Linear Algebra Primer Part 2

Liangcheng Tao, Vivian Hoang-Dung Nguyen, Roma Dziembaj, Sona Allahverdiyeva
Department of Computer Science
Stanford University
Stanford, CA 94305
{lctao13, vnguyen2, romad, sonakhan}@cs.stanford.edu

1 Vectors and Matrices Recap

1.1 Vector

A column vector $\mathbf{v} \in \mathbb{R}^{\mathrm{nx1}}$ where

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

A row vector $\mathbf{v}^T \in \mathbb{R}^{\mathrm{1xn}}$ where

$$\mathbf{v}^T = \begin{bmatrix} v_1 & v_2 & \cdots & v_n \end{bmatrix}$$

T denotes the transpose operation.

The **norm** is

$$||x||_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

Formally, the norm can also be defined as any function $f : \mathbb{R}^{\mathrm{n}} \mapsto \mathbb{R}$ that satisfies 4 properties:

- **Non-negativity:** For all $x \in \mathbb{R}^{\mathrm{n}}$, $f(x) \geq 0$
- **Definiteness:** $f(x) = 0$ if and only if $x = 0$
- **Homogeneity:** For all $x \in \mathbb{R}^{\mathrm{n}}$, $t \in \mathbb{R}$, $f(tx) = |t|f(x)$
- **Triangle inequality:** For all $x, y \in \mathbb{R}^{\mathrm{n}}$, $f(x + y) \leq f(x) + f(y)$

1.1.1 Projection

A **projection** is an inner product (dot product) of vectors. If B is a unit vector, then $A \cdot B$ gives the length of A which lies in the direction of B.

1.2 Matrix

A matrix $\mathbf{A} \in \mathbb{R}^{\mathrm{mxn}}$ is an array of numbers with size m by n.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & & & & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}$$

If $m = n$, we say that $\mathbf{A}$ is square.

1.2.1 An Application of Matrices

Grayscale images have one number per pixel and are stored as an mxn matrix. Color images have 3 numbers per pixel - red, green, and blue brightnesses (RGB), and are stored as an $mxnx3$ matrix.

2 Transformation Matrices

Matrices can be used to transform vectors in useful ways, through multiplication: x'= Ax. The simplest application of that is through scaling, or multiplying a scaling matrix with scalars on its diagonal by the vector.

We can also use matrices to rotate vectors. When we multiply a matrix and a vector, the resulting x coordinate is the original vector **dot** the first row.

In order to rotate a vector by an angle θ, counter-clockwise, we see that we need our

$x' = cos\theta x - sin\theta y$ and

$y' = cos\theta y + sin\theta x$

therefore, we multiply it by the matrix

$$M = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

which gives us that P' = R P.

We can also use multiple matrices to transform a point. For example, $p' = R_2 R_1 Sp$. The transformations are applied one after another, from right to left. In our example, this would be $(R_2(R_1(Sp)))$.

In order to also be able to translate vectors, we have to implement a somewhat hacky solution of adding a "1" at the very end of the vector. That way, with these "homogeneous coordinates", we can also translate vectors. The multiplication works out so the rightmost column of our matrix gets added to the respective coordinates. A homogenous matrix will have [0 0 1] in the bottom row to ensure that the elements get added correctly, and the resulting vector has '1' at the bottom, too.

By convention, in homogeneous coordinates, we divide the result by its last coordinate after doing matrix multiplication.

$$\begin{bmatrix} x \\ y \\ 7 \end{bmatrix}$$

$$\begin{bmatrix} x/7 \\ y/7 \\ 1 \end{bmatrix}$$

So to obtain the result of $P(x, y) -> P' = (s_x x, s_y y)$

we have to first $P = (x, y) -> (x, y, 1)$ and then $P' = (s_x x, s_y y) -> (s_x x, s_y y, 1)$ so we can then do the matrix multiplication S*P. Though, we have to note that scaling and translating is not the same as translating and scaling. In other words, $T * S * P \neq S * T * P$

Any rotation matrix R belongs to the cateogry of normal matrices that satisfies interesting properties. For example, $R\dot{R}^T = I$ and $det(R) = 1$

The rows of a rotation matrix are always mutually perpendicular (a.k.a. orthogonal) unit vectors - this is what allows for it to satisfy some of the few unique properties mentioned above.

3 Matrix Inverse

Given a matrix A, its inverse A^{-1} is a matrix such that:

$$AA^{-1} = A^{-1}A = I$$

where I is the identity matrix of the same size.

An example of a matrix inverse is:

$$\begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{3} \end{bmatrix}$$

A matrix does not necessarily have an inverse. If A^{-1} exists, A is known as *invertible* or *non-singular*.

Some useful identities for matrices that are invertible are:

- $(A^{-1})^{-1} = A$
- $(AB)^{-1} = B^{-1}A^{-1}$
- $A^{-T} \triangleq (A^T)^{-1} = (A^{-1})^T$

3.1 Pseudoinverse

Frequently in linear algebra problems, you want to solve the equation $AX = B$ for X. You would like to be able to compute A^{-1} and multiply both sides to get $X = A^{-1}B$. In python, this command would be: np.linalg.inv(A)*B.

