

Vector

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

In physics, vector has a direction and magnitude, like velocity of a moving object.

In data analysis, a series of discrete data can be expressed as a vector (let's say there are m data points, then you have a m -dimensional vector)

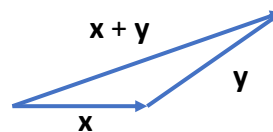
In computational model, model state can be expressed as a vector (e.g. chemical concentrations in box models, dynamic height in geophys fluid dynamics, with m being the number of grid points)

1

Vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

When there are more than two vectors, they can add together to form another vector



2

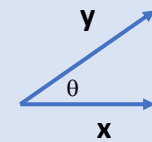
Vector

Vectors can multiply together: **inner product produces a scalar**
(remember “row times column”)

$$\mathbf{x} \cdot \mathbf{y} = \mathbf{x}^T \mathbf{y} = \begin{bmatrix} x_1 & x_2 & \cdots & x_m \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$$= x_1 y_1 + x_2 y_2 + \cdots + x_m y_m$$

In a 2D example,



$$\mathbf{x} \cdot \mathbf{y} = |\mathbf{x}| |\mathbf{y}| \cos \theta$$

3

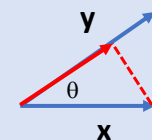
Vector

Vectors can multiply together: **inner product produces a scalar**
(remember “row times column”)

The projection of \mathbf{x} onto \mathbf{y} is:

$$|\mathbf{x}| \cos \theta = \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{y}|}$$

In a 2D example,



$$\mathbf{x} \cdot \mathbf{y} = |\mathbf{x}| |\mathbf{y}| \cos \theta$$

4

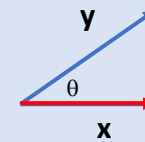
Vector

Vectors can multiply together: **inner product produces a scalar**
(remember “row times column”)

The projection of \mathbf{y} onto \mathbf{x} is:

$$|\mathbf{y}| \cos \theta = \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{x}|}$$

In a 2D example,



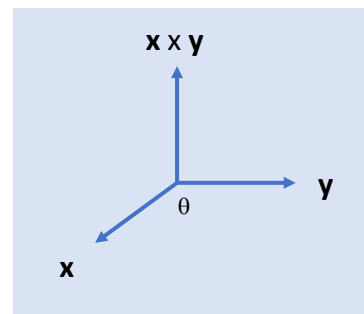
$$\mathbf{x} \cdot \mathbf{y} = |\mathbf{x}| |\mathbf{y}| \cos \theta$$

5

Vector

Vectors can multiply in a different way: **cross product produces a vector**
(remember “the right hand rule”)

$$\mathbf{x} \times \mathbf{y} = |\mathbf{x}| |\mathbf{y}| \sin \theta \hat{\mathbf{z}}$$



6

Vector

Vectors can multiply together in another different ways: **outer product produces a matrix (again, remember “row times column”)**

$$\mathbf{xy}^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \begin{bmatrix} y_1 & y_2 & \cdots & y_m \end{bmatrix} = \begin{bmatrix} x_1 y_1 & x_1 y_2 & \cdots & x_1 y_m \\ x_2 y_1 & x_2 y_2 & \cdots & x_2 y_m \\ & & \vdots & \\ x_m y_1 & x_m y_2 & \cdots & x_m y_m \end{bmatrix}$$

7

Inner product and (co)variance

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

In data analysis, vector can contain a collection of measurements (e.g. mean August temperature of Atlanta from last 140 years)

8

Inner product and (co)variance

$$\mathbf{x}' = \begin{bmatrix} x_1 - \bar{x} \\ x_2 - \bar{x} \\ \vdots \\ x_m - \bar{x} \end{bmatrix} = \begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_m \end{bmatrix}$$

Let's subtract mean of \mathbf{x} from all elements, so \mathbf{x} becomes \mathbf{x}'

9

Inner product and (co)variance

$$\mathbf{x}' \cdot \mathbf{x}' = |\mathbf{x}'|^2 = \sum x_n'^2 = (N - 1)\sigma^2 = (N - 1)\sigma_{xx}$$

Inner product of \mathbf{x}' with itself the square of the **"length" of the \mathbf{x}' vector.**

