Lecture 7: Mathematics for AI (an introduction)

Matrices, Least Squares, Weights and Activation Functions

Contents of lecture 7

Part 1 – Matrices

- What is a matrix?
- Dimensions and notation
- Addition and scalar multiplication
- Matrix–vector multiplication
- Matrix–matrix multiplication
- Transpose, identity and inverse
- Matrices as transformations

Part 2 – Least squares and AI

- Overdetermined systems
- Residuals and squared error
- Least-squares principle
- Geometric interpretation
- Normal equations
- Weights and activation functions
- A simple chemical example
- Connection to AI

Part 1: why do we need matrices?

Consider the system

$$x + 2y = 5,$$
$$3x - y = 4.$$

We can write this as

$$\begin{pmatrix} 1 & 2 \\ 3 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 5 \\ 4 \end{pmatrix}.$$

In short,

$$\boxed{A\boldsymbol{x} = \boldsymbol{b}.}$$

A matrix gives us a compact way to represent

- many numbers;
- many equations;
- linear relationships between variables.

What is a matrix?

A matrix is a rectangular array of numbers:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix}.$$

It has

$$\boxed{\text{2 rows and 3 columns.}}$$

Therefore,

$$\boxed{A \in \mathbb{R}^{2\times 3}.}$$

The entry in row i and column j is denoted by

$$a_{ij}.$$

For example,

$$A = \begin{pmatrix} 2 & -1 & 4 \\ 0 & 3 & 5 \end{pmatrix}$$

has

$$a_{12} = -1, \qquad a_{23} = 5.$$

Vectors are special matrices

A column vector is an $n \times 1$ matrix:

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

A row vector is a $1 \times n$ matrix:

$$\boldsymbol{x}^T = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}.$$

The transpose changes rows into columns:

$$\boxed{\boldsymbol{x}^T = \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}}$$

when

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

Matrix addition

Matrices can be added when they have the same dimensions.

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \qquad B = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}.$$

Then

$$A + B = \begin{pmatrix} 1+5 & 2+6 \\ 3+7 & 4+8 \end{pmatrix} = \begin{pmatrix} 6 & 8 \\ 10 & 12 \end{pmatrix}.$$

In general,

$$\boxed{(A+B)_{ij} = a_{ij} + b_{ij}.}$$

Addition is therefore **componentwise**.

Scalar multiplication

A matrix can be multiplied by a scalar:

$$3\begin{pmatrix} 1 & 2 \\ -1 & 4 \end{pmatrix} = \begin{pmatrix} 3 & 6 \\ -3 & 12 \end{pmatrix}.$$

In general,

$$\boxed{(\alpha A)_{ij} = \alpha a_{ij}.}$$

The same idea applies to vectors:

$$-2\begin{pmatrix} 1 \\ 3 \\ -1 \end{pmatrix} = \begin{pmatrix} -2 \\ -6 \\ 2 \end{pmatrix}.$$

So matrix arithmetic extends familiar vector arithmetic.

Matrix–vector multiplication

Let

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \qquad \boldsymbol{x} = \begin{pmatrix} 5 \\ 1 \end{pmatrix}.$$

Then

$$A\boldsymbol{x} = \begin{pmatrix} 1 \cdot 5 + 2 \cdot 1 \\ 3 \cdot 5 + 4 \cdot 1 \end{pmatrix}.$$

Therefore

$$\boxed{A\boldsymbol{x} = \begin{pmatrix} 7 \\ 19 \end{pmatrix}.}$$

The matrix acts on the vector and produces another vector.

$$\boxed{A : \mathbb{R}^n \longrightarrow \mathbb{R}^m.}$$

Matrix–vector multiplication: another view

Write the columns of A as

$$A = \begin{pmatrix} | & | \\ \boldsymbol{a}_1 & \boldsymbol{a}_2 \\ | & | \end{pmatrix}.$$

Then

$$\boxed{A\boldsymbol{x} = x_1\boldsymbol{a}_1 + x_2\boldsymbol{a}_2.}$$

For example,

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} 5 \\ 1 \end{pmatrix} = 5 \begin{pmatrix} 1 \\ 3 \end{pmatrix} + \begin{pmatrix} 2 \\ 4 \end{pmatrix}.$$

Thus

$$\boxed{A\boldsymbol{x} \text{ is a linear combination of the columns of } A.}$$

This observation will be important in least squares.

