For a vector of variable $x = [x_1, \ldots, x_p] \in \mathbb{R}^p$ and a constant symmetirc matrix A of size $p \times p$, what is the gradients of $x^\top Ax$?

- $x^\top Ax = \sum_{i=1}^n \sum_{j=1}^n a_{ij}x_ix_j = \sum_{i=1}^n a_{ii}x_i^2 + 2\sum_{1\leq i<j\leq n} a_{ij}x_ix_j$.

$$\frac{\partial x^\top Ax}{\partial x_i} = 2a_{ii}x_i + 2\sum_{j\neq i} a_{ij}x_j = 2\sum_{1\leq j\leq n} a_{ij}x_j = 2x^\top a_i.$$

where $a_i \in \mathbb{R}^n$ stores the i-th row of A.

$$\text{Gradients:} \quad \nabla\partial x^\top Ax = 2x^\top A.$$

$$\text{Hessian:} \quad H = 2A.$$

Flops

Flop

A floating point operation (flop) adds, multiplies, subtracts, or divides two floating point numbers.

For $x \in \mathbb{R}^p, y \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times p}$, how many flops are needed for computing $||y - Ax||_2^2$?

Flops

Flop

A floating point operation (flop) adds, multiplies, subtracts, or divides two floating point numbers.

For $x \in \mathbb{R}^p, y \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times p}$, how many flops are needed for computing $||y - Ax||_2^2$?

- $a_1^\top x$ takes p flops for multiplies and $p - 1$ flops for adds, thus $2p - 1$ in total.
- Ax takes $(2p - 1)n$ flops in total.
- $y - AX$ takes n flops for subtracts.
- $||y - Ax||_2^2$ takes n flops for multiplies and $n - 1$ flops for add, thus $2n - 1$ in total.

In total, we need $2np + 2n - 1$ flops.

Big O Notation

Big O Notation

Let $f(n)$ be the run-time of some algorithm. If $f(n) = O(g(n))$, then there exists a constant C and a constant N such that:
$|f(n)| \leq C|g(n)|$ for all $n > N$.

In the last example, we need $2np + 2n - 1$ flops. Now assume p is a constant while n can increase to infinity.

Big O Notation

Big O Notation

Let $f(n)$ be the run-time of some algorithm. If $f(n) = O(g(n))$, then there exists a constant C and a constant N such that:
$|f(n)| \leq C|g(n)|$ for all $n > N$.

In the last example, we need $2np + 2n - 1$ flops. Now assume p is a constant while n can increase to infinity.

$$0 < 2np + 2n - 1 < 2np + 2n = (2p + 2)n.$$

Big O Notation

Big O Notation

Let $f(n)$ be the run-time of some algorithm. If $f(n) = O(g(n))$, then there exists a constant C and a constant N such that:
$|f(n)| \leq C|g(n)|$ for all $n > N$.

In the last example, we need $2np + 2n - 1$ flops. Now assume p is a constant while n can increase to infinity.

$$0 < 2np + 2n - 1 < 2np + 2n = (2p + 2)n.$$

Thus

$$2np + 2n - 1 = |2np + 2n - 1| \leq (2p + 2)|n| = (2p + 2)n$$

thus $2np + 2n - 1$ has complexity $O(n)$.

Big O Notation

Big O Notation

Let $f(n)$ be the run-time of some algorithm. If $f(n) = O(g(n))$, then there exists a constant C and a constant N such that:
$|f(n)| \leq C|g(n)|$ for all $n > N$.

In the last example, we need $2np + 2n - 1$ flops. Now assume both n, p can increase to infinity.

Big O Notation

Big O Notation

Let $f(n)$ be the run-time of some algorithm. If $f(n) = O(g(n))$, then there exists a constant C and a constant N such that:
$|f(n)| \leq C|g(n)|$ for all $n > N$.

In the last example, we need $2np + 2n - 1$ flops. Now assume both n, p can increase to infinity.

$$|2np + 2n - 1| = 2np + 2n - 1 < 2np + 2n < 2np + np = 3np$$

when $p > 2$. Thus $2np + 2n - 1$ has complexity $O(np)$.

Matrix product order matters!

- The computation complexity of matrix product AB with $A \in \mathbb{R}^{n\times p}, B \in \mathbb{R}^{p\times m}$ is $O(npm)$.

Matrix product order matters!

- The computation complexity of matrix product AB with $A \in \mathbb{R}^{n \times p}, B \in \mathbb{R}^{p \times m}$ is $O(npm)$.

With matrices $A \in \mathbb{R}^{n \times p}, C \in \mathbb{R}^{n \times n}$ and a vector $x \in \mathbb{R}^{p}$ (or $\mathbb{R}^{p \times 1}$),

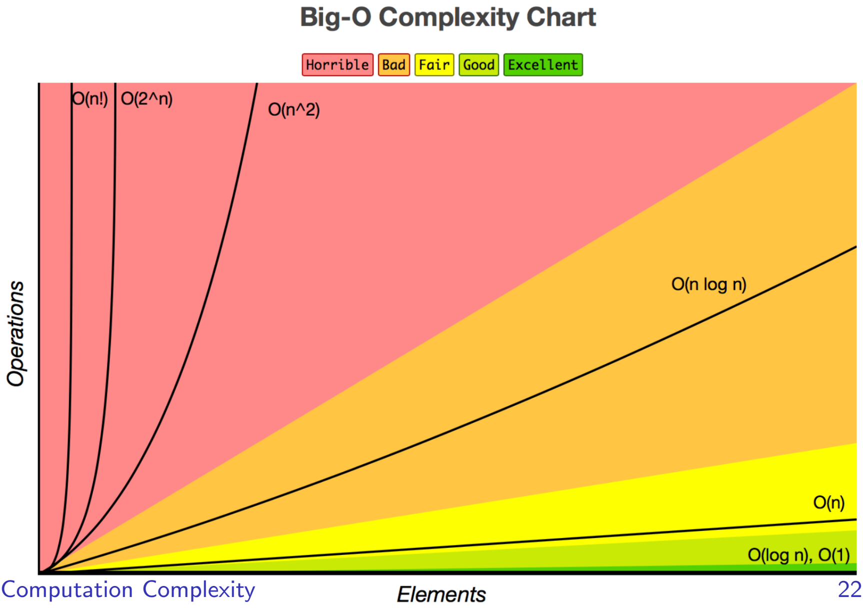
- $(CA)x$ has complexity $O(n^2 p)$ for computing CA and $O(np)$ for computing $(CA)x$ with computed (CA), thus $O(n^2 p)$ in total.

Matrix product order matters!

- The computation complexity of matrix product AB with $A \in \mathbb{R}^{n \times p}, B \in \mathbb{R}^{p \times m}$ is $O(npm)$.

With matrices $A \in \mathbb{R}^{n \times p}, C \in \mathbb{R}^{n \times n}$ and a vector $x \in \mathbb{R}^{p}$ (or $\mathbb{R}^{p \times 1}$),

- $(CA)x$ has complexity $O(n^2 p)$ for computing CA and $O(np)$ for computing $(CA)x$ with computed (CA), thus $O(n^2 p)$ in total.
- $C(Ax)$ has complexity $O(np)$ for computing Ax and $O(n^2)$ for computing $C(Ax)$ with computed (Ax), thus $O(n \max(n, p))$ in total.


Big-O Complexity Chart
Horrible
Bad
Fair
Good
Excellent
O(n!)
O(2^n)
O(n^2)
O(n log n)
O(n)
O(log n), O(1)
Operations
Elements