Homework 4

posted on Brightspace

due next Mon (Oct 3) at 11:59pm*

(pushed back so I'm not lecturing about topics for a homework due two days later)

(due late on Mon so people can ask last-minute questions that Mon after class)

`Homework4.pdf` (written description)

`Homework4.ipynb` (notebook to use for your solution)

`difdata.csv`

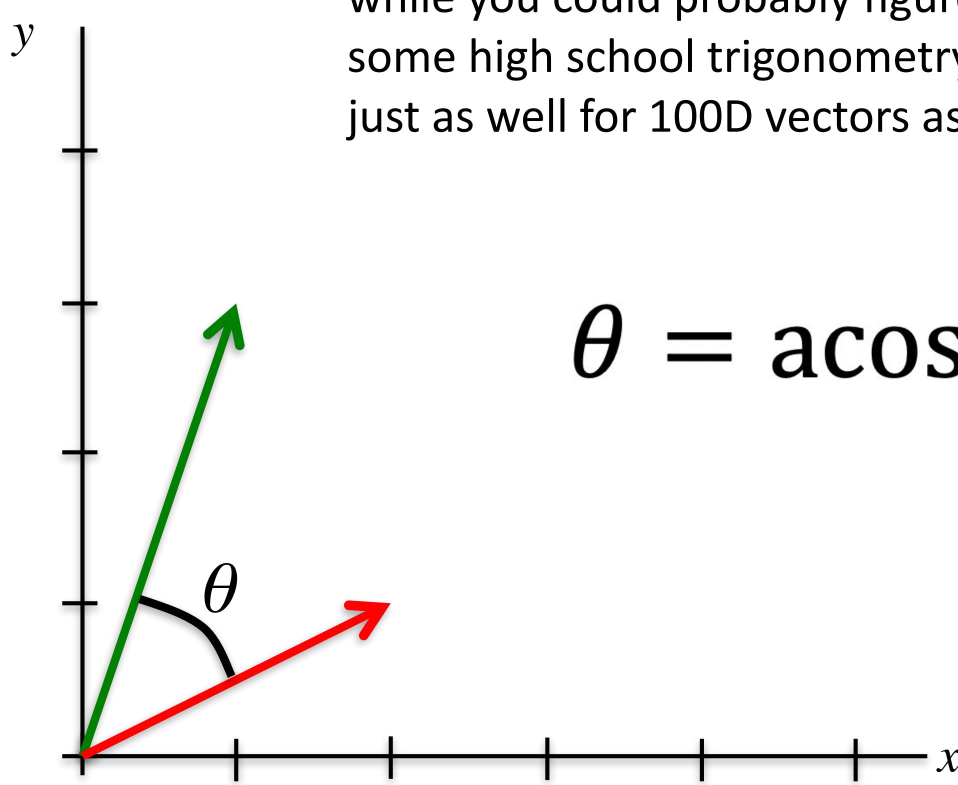
* most homeworks will continue to be due at class time

download from Brightspace

`VectorsMatricesLinearAlgebra.ipynb`
`ControlFlow.ipynb`

angle between two vectors

```
an = np.linalg.norm(a)
bn = np.linalg.norm(b)
theta = math.acos(np.dot(a,b)/(an*bn))
print(np.rad2deg(theta))
```

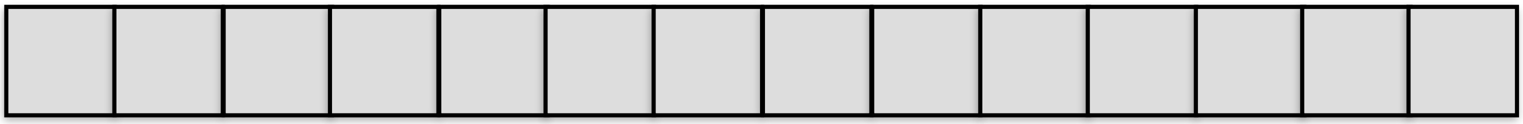


while you could probably figure this angle out using some high school trigonometry, this method works just as well for 100D vectors as for 2D vectors

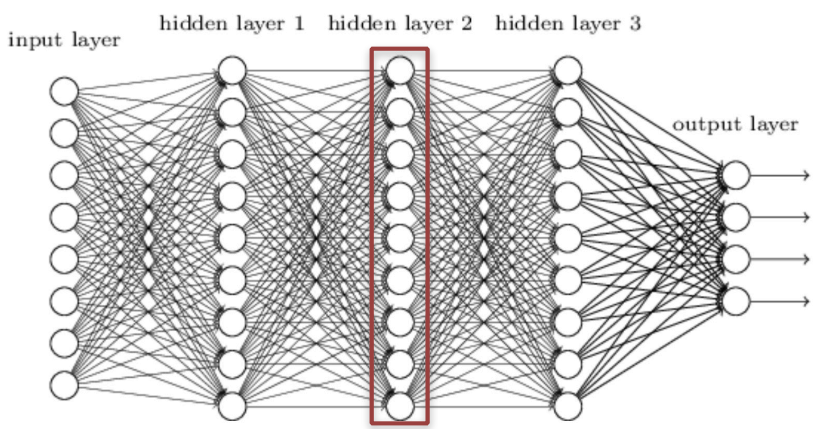
$$\theta = \operatorname{acos}\left(\frac{\boldsymbol{a}\cdot\boldsymbol{b}}{\|\boldsymbol{a}\|\|\boldsymbol{b}\|}\right)$$

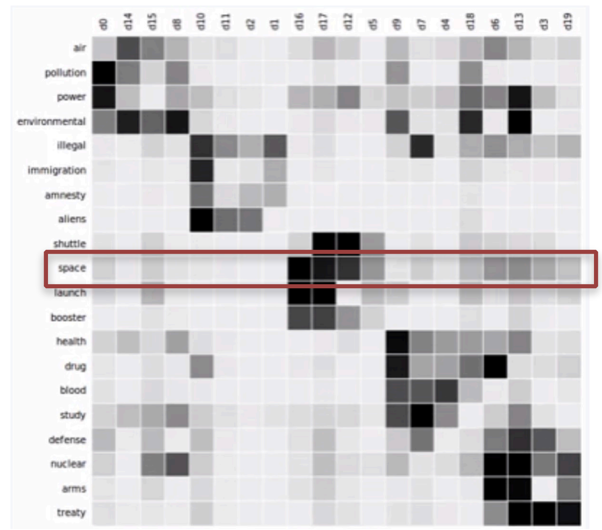
$$\theta = \operatorname{acos}\left(\frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\|\boldsymbol{a}\|\,\|\boldsymbol{b}\|}\right)$$

cosine angle is sometimes used to measure the similarity between two vector representations