However, for large floating point matrices, calculating inverses can be very expensive and possibly inaccurate. An inverse could also not even exist for A. What should we do?

Luckily, we have what is known as a *pseudoinverse*. This other matrix can be used to solve for AX=B. Python will try many methods, including using the pseudoinverse, if you use the following command: np.linalg.solve(A,B). Additionally, with the pseudoinverse, even if there is no solution to $AX = B$, Python can find the closest solution instead.

4 Matrix Rank

- The rank of a transformation matrix A tells you how many dimensions it transforms a matrix to.
- col-rank(A) = maximum number of linearly independent column vectors of A
- row-rank(A) = maximum number of linearly independent row vectors of A. Column rank always equals row rank.
- For transformation matrices, the rank tells you the dimensions of the output.
- For instance, if rank of A is 1, then the transformation $p' = Ap$ points onto a line.
- Full rank matrix- if mxm and rank is m
- Singular matrix- if mxm matrix rank is less than m, because at least one dimension is getting collapsed. (No way to tell what input was from result) –> inverse does not exist for non-square matrices.

5 Eigenvalues and Eigenvectors (SVD)

5.1 Definitions

An *eigenvector* **x** of a linear transformation A is a non-zero vector that, when A is applied to it, does not change its direction. Applying A to the eigenvector scales the eigenvector by a scalar value λ, called an *eigenvalue*.

The following equation describes the relationship between eigenvalues and eigenvectors:

$$A\mathbf{x} = \lambda\mathbf{x}, \quad \mathbf{x} \neq \mathbf{0}$$

5.2 Finding eigenvectors and eigenvalues

If we want to find the eigenvalues of A, we can manipulate the above definition as follows:

$$\begin{aligned} A\mathbf{x} &= \lambda\mathbf{x}, \quad \mathbf{x} \neq \mathbf{0} \\ A\mathbf{x} &= (\lambda I\mathbf{x}), \quad \mathbf{x} \neq \mathbf{0} \\ (\lambda I - A)\mathbf{x} &= \mathbf{0}, \quad \mathbf{x} \neq \mathbf{0} \end{aligned}$$

Since we are looking for non-zero **x**, we can equivalently write the above relation as:

$$|\lambda I - A| = \mathbf{0}$$

Solving this equation for λ gives the eigenvalues of A, and these can be substituted back into the original equation to find the corresponding eigenvectors.

5.3 Properties

- The trace of A is equal to the sum of its eigenvalues:
$$tr(A) = \sum_{i=1}^{n} \lambda_i$$
- The determinant of A is equal to the product of its eigenvalues:
$$|A| = \prod_{i=1}^{n} \lambda_i$$
- The rank of A is equal to the number of non-zero eigenvalues of A.
- The eigenvalues of a diagonal matrix $D = diag(d_1, \ldots, d_n)$ are just the diagonal entries $d_1, \ldots d_n$.

5.4 Spectral Theory

5.4.1 Definitions

- An *eigenpair* is the pair of an eigenvalue and its associated eigenvector.
- An *eigenspace* of A associated with λ is the space of vectors where:
$$(A - \lambda I) = 0$$
- The *spectrum* of A is the set of all its eigenvalues:
$$\sigma(A) = \{\lambda \in \mathbb{C} : \lambda I - A \text{ is singular}\}$$
Where $\mathbb{C}$ is the space of all eigenvalues of A
- The *spectral radius* of A is the magnitude of its largest magnitude eigenvalue:
$$\rho(A) = max\{|\lambda_1|, \ldots, |\lambda_n|\}$$

5.4.2 Theorem: Spectral radius bound

Spectral radius is bounded by the infinity norm of a matrix:

$$\rho(A) = \lim_{k\to\infty} ||A^k||^{1/k}$$

Proof:

$$|\lambda|^k||\mathbf{v}|| = ||\lambda|^k\mathbf{v}|| = ||A^k\mathbf{v}||$$

By the Cauchy–Schwarz inequality ($||\mathbf{uv}|| \leq ||\mathbf{u}|| \cdot ||\mathbf{v}||$):

$$|\lambda|^k||\mathbf{v}|| \leq ||A^k|| \cdot ||\mathbf{v}||$$

Since $\mathbf{v} \neq 0$:

$$|\lambda|^k \leq ||A^k||$$

And we thus arrive at:

$$\rho(A) = \lim_{k\to\infty} ||A^k||^{1/k}$$

5.5 Diagonalization

An nxn matrix A is diagonalizable if it has n linearly independent eigenvectors.

Most square matrices are diagonalizable

- Normal matrices are diagonalizable
  Note: Normal matrices are matrices that satisfy:
  $$A^*A = AA^*$$
  Where A^* is the complex conjugate of A
- Matrices with n distinct eigenvalues are diagonalizable

Lemma: Eigenvectors associated with distinct eigenvalues are linearly independent.

