Linear Algebra

Vishnu Lokhande

Department of Computer Science and Engineering
University at Buffalo, SUNY
vishnulo@buffalo.edu

January 29, 2025

Some Resources for Free GPUs

https://code-love.com/2020/08/08/where-to-get-free-gpu-cloud-hours-for-machine-learning/
https://www.analyticsvidhya.com/blog/2023/02/get-free-gpu-online-to-train-your-deep-learning-model/

Preface

1. Not a comprehensive survey of all of linear algebra.
2. Focused on the subset most relevant to deep learning.

Scalar

1. A scalar is a single number.
2. Integers, real numbers, rational numbers, *etc*.
3. Usually denote it with lower-case character of italic font:

$$a, n, x \cdots$$

Vectors

1. A vector is a 1-D array of numbers:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

2. Can be real, binary, integer, *etc*.
3. Usually denote it with lower-case character of bold font.
4. Example notation for type and size: $\mathbb{R}^n, \mathbb{Z}^m$:
   - $\mathbf{x} \in \mathbb{R}^n$: $\mathbf{x}$ is a n-dimensional vector of real values.
   - $\mathbf{x} \in \mathbb{Z}^m$: $\mathbf{x}$ is a m-dimensional vector of integer values.

Matrices

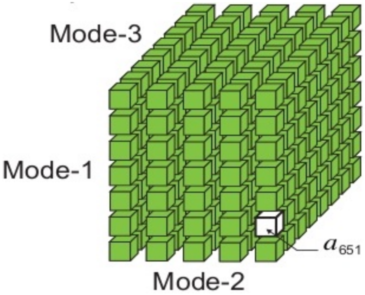
1. A matrix is a 2-D array of numbers:

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix}$$

2. Usually denote it with upper-case character of bold font.
3. Example notation for type and size: $\mathbf{A} \in \mathbb{R}^{m \times n}$:
   - $\mathbf{A} \in \mathbb{R}^{m \times n}$: $\mathbf{A}$ is a $m \times n$ matrix of real values.

Tensors

1. A tensor is an array of numbers, that may have
   - zero dimensions $\rightarrow$ a scalar.
   - one dimensions $\rightarrow$ a vector.
   - two dimensions $\rightarrow$ a matrix.
   - three dimensions $\rightarrow$ a cubic.
   - or more dimensions.



Matrix Transpose

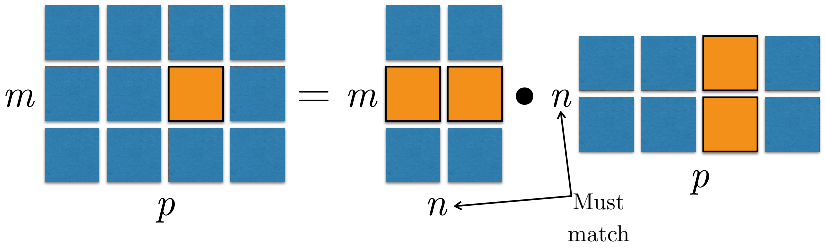
$$(\mathbf{A}^T)_{ij} = \mathbf{A}_{ji}$$

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \\ A_{31} & A_{32} \end{bmatrix} \rightarrow \mathbf{A}^T = \begin{bmatrix} A_{11} & A_{21} & A_{31} \\ A_{12} & A_{22} & A_{32} \end{bmatrix}$$

Transpose rule: $(\mathbf{A}\,\mathbf{B})^T = \mathbf{B}^T\,\mathbf{A}^T$

Matrix (Dot) Product

$$\mathbf{C} = \mathbf{A}\,\mathbf{B} \quad \Rightarrow \quad C_{ij} = \sum_k A_{ik} B_{kj}$$

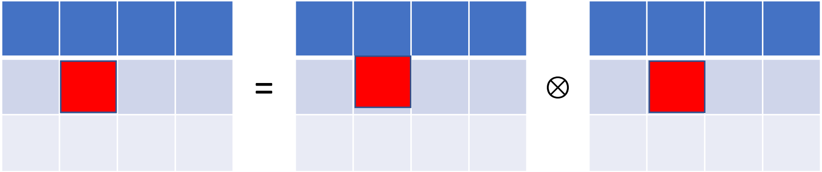


torch.matmul(input, other, *, out=None)

Matrix Hadamard product

- Also called element-wise product.

$$\mathbf{C} = \mathbf{A} \otimes \mathbf{B} \quad \Rightarrow \quad C_{ij} = A_{ij} B_{ij}$$



- Also generalize to tensor element-wise product.
- Usually seen in deep learning.