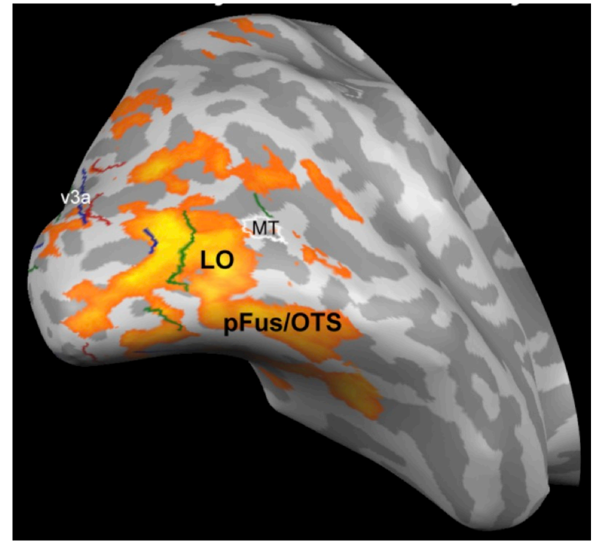
- some number of psychological measures
- or psychological, social, and demographic measures
- or some number of neural measures



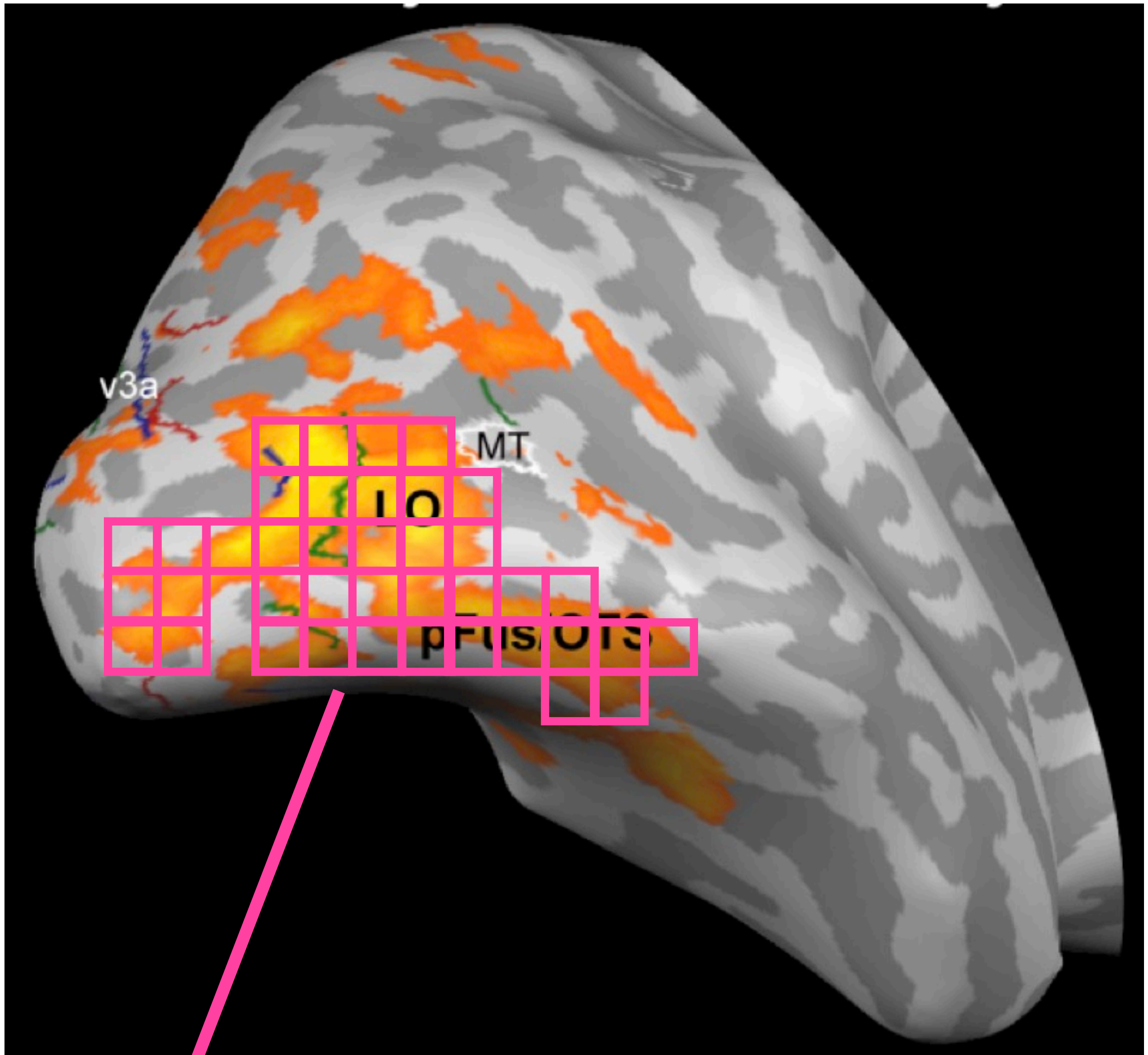


- activity of a units in a layer of a neural network model
- semantic representation of a word from latent semantic analysis