Matrix multiplication: dimensions

Suppose

$$A \in \mathbb{R}^{m \times n}, \qquad B \in \mathbb{R}^{n \times p}.$$

Then

$$\boxed{AB \in \mathbb{R}^{m \times p}.}$$

The dimensions must match:

$$\boxed{(m \times n)(n \times p) = (m \times p).}$$

The two **inner dimensions** must be equal.
For example,

$$(3 \times 2)(2 \times 4) = (3 \times 4).$$

But

$$(3 \times 2)(4 \times 3)$$

is not defined.

Matrix multiplication: entries

The entry in row i, column j of AB is

$$\boxed{(AB)_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}.}$$

For example,

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \qquad B = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}.$$

Then

$$AB = \begin{pmatrix} 1 \cdot 5 + 2 \cdot 7 & 1 \cdot 6 + 2 \cdot 8 \\ 3 \cdot 5 + 4 \cdot 7 & 3 \cdot 6 + 4 \cdot 8 \end{pmatrix}$$

and hence

$$\boxed{AB = \begin{pmatrix} 19 & 22 \\ 43 & 50 \end{pmatrix}.}$$

Important: AB is usually not BA

For the matrices above,

$$AB = \begin{pmatrix} 19 & 22 \\ 43 & 50 \end{pmatrix}.$$

But

$$BA = \begin{pmatrix} 23 & 34 \\ 31 & 46 \end{pmatrix}.$$

Therefore

$$\boxed{AB \neq BA.}$$

Matrix multiplication is generally **not commutative**.

Compare this with ordinary numbers:

$$2 \cdot 3 = 3 \cdot 2.$$

For matrices, changing the order can change the result.

The transpose

For

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix},$$

the transpose is

$$A^T = \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix}.$$

Thus

$$\boxed{(A^T)_{ij} = A_{ji}.}$$

Important properties are

$$(A + B)^T = A^T + B^T,$$

$$(\alpha A)^T = \alpha A^T,$$

and

$$\boxed{(AB)^T = B^T A^T.}$$

Notice the reversed order in the last formula.

The identity matrix

The identity matrix is the matrix analogue of the number 1.
For example,

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

It satisfies

$$AI = IA = A.$$

For a vector,

$$I\boldsymbol{x} = \boldsymbol{x}.$$

Thus

the identity matrix does nothing.

For every n there is an $n \times n$ identity matrix I_n.

The inverse matrix

For a square matrix A, an inverse A^{-1} satisfies

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

This is analogous to

$$a\frac{1}{a} = 1.$$

If

$$A\boldsymbol{x} = \boldsymbol{b}$$

and A^{-1} exists, then

$$\boxed{\boldsymbol{x} = A^{-1}\boldsymbol{b}.}$$

However,

$$\boxed{\text{not every matrix has an inverse.}}$$

A matrix must satisfy certain conditions, such as being square and non-singular.

Matrices as transformations

A matrix can be viewed as a function:

$$\boxed{A : \mathbb{R}^n \to \mathbb{R}^m.}$$

It takes

$$\boldsymbol{x} \in \mathbb{R}^n$$

and produces

$$A\boldsymbol{x} \in \mathbb{R}^m.$$

Examples of transformations include

- rotations;
- stretching;
- projections;
- changing coordinates.

This viewpoint will be useful when interpreting least squares geometrically.

Summary of Part 1

We have encountered

$$A + B, \qquad \alpha A, \qquad A\boldsymbol{x}, \qquad AB, \qquad A^T, \qquad A^{-1}.$$

The central idea is

$$\boxed{A\boldsymbol{x}}$$

as a linear transformation.

Matrices can represent

- systems of equations;
- transformations;
- collections of measurements;
- linear models;
- data.

Matrices are fundamental to AI.

Part 2: why do we need least squares?

Suppose we have measurements

$$(x_1, y_1), \ldots, (x_m, y_m).$$

We want a straight line

$$y = ax + b.$$

For every measurement we would like

$$ax_i + b = y_i.$$

Thus,

$$\begin{aligned} ax_1 + b &= y_1, \\ ax_2 + b &= y_2, \\ &\vdots \\ ax_m + b &= y_m. \end{aligned}$$

Usually there is no exact solution.
Why?

Measurements contain noise.

From a line to a matrix

Define

$$\boldsymbol{x} = \begin{pmatrix} a \\ b \end{pmatrix}.$$

Then

$$\begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_m & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \approx \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}.$$

In short,

$$\boxed{A\boldsymbol{x} \approx \boldsymbol{b}.}$$

The approximation sign is important.