It is equal to the **variance (σ^2 or σ_{xx})** of the data contained in vector \mathbf{x} multiplied by N-1

The "length" $|\mathbf{x}'|$ is also called the **L2 norm**

10

Inner product and (co)variance

$$\mathbf{x}' \cdot \mathbf{y}' = \sum x'_n y'_n = (N - 1)\sigma_{xy}$$

Inner product of \mathbf{x}' and \mathbf{y}' is:

- (1) the **covariance** of the data contained in vector \mathbf{x} and \mathbf{y} (multiplied by N-1)
- (2) the **projection** of \mathbf{x} onto \mathbf{y} (multiplied by $|\mathbf{y}'|$)
- (3) Proportional to the **correlation** between \mathbf{x} and \mathbf{y}

11

Inner product and (co)variance

$$\mathbf{x}' \cdot \mathbf{y}' = \sum x'_n y'_n = (N - 1)\sigma_{xy}$$

Inner product of \mathbf{x}' and \mathbf{y}' is:

- (1) the **covariance** of the data contained in vector \mathbf{x} and \mathbf{y} (multiplied by N-1)
- (2) the **projection** of \mathbf{x} onto \mathbf{y} (multiplied by $|\mathbf{y}'|$)

12

Matrix-vector product and $\mathbf{Ax} = \mathbf{b}$

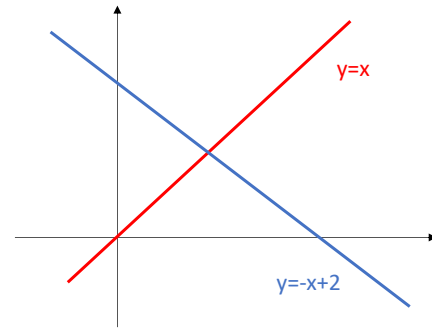
A (2 x 2) example

$$\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

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

The “row” point of view: $x - y = 0$

Intersection of two lines $x + y = 2$



The problem $\mathbf{Ax} = \mathbf{b}$ is solvable only if the two lines intersect at a point

13

Matrix-vector product and $\mathbf{Ax} = \mathbf{b}$

Inverse matrix \mathbf{A}^{-1} $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$

If inverse matrix of \mathbf{A} exists, we have

In MATLAB it is **inv** function. Also the backslash operation “ $\mathbf{A} \backslash \mathbf{b}$ ” does it for you...

In Python, **np.linalg.inv** (or **scipy.linalg.inv**)

```
% create a random matrix A and a vector b
A=rand(2,2);
b=rand(2,1);
% solve A^-1 b
x=inv(A)*b
```

```
x = 2x1
    -1.1415
     1.9246
```

```
x=A\b
```

```
x = 2x1
    -1.1415
     1.9246
```

14

Matrix-vector product and $\mathbf{Ax} = \mathbf{b}$

What if \mathbf{A} is not a square matrix?

Let's say there are too many (m) equations than (n) unknowns.

In this case, there is no unique solution \mathbf{x} that can satisfy all equations in $\mathbf{Ax} = \mathbf{b}$ in the exact sense

15

Matrix-vector product and $\mathbf{Ax} = \mathbf{b}$

Consider \mathbf{A} is ($m \times 2$) matrix, \mathbf{x} is (2×1) and \mathbf{b} is ($m \times 1$) and m is much larger than 2.

In this case, \mathbf{Ax} is unlikely equal to \mathbf{b} in the exact sense.

However, we could look for the approximate solution \mathbf{x} that minimizes the magnitude of the misfit $|\mathbf{Ax} - \mathbf{b}|$ (aka least square method).

16

Matrix-vector product and $\mathbf{Ax} = \mathbf{b}$

In MATLAB, "backslash"(\) takes care of this problem.

It is also called pseudoinverse problem, and "**pinv**" is the MATLAB function.

In python, it is available as **numpy.linalg.pinv** or **scipy.linalg.pinv**

```
A=rand(20,2);
b=rand(20,1);
% solve A^-1 b
x=A\b
```

```
x = 2x1
    0.2472
    0.8127
```

```
x=pinv(A)*b
```

```
x = 2x1
    0.2472
    0.8127
```

```
x=inv(A)*b
```

```
Error using inv
Matrix must be square.
```