e.g., computing similarity in fMRI activation for objects



e.g., computing similarity in fMRI activation for objects

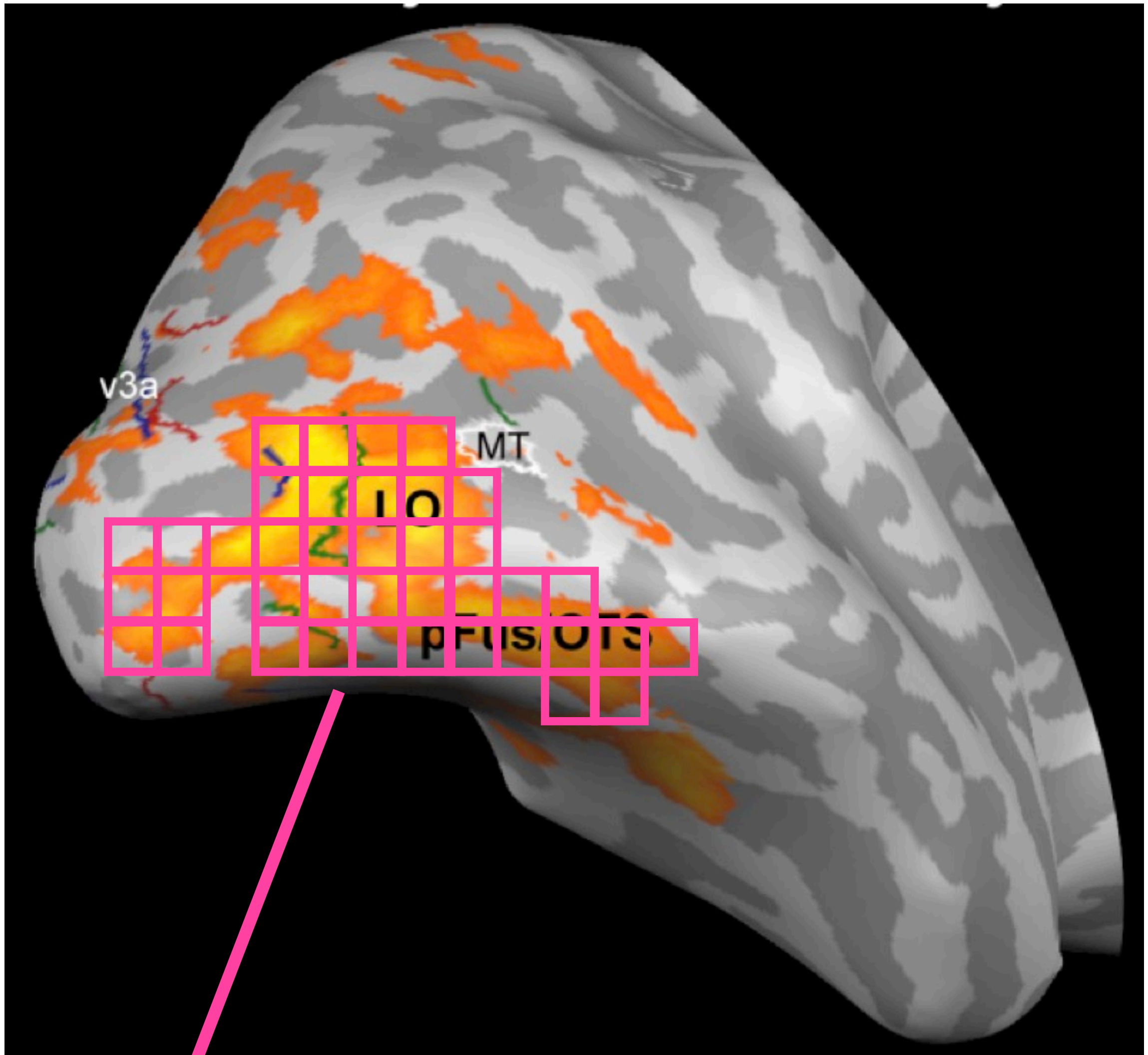


.23	.49	.12	.72	.11	. . .	.43	.87

.18	.39	.22	.83	.09	. . .	.21	.76

similarity of (voxel) representations?

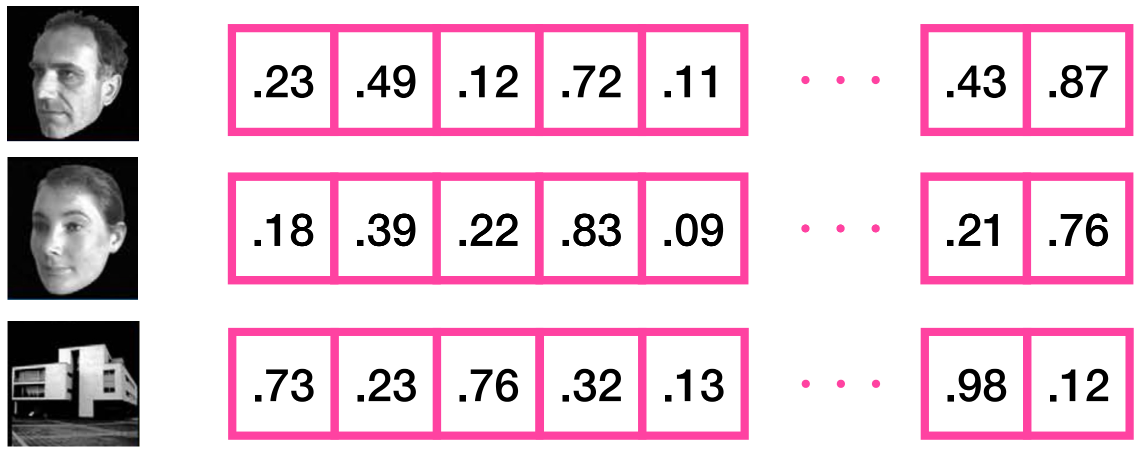
e.g., computing similarity in fMRI activation for objects