Usually,

$$A\boldsymbol{x} = \boldsymbol{b}$$

has no solution.
So we ask:

Which $\boldsymbol{x}$ makes $A\boldsymbol{x}$ as close as possible to $\boldsymbol{b}$?

Residuals

Define the residual vector

$$\boxed{\boldsymbol{r} = \boldsymbol{b} - A\boldsymbol{x}.}$$

Its entries are the individual errors:

$$r_i = b_i - (A\boldsymbol{x})_i.$$

We want the residuals to be small.
A natural measure is

$$\|\boldsymbol{r}\|^2 = r_1^2 + \cdots + r_m^2.$$

Therefore the least-squares problem is

$$\boxed{\min_{\boldsymbol{x}} \|A\boldsymbol{x} - \boldsymbol{b}\|^2.}$$

This means:

$$\boxed{\text{minimize the total squared error.}}$$

Why square the errors?

Suppose the errors are

$$-2, \qquad 1, \qquad 3.$$

Adding them gives

$$-2 + 1 + 3 = 2.$$

Positive and negative errors can cancel.

Instead,

$$(-2)^2 + 1^2 + 3^2 = 4 + 1 + 9 = 14.$$

Squaring has two useful effects:

- all contributions are non-negative;
- large errors receive relatively more weight.

Thus

$$\boxed{J(\boldsymbol{x}) = \|A\boldsymbol{x} - \boldsymbol{b}\|^2}$$

is a natural objective function.

A simple numerical example

Suppose

x	y
1	2.1
2	3.9
3	6.2

We want

$$y \approx ax + b.$$

The matrix formulation is

$$\begin{pmatrix} 1 & 1 \\ 2 & 1 \\ 3 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} \approx \begin{pmatrix} 2.1 \\ 3.9 \\ 6.2 \end{pmatrix}.$$

There are

3 equations

and only

2 unknowns.

Therefore the system is **overdetermined**.

The least-squares question

We cannot satisfy all equations exactly.
Instead we solve

$$\min_{a,b} \sum_{i=1}^{m} (ax_i + b - y_i)^2 .$$

Equivalently,

$$\min_{\boldsymbol{x}} \|A\boldsymbol{x} - \boldsymbol{b}\|^2.$$

The result is the line that gives the smallest total squared error.

So least squares turns

an impossible exact problem

into

an optimization problem.

Least squares as a projection

Every vector of the form

$$A\boldsymbol{x}$$

lies in the column space of A:

$$A\boldsymbol{x} \in \text{col}(A).$$

But generally,

$$\boldsymbol{b} \notin \text{col}(A).$$

Least squares finds the vector

$$A\hat{\boldsymbol{x}}$$

that is closest to $\boldsymbol{b}$.
Thus

$$\boxed{A\hat{\boldsymbol{x}} = \text{proj}_{\text{col}(A)}\, \boldsymbol{b}.}$$

This gives a geometric interpretation:

$$\boxed{\text{least squares} = \text{orthogonal projection}.}$$

The residual is orthogonal

At the least-squares solution,

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|A\boldsymbol{x} - \boldsymbol{b}\|^2.$$

Define

$$\boldsymbol{r} = \boldsymbol{b} - A\hat{\boldsymbol{x}}.$$

The residual is orthogonal to every column of A.
Therefore,

$$\boxed{A^T\boldsymbol{r} = 0.}$$

Substituting the residual gives

$$A^T(\boldsymbol{b} - A\hat{\boldsymbol{x}}) = 0.$$

The normal equations

From

$$A^T(\boldsymbol{b} - A\hat{\boldsymbol{x}}) = 0$$

we obtain

$$A^T\boldsymbol{b} - A^TA\hat{\boldsymbol{x}} = 0.$$

Therefore

$$\boxed{A^TA\hat{\boldsymbol{x}} = A^T\boldsymbol{b}.}$$

These are the

$$\boxed{\text{normal equations.}}$$

If A^TA is invertible,

$$\boxed{\hat{\boldsymbol{x}} = (A^TA)^{-1}A^T\boldsymbol{b}.}$$

A numerical point

The formula

$$\hat{\boldsymbol{x}} = (A^T A)^{-1} A^T \boldsymbol{b}$$

is mathematically useful.

But in numerical computation we usually do **not** calculate the inverse explicitly. Instead, methods such as

QR factorization

or

singular value decomposition (SVD)

are commonly used.

The important idea for today is not the numerical algorithm, but

find the parameters that minimize the error.