To diagonalize the matrix A, consider its eigenvalues and eigenvectors. We can construct matrices D and V, where D is the diagonal matrix of the eigenvalues of A, and V is the matrix of corresponding eigenvectors:

$$D = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

$$V = [v_1 \quad v_2 \quad \dots \quad v_n]$$

Since we know that:

$$AV = VD$$

We can diagonalize A by:

$$A = VDV^{-1}$$

If all eigenvalues are unique, then V is orthogonal. Since the inverse of an orthogonal matrix is its transpose, we can write the diagonalization as:

$$A = VDV^T$$

5.6 Symmetric Matrices

If A is symmetric, then all its eigenvalues are real, and its eigenvectors are orthonormal. Recalling the above diagonalization equation, we can diagonalize A by:

$$A = VDV^T$$

Using the above relation, we can also write the following relationship:

Given $y = V^T x$:

$$x^T Ax = x^T VDV^T x = y^T Dy = \sum_{i=1}^{n} \lambda_i y_i^2$$

Thus, if we want to do the following maximization:

$$max_{x \in \mathbb{R}^n}(x^T Ax) \quad \text{subject to } ||x||_2^2 = 1$$

Then the maximizing x can be found by finding the eigenvector corresponding to the largest eigenvalue of A.

5.7 Applications

Some applications of eigenvalues and eigenvectors are:

- PageRank
- Schroedinger's equation
- Principle component analysis (PCA)

6 Matrix Calculus

6.1 The Gradient

If a function $f : \mathbb{R}^{\text{mxn}} \mapsto \mathbb{R}$ takes as input a matrix A of size mxn and returns a real value, then the gradient of f is

$$\nabla_A f(A) \in \mathbb{R}^{\text{mxn}} = \begin{bmatrix} \frac{\partial f(A)}{\partial A_{11}} & \frac{\partial f(A)}{\partial A_{12}} & \cdots & \frac{\partial f(A)}{\partial A_{1n}} \\ \frac{\partial f(A)}{\partial A_{21}} & \frac{\partial f(A)}{\partial A_{22}} & \cdots & \frac{\partial f(A)}{\partial A_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(A)}{\partial A_{m1}} & \frac{\partial f(A)}{\partial A_{m2}} & \cdots & \frac{\partial f(A)}{\partial A_{mn}} \end{bmatrix}$$

Every entry in the matrix is:

$$\nabla_A f(A)_{ij} = \frac{\partial f(A)}{\partial A_{ij}}$$

The size of $\nabla_A f(A)$ is always the same as the size of A. Thus, if A is a vector x:

$$\nabla_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$

6.2 The Gradient: Properties

- $\nabla_x(f(x) + g(x)) = \nabla_x f(x) + \nabla_x g(x)$
- For $t \in \mathbb{R}$, $\nabla_x(tf(x)) = t\nabla_x f(x)$

6.3 The Hessian

The Hessian matrix with respect to x can be written as $\nabla_x^2 f(x)$ or as H. It is an nxn matrix of partial derivatives

$$\nabla_x^2 f(x) \in \mathbb{R}^{\text{nxn}} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$$

Every entry in the matrix is:

$$\nabla_x^2 f(x)_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$$

It's important to note that the Hessian is the gradient of **every entry** of the gradient of the vector. For instance, the first column of the Hessian is the gradient of $\frac{\partial f(x)}{\partial x_1}$.

6.4 The Hessian: Properties

Schwarz's theorem: The order of partial derivatives doesn't matter so long as the second derivative exists and is continuous.

Thus, the Hessian is always symmetric:

$$\frac{\partial^2 f(x)}{\partial x_i \partial x_j} = \frac{\partial^2 f(x)}{\partial x_j \partial x_i}$$

6.5 Example Calculations

6.5.1 Example Gradient Calculation

For $x \in \mathbb{R}^n$, let $f(x) = b^T x$ for some known vector $b \in \mathbb{R}^n$

$$f(x) = [b_1 \quad b_2 \quad \cdots \quad b_n]^T \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Thus,

$$f(x) = \sum_{i=1}^{n} b_i x_i$$

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_{i=1}^{n} b_i x_i = b_k$$

Therefore, we can conclude that: $\nabla_x b^T x = b$.

6.5.2 Example Hessian Calculation

Consider the quadratic function $f(x) = x^T A x$

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j$$

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j$$

$$= \frac{\partial}{\partial x_k} [\sum_{i \neq k} \sum_{j \neq k} A_{ij} x_i x_j + \sum_{i \neq k} A_{ik} x_i x_k + \sum_{j \neq k} A_{kj} x_k x_j + A_{kk} x_k^2]$$

$$= \sum_{i \neq k} A_{ik} x_i + \sum_{j \neq k} A_{kj} x_j + 2A_{kk} x_k$$

$$= \sum_{i=1}^{n} A_{ik} x_i + \sum_{j=1}^{n} A_{kj} x_j = 2 \sum_{i=1}^{n} A_{ki} x_i$$

$$\frac{\partial^2 f(x)}{\partial x_k \partial x_l} = \frac{\partial}{\partial x_k} [\frac{\partial f(x)}{\partial x_l}] = \frac{\partial}{\partial x_k} [\sum_{i=1}^{n} 2A_{li} x_i]$$

$$= 2A_{lk} = 2A_{kl}$$

Thus,

$$\nabla_x^2 f(x) = 2A$$