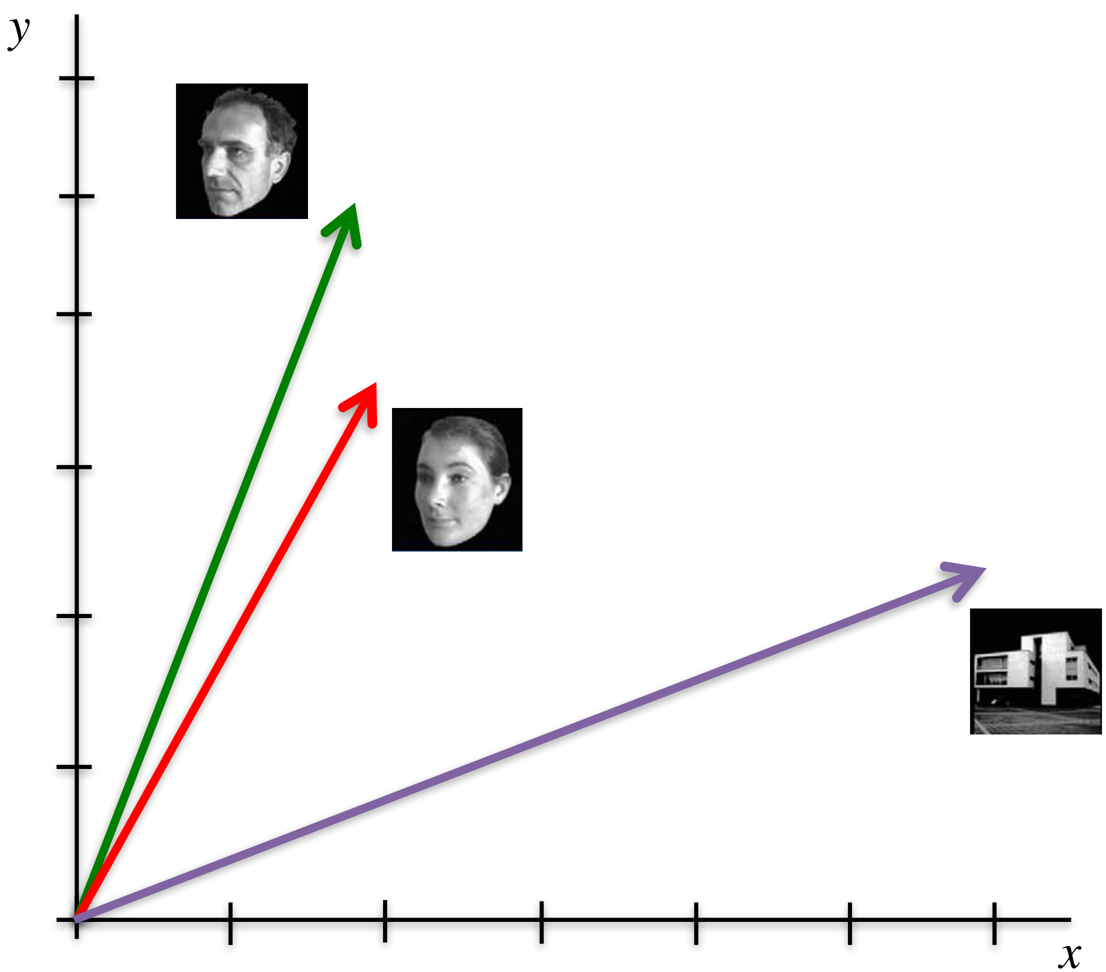
.23	.49	.12	.72	.11	. . .	.43	.87

.73	.23	.76	.32	.13	. . .	.98	.12

similarity of (voxel) representations?



similarity of (voxel) representations?