From linear models to AI

Suppose an experiment produces several measurements:

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

For example,

$$\boldsymbol{x} = \begin{pmatrix} \text{temperature} \\ \text{pressure} \\ \text{concentration} \end{pmatrix}.$$

A simple model is

$$z = w_1 x_1 + w_2 x_2 + w_3 x_3 + b.$$

In vector notation,

$$\boxed{z = \boldsymbol{w}^T \boldsymbol{x} + b.}$$

What are weights?

The vector

$$\boxed{\boldsymbol{w} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}}$$

contains the **weights**.
Consider

$$z = 0.2x_1 + 2.5x_2 - 0.7x_3 + b.$$

Then:

- $w_1 = 0.2$: relatively small positive influence;
- $w_2 = 2.5$: strong positive influence;
- $w_3 = -0.7$: negative influence.

The weights are parameters of the model.
Therefore learning means, in part,

finding suitable weights from data.

One artificial neuron

A very simple artificial neuron has two steps.

1. Weighted sum

$$z = \boldsymbol{w}^T \boldsymbol{x} + b.$$

2. Activation

$$y = \sigma(z).$$

Thus

$$\boxed{y = \sigma\left(\boldsymbol{w}^T \boldsymbol{x} + b\right).}$$

Here

- $\boldsymbol{x}$ contains the inputs;
- $\boldsymbol{w}$ contains the weights;
- b is the bias;
- σ is the activation function.

Why do we need an activation function?

Suppose we only use

$$y = \boldsymbol{w}^T \boldsymbol{x} + b.$$

This is a linear function.

Even several linear layers remain linear.
For example,

$$A_2(A_1\boldsymbol{x}) = (A_2A_1)\boldsymbol{x}.$$

Thus several linear transformations can be replaced by one linear transformation.

To build a genuinely nonlinear model, we need a nonlinear operation between the linear transformations.

$$\boxed{\text{linear transformation} + \text{nonlinearity} = \text{more expressive model.}}$$

The sigmoid function

A classical activation function is

$$\boxed{\sigma(z) = \frac{1}{1+e^{-z}}.}$$

It satisfies

$$0 < \sigma(z) < 1$$

and

$$\sigma(0) = \frac{1}{2}.$$

Furthermore,

$$\lim_{z \to -\infty} \sigma(z) = 0,$$

$$\lim_{z \to +\infty} \sigma(z) = 1.$$

Thus

$$\boxed{\sigma : \mathbb{R} \to (0,1).}$$

This makes the sigmoid useful when an output is interpreted as a probability.

The ReLU function

Another common activation function is

$$\boxed{\text{ReLU}(z) = \max(0, z).}$$

Equivalently,

$$\text{ReLU}(z) = \begin{cases} 0, & z < 0, \\ z, & z \geq 0. \end{cases}$$

Properties:

- nonlinear;
- very simple to compute;
- negative inputs become zero;
- positive inputs pass through unchanged.

Thus a neuron can be written as

$$\boxed{y = \text{ReLU}\left(\boldsymbol{w}^T \boldsymbol{x} + b\right).}$$

Weights are matrix operations

The expression

$$\boldsymbol{w}^T\boldsymbol{x}$$

is a matrix operation.
Indeed,

$$\boldsymbol{w}^T$$

is a row vector and

$$\boldsymbol{x}$$

is a column vector.
Their product is a scalar:

$$\boxed{\boldsymbol{w}^T\boldsymbol{x} \in \mathbb{R}.}$$

For many observations we can collect the inputs into a matrix X:

$$X = \begin{pmatrix} - - - \boldsymbol{x}_1^T - - - \\ - - - \boldsymbol{x}_2^T - - - \\ \vdots \\ - - - \boldsymbol{x}_m^T - - - \end{pmatrix}.$$

Then the linear model becomes

$$\boxed{X\boldsymbol{w}.}$$

Chemistry: concentration versus time

Suppose a chemical reaction is monitored by measuring concentration.
Assume the simple model

$$c(t) = c_0 - kt.$$

Here

$$c_0 = \text{initial concentration},$$

and

$$k = \text{rate parameter}.$$

Suppose we obtain

t	$c(t)$
0	10.1
1	8.9
2	7.8
3	6.1

We want to estimate

$$c_0 \quad \text{and} \quad k.$$

Writing the chemistry as a matrix

The model

$$c(t) = c_0 - kt$$

can be written as

$$c(t) = \begin{pmatrix} 1 & -t \end{pmatrix} \begin{pmatrix} c_0 \\ k \end{pmatrix}.$$