Simply use “*”

Identity Matrix

- A squared matrix, denoted as $\mathbf{I}_n$, where n represents the size (sometimes ignored).

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

- $\forall \mathbf{x} \in \mathbb{R}^n, \mathbf{I}_n \mathbf{x} = \mathbf{x}$.
- $\forall \mathbf{X} \in \mathbb{R}^{n \times m}, \mathbf{I}_n \mathbf{X} = \mathbf{X}$.

Linear Systems of Equations

$$\mathbf{A}\,\mathbf{x} = \mathbf{b}$$

expands to (Matlab format)

$$\mathbf{A}_{1:}\,\mathbf{x} = b_1$$
$$\mathbf{A}_{2:}\,\mathbf{x} = b_2$$
$$\vdots$$
$$\mathbf{A}_{m:}\,\mathbf{x} = b_m$$

Solving Systems of Equations

Span of a set of vectors $\{\mathbf{x}_1, \cdots, \mathbf{x}_n\}$

How large space the vectors can represent.

1. A linear system of equations, $\mathbf{A}\,\mathbf{x} = \mathbf{b} = \sum_i \mathbf{A}_{:,i}\, x_i$ where $\mathbf{A} \in \mathbb{R}^{m \times n}$, can have:
   - no solution: $\mathbf{b}$ is not in the span of $\{\mathbf{A}_{:,1}, \cdots, \mathbf{A}_{:,n}\}$.
   - many solutions: $\mathbf{b}$ is in the span of $\{\mathbf{A}_{:,1}, \cdots, \mathbf{A}_{:,n}\}$ and dimension of the span is less than n.
   - exactly one solution: dimension of the span of $\{\mathbf{A}_{:,1}, \cdots, \mathbf{A}_{:,n}\}$ is equal to n.
     - ★ meaning multiplication by the matrix is an invertible function.

Matrix Inversion

1. Matrix inversion:

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}_n$$

2. Solving a system using an inversion:

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$
$$\mathbf{A}^{-1}\mathbf{A}\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$
$$\mathbf{I}_n\mathbf{x} = \mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$

3. Numerically unstable, but useful for abstract analysis.

Invertibility

1. Matrix cannot be inverted if
   - more rows than columns.
   - more columns than rows.
   - same number of rows and columns but with redundant rows / columns (linearly dependent, low rank).

Norms

1. The norm of a vector is a functions f that measure how "large" a vector is.
2. Similar to a distance between zero and the point represented by the vector.
   - $f(\mathbf{x}) = 0 \Rightarrow \mathbf{x} = 0$.
   - $f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$ (the triangle inequality).
   - $\forall \alpha \in \mathbb{R}, f(\alpha \, \mathbf{x}) = |\alpha| f(\mathbf{x})$.

Norms

- L^p norm

$$\| \mathbf{x} \|_p = \left(\sum_i |x_i|^p \right)^{1/p}$$

- Most popular norm: L^2 norm, $p = 2$.
- L^1 norm, $p = 1$, $\| \mathbf{x} \|_1 = \sum_i |x_i|$.
- Max norm: infinite p: $\| \mathbf{x} \|_\infty = \max_i |x_i|$.

Special Matrices and Vectors

1. Unit vector:

$$\| \mathbf{x} \|_2 = 1$$

2. Symmetric matrix:

$$\mathbf{A} = \mathbf{A}^T$$

3. Orthogonal matrix:

$$\mathbf{A}^T \mathbf{A} = \mathbf{A} \mathbf{A}^T = \mathbf{I}$$
$$\mathbf{A}^{-1} = \mathbf{A}^T$$

Eigendecomposition

1. Eigenvector and eigenvalue of $\mathbf{A}$:

$$\mathbf{A}\,\mathbf{v}_i = \lambda_i\,\mathbf{v}_i$$

   - $\mathbf{v}_i$ is an eigenvector of $\mathbf{A}$.
   - λ_i is the corresponding eigenvalue.