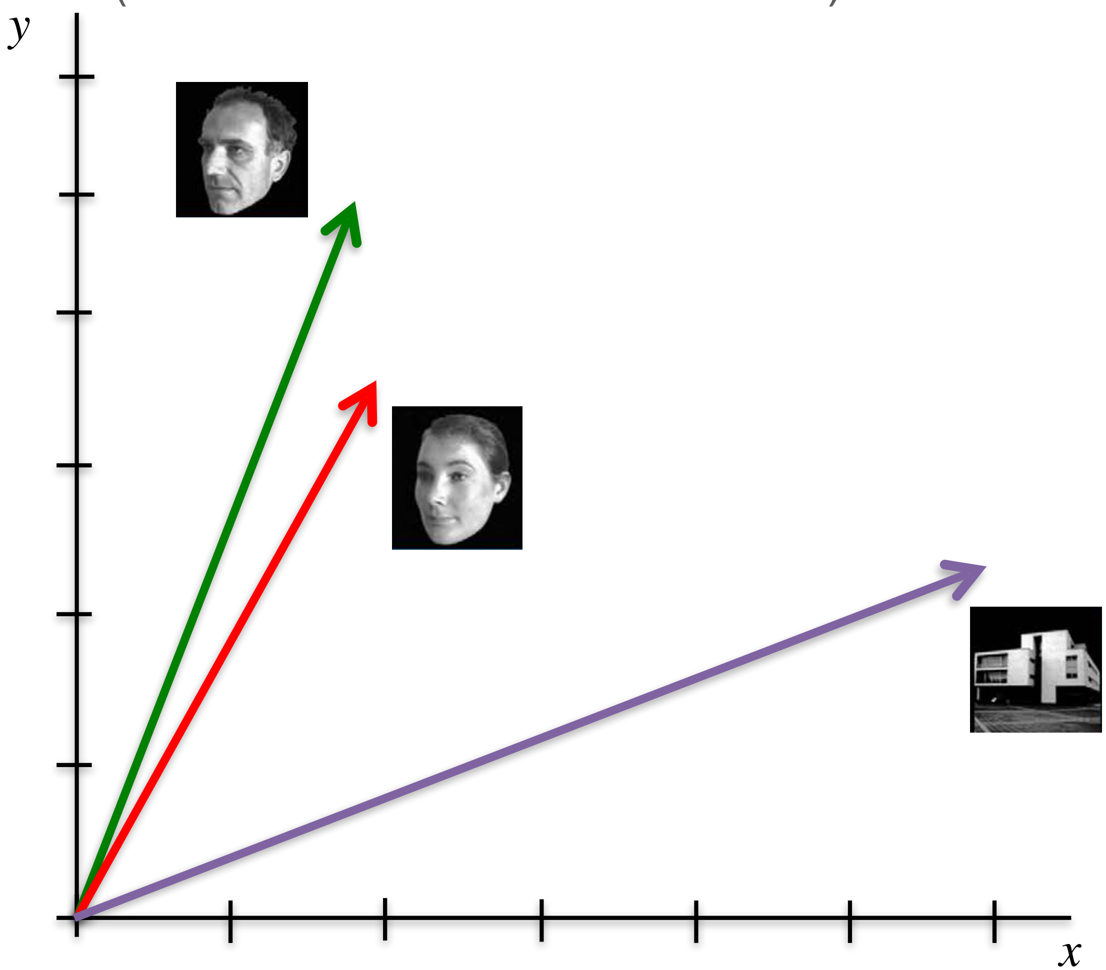


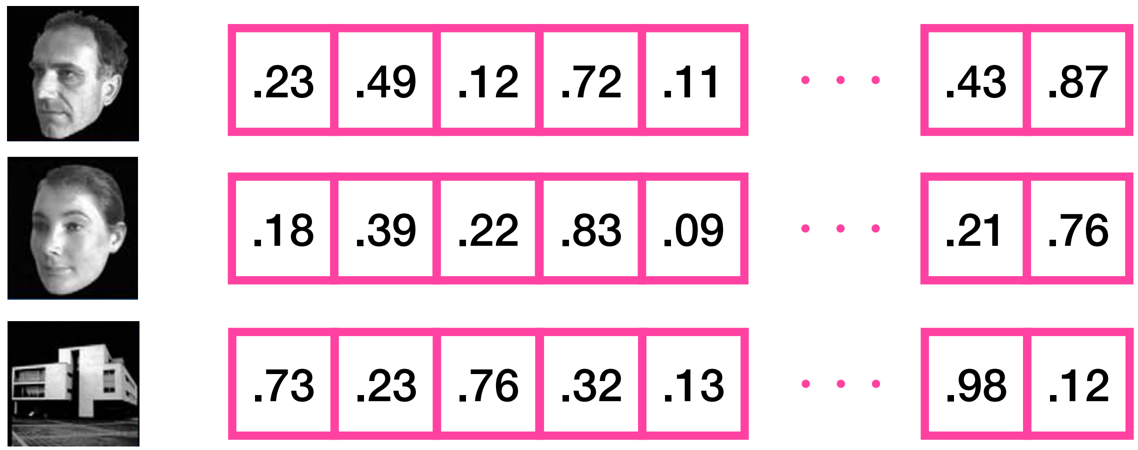
could be in a 40D space
(here only showing 2D)

.23	.49	.12	.72	.11	· · ·	.43	.87
.18	.39	.22	.83	.09	· · ·	.21	.76
.73	.23	.76	.32	.13	· · ·	.98	.12

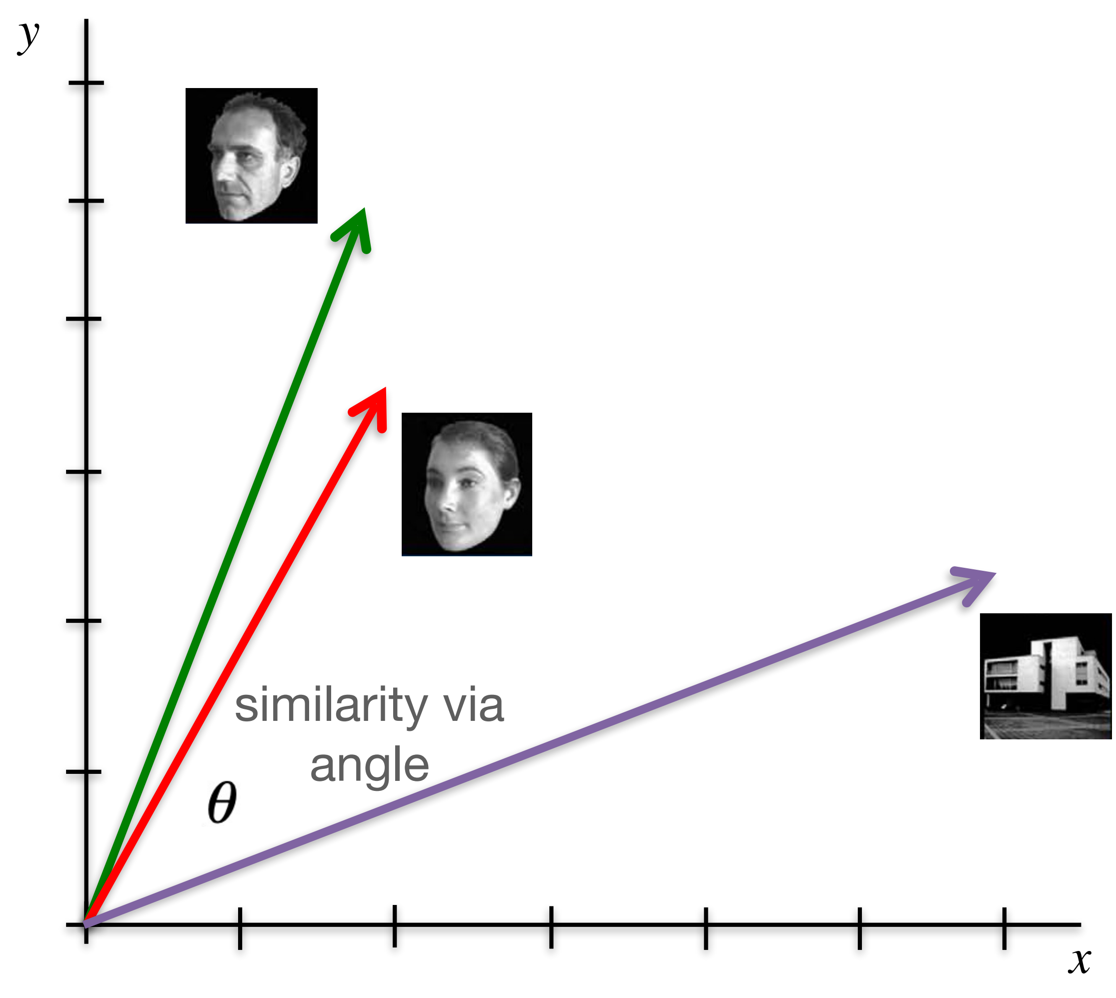
similarity of (voxel) representations?