For all measurements,

$$\underbrace{\begin{pmatrix} 1 & 0 \\ 1 & -1 \\ 1 & -2 \\ 1 & -3 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} c_0 \\ k \end{pmatrix}}_{\boldsymbol{x}} \approx \underbrace{\begin{pmatrix} 10.1 \\ 8.9 \\ 7.8 \\ 6.1 \end{pmatrix}}_{\boldsymbol{b}}.$$

Therefore,

$$\boxed{A\boldsymbol{x} \approx \boldsymbol{b}.}$$

Learning the chemical parameters

We solve

$$\boxed{\min_{\boldsymbol{x}} \|A\boldsymbol{x} - \boldsymbol{b}\|^2.}$$

The unknown vector is

$$\boldsymbol{x} = \begin{pmatrix} c_0 \\ k \end{pmatrix}.$$

Thus least squares finds the parameters that give the best fit to the measurements.

In words,

$$\boxed{\text{measurements} \longrightarrow \text{best parameters.}}$$

This is already a basic form of

$$\boxed{\text{learning from data.}}$$

From chemistry to a data matrix

Now suppose every experiment produces several measurements:

$$\boldsymbol{x} = \begin{pmatrix} \text{temperature} \\ \text{pressure} \\ \text{concentration} \\ \text{pH} \\ \vdots \end{pmatrix}.$$

Each experiment gives one vector $\boldsymbol{x}_i$.
Many experiments give a matrix

$$X = \begin{pmatrix} - - - \boldsymbol{x}_1^T - - - \\ - - - \boldsymbol{x}_2^T - - - \\ \vdots \\ - - - \boldsymbol{x}_m^T - - - \end{pmatrix}.$$

Thus

$$\boxed{\text{many experiments} \longrightarrow \text{matrix of data.}}$$

A simple predictive model

Suppose we want to predict a chemical property y.
A linear model is

$$y \approx w_1 x_1 + \cdots + w_n x_n + b.$$

In vector notation,

$$\boxed{y \approx \boldsymbol{w}^T \boldsymbol{x} + b.}$$

For many experiments,

$$\boxed{\boldsymbol{y} \approx X\boldsymbol{w}.}$$

We can learn the weights by minimizing

$$\boxed{\min_{\boldsymbol{w}} \|X\boldsymbol{w} - \boldsymbol{y}\|^2.}$$

This is exactly the least-squares principle.

Least squares as learning

Compare the two problems.

Chemistry:

$$A\boldsymbol{x} \approx \boldsymbol{b}$$

and find the best parameters $\boldsymbol{x}$.

AI:

$$X\boldsymbol{w} \approx \boldsymbol{y}$$

and find the best weights $\boldsymbol{w}$.

Mathematically:

$$\boxed{\text{minimize prediction error.}}$$

So a simple linear AI model is closely related to the same mathematics used for fitting experimental data.

Where does the neural network enter?

A linear model gives

$$y = \boldsymbol{w}^T \boldsymbol{x} + b.$$

A neuron adds a nonlinear activation:

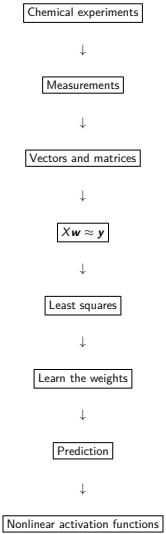
$$\boxed{y = \sigma\left(\boldsymbol{w}^T \boldsymbol{x} + b\right).}$$

A neural network combines many such operations:

$$\boldsymbol{x} \longrightarrow W_1 \boldsymbol{x} \longrightarrow \sigma \longrightarrow W_2 \longrightarrow \sigma \longrightarrow \cdots$$

The matrices contain the weights.
The activation functions provide the nonlinearity.

The complete picture



The big idea

Matrices organize information.

Least squares learns parameters from data.

Weights determine how features contribute.

Activation functions introduce nonlinearity.

Together, these ideas form basic building blocks of AI.

The central mathematical pattern is

$$\boxed{\text{data} \longrightarrow \text{model} \longrightarrow \text{parameters} \longrightarrow \text{prediction.}}$$

One final question

Suppose

$$X\boldsymbol{w} \approx \boldsymbol{y}.$$

What do the three objects represent?

$\boxed{X}$ $\boxed{\boldsymbol{w}}$ $\boxed{\boldsymbol{y}}$

- X: measured input data or features;
- $\boldsymbol{w}$: parameters or weights;
- $\boldsymbol{y}$: measured target values.

Least squares asks:

Which weights make the predictions as close as possible to the observations?