2. Eigendecomposition of a matrix:

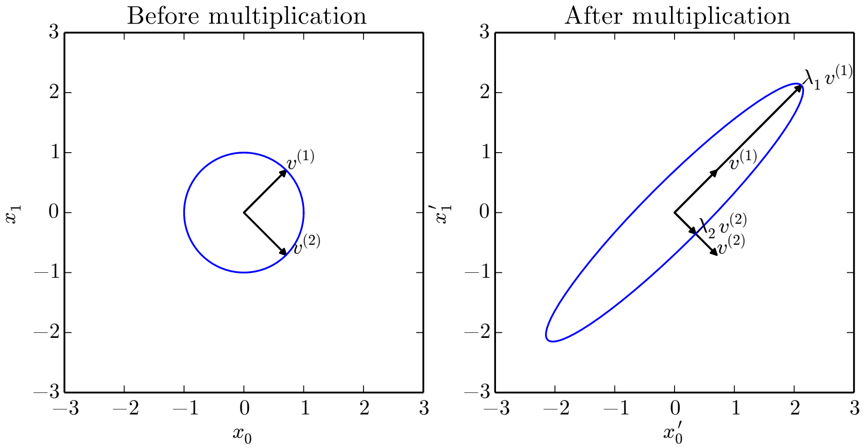
$$\mathbf{A} = \mathbf{V}\,\mathrm{diag}(\lambda)\,\mathbf{V}^{-1}$$

3. Every real symmetric matrix has a real, orthogonal eigendecomposition:

$$\mathbf{A} = \mathbf{Q}\underbrace{\Lambda}_{\text{diagonal matrix}}\mathbf{Q}^T$$

Effect of Eigenvalues

$$\mathbf{A} = \mathbf{V}\,\mathrm{diag}(\lambda)\,\mathbf{V}^{-1}$$



Singular Value Decomposition

1. Similar to eigendecomposition.
2. More general; matrix need not be square.

$$\mathbf{A} = \mathbf{U}\,\mathbf{D}\,\mathbf{V}^T$$

- $\mathbf{U}, \mathbf{V}$ are orthogonal matrices; $\mathbf{D}$ is a diagonal matrix.

Moore-Penrose Pseudoinverse

$$\mathbf{x} = \mathbf{A}^{+}\,\mathbf{y}$$

1. If the equation $\mathbf{A}\,\mathbf{x} = \mathbf{y}$ has:
    - exactly one solution: this is the same as the inverse.
    - no solution: this gives us the solution with the smallest error: $\mathbf{A}^{+} \triangleq \arg\min_{\mathbf{A}} \| \mathbf{A}\,\mathbf{x} - \mathbf{y} \|_2$.
    - many solutions: this gives us the solution with the smallest norm of $\mathbf{x}$.

Computing the Pseudoinverse

1. The SVD allows the computation of the pseudoinverse:

$$\mathbf{A} = \mathbf{U}\,\mathbf{D}\,\mathbf{V}^T \Rightarrow \mathbf{A}^+ = \mathbf{V}\,\mathbf{D}^+\,\mathbf{U}^T$$

   - $\mathbf{D}^+$ contains the reciprocal of non-zero entries of $\mathbf{D}$.

Trace

$$\mathrm{Tr}(\mathbf{A}) = \sum_i \mathbf{A}_{ii}$$

$$\mathrm{Tr}(\mathbf{A\,B\,C}) = \mathrm{Tr}(\mathbf{C\,A\,B}) = \mathrm{Tr}(\mathbf{B\,C\,A}) \quad \text{(rotation invariant)}$$

Derivative

1. The derivative of a matrix $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$ w.r.t. a matrix $\mathbf{B} \in \mathbb{R}^{m_1 \times m_2}$ is a tensor $\mathbf{Z}$ of size $n_1 \times n_2 \times m_1 \times m_2$, with elements

$$\mathbf{Z}_{ijk\ell} = \frac{\mathrm{d}\,\mathbf{A}_{ij}}{\mathrm{d}\,\mathbf{B}_{k\ell}}$$

   - If $\mathbf{A}$ is a scalar, $\mathbf{B}$ is a vector of size m, $\frac{\mathrm{d}\,\mathbf{A}}{\mathrm{d}\,\mathbf{B}}$ is a vector of size m.
   - If $\mathbf{A}$ is a scalar, $\mathbf{B}$ is a matrix of size $m_1 \times m_2$, $\frac{\mathrm{d}\,\mathbf{A}}{\mathrm{d}\,\mathbf{B}}$ is a matrix of size $m_1 \times m_2$.
2. In deep learning, we usually face with the problem of calculating derivatives of an objective function (a scalar) w.r.t. a matrix/vector.