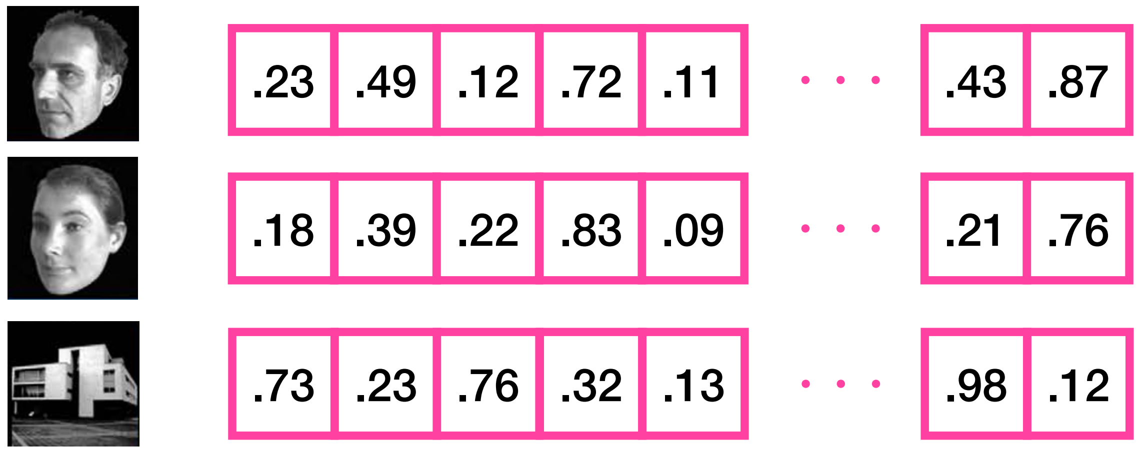
similarity via Pearson correlation
(cos θ of vectors mean-centered)



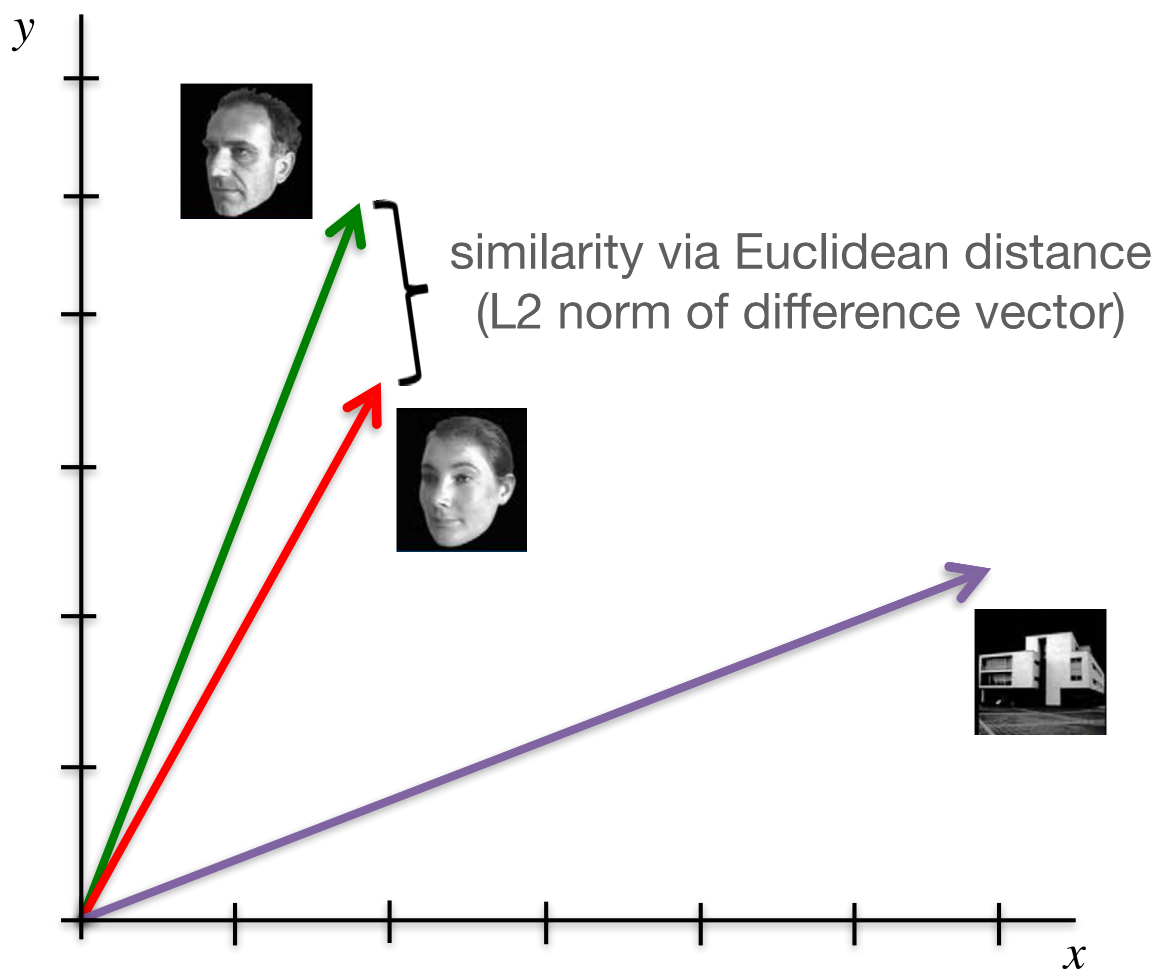


similarity of (voxel) representations?



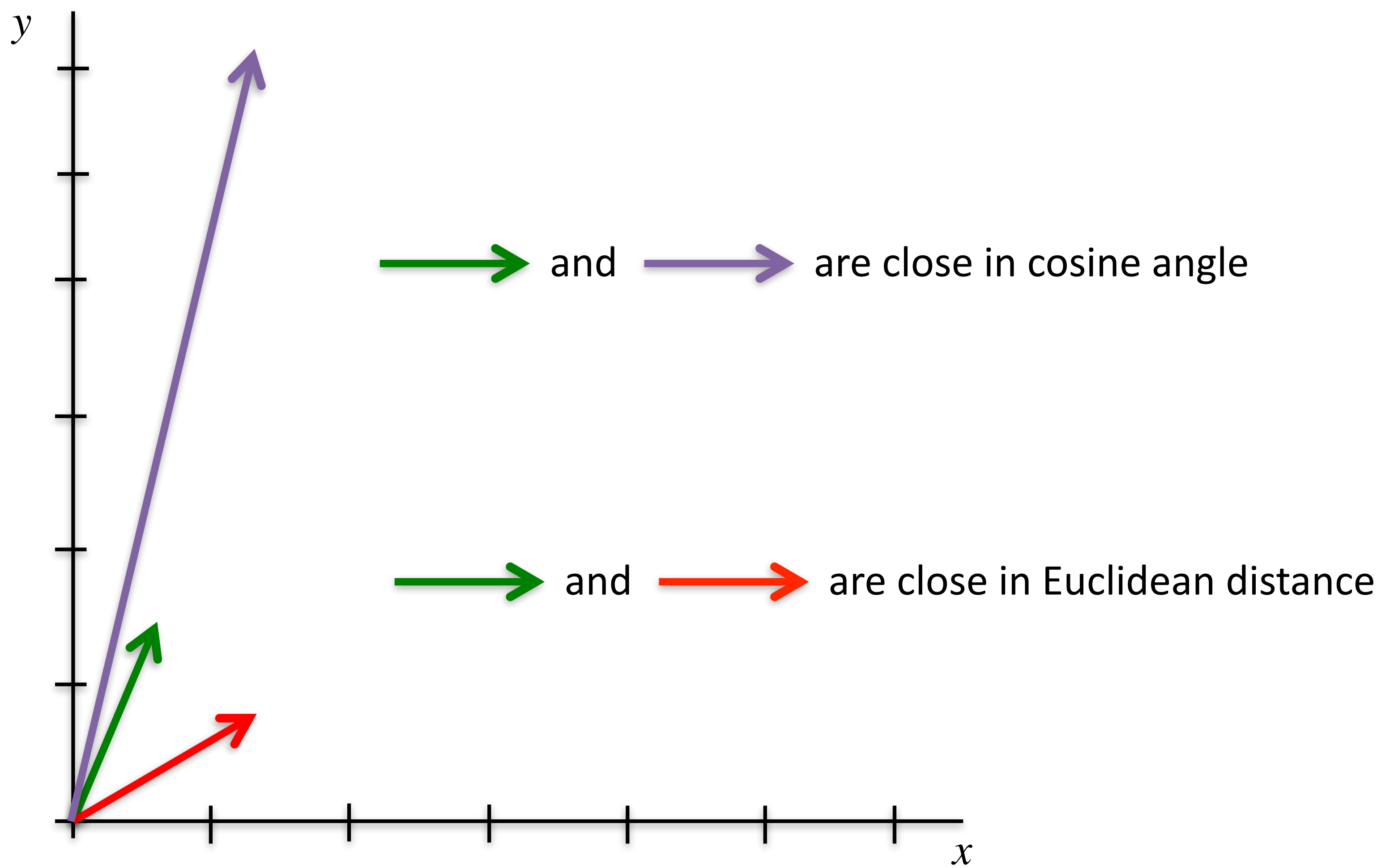


similarity of (voxel) representations?

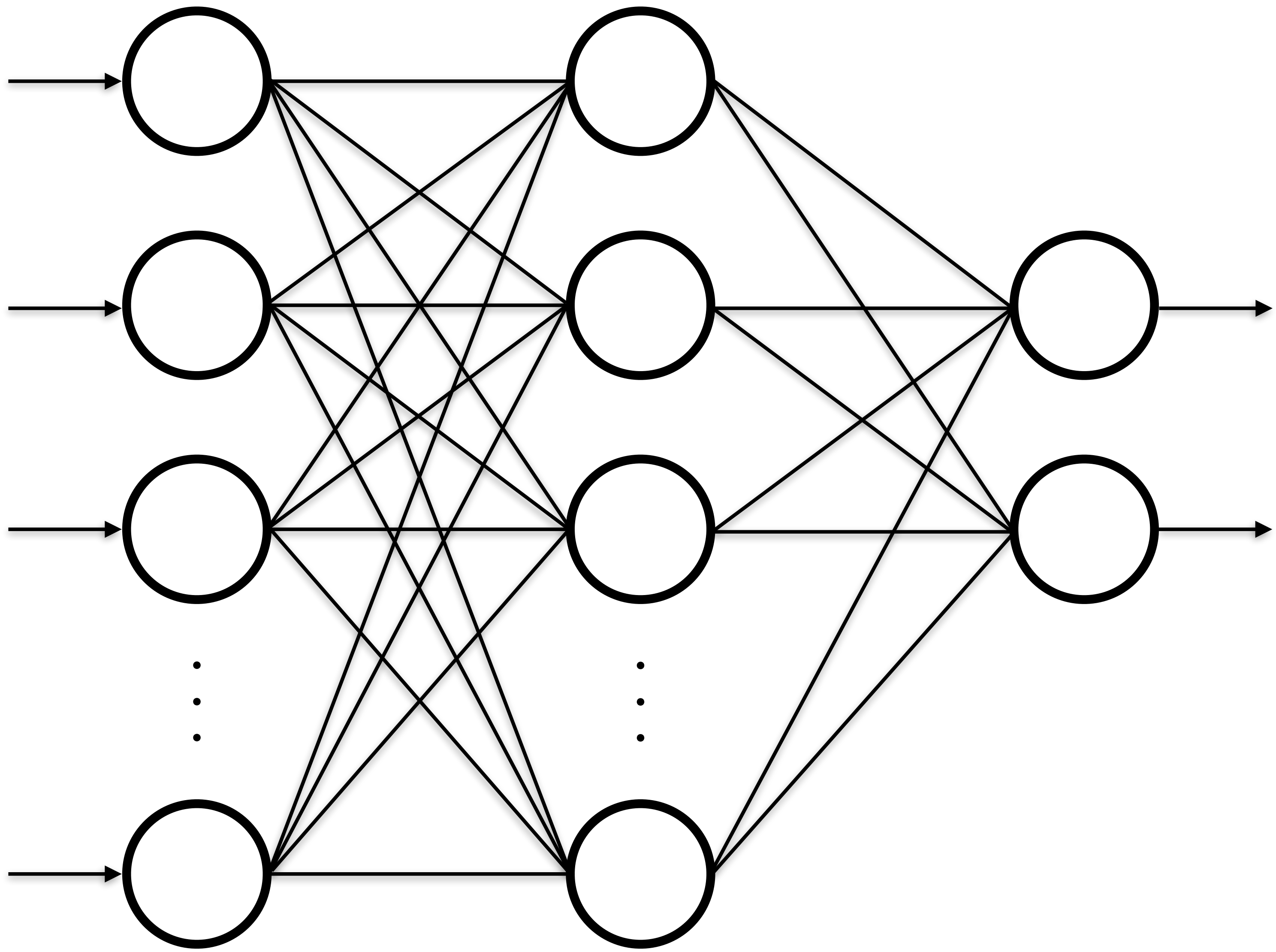


cosine angle vs. Euclidean distance

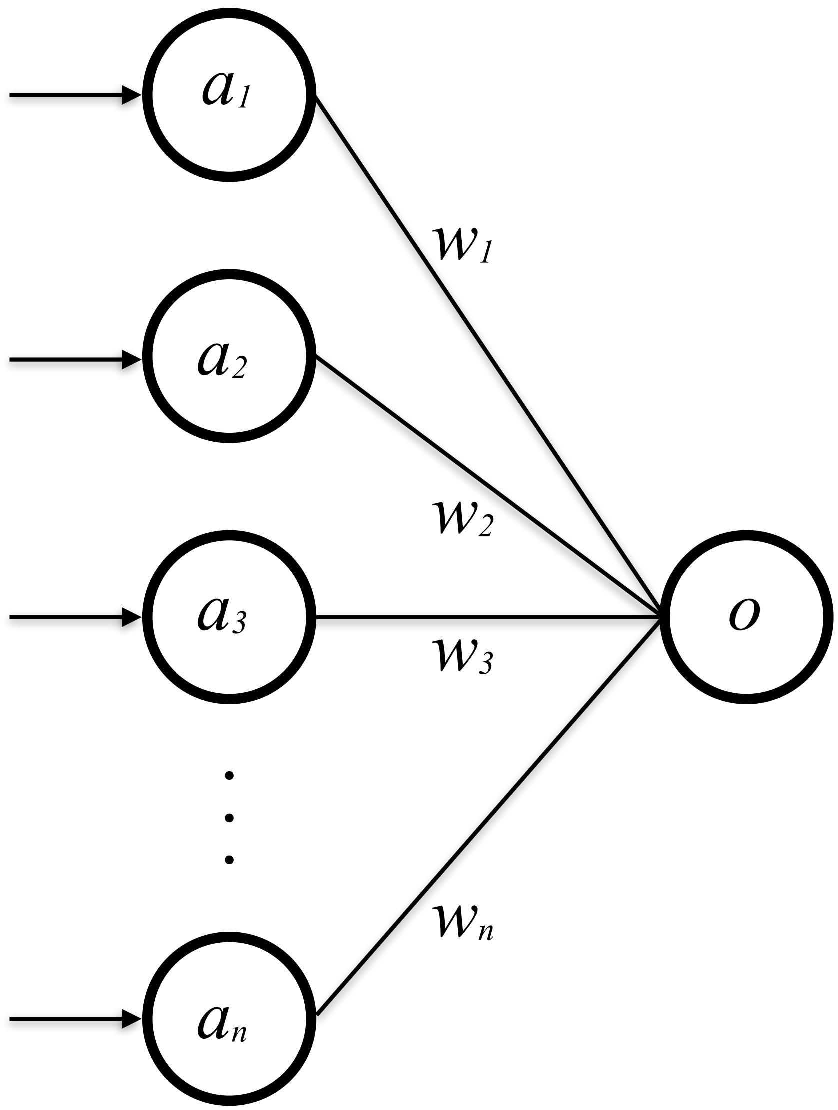
depends on what the vectors (and their distances) mean



math of a simple neural network model

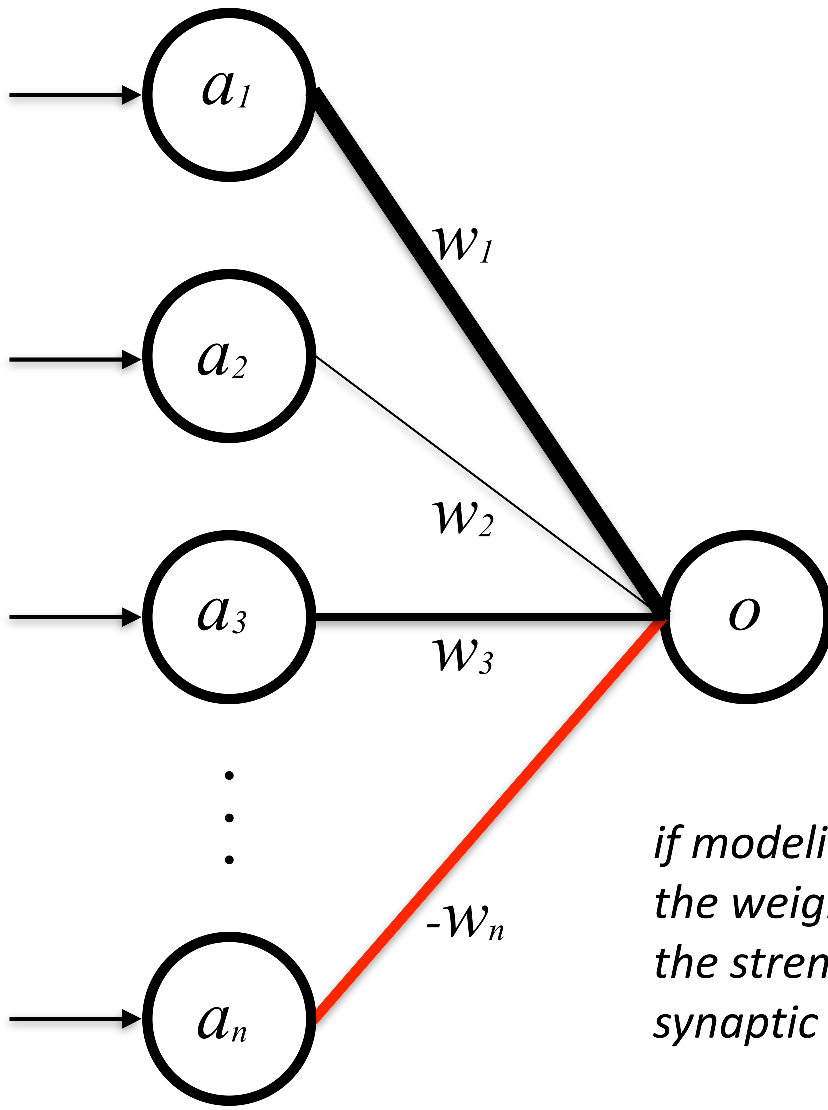


math of a simple neural network model



$$o = f\left(\sum_{i=1}^{n} a_i w_i\right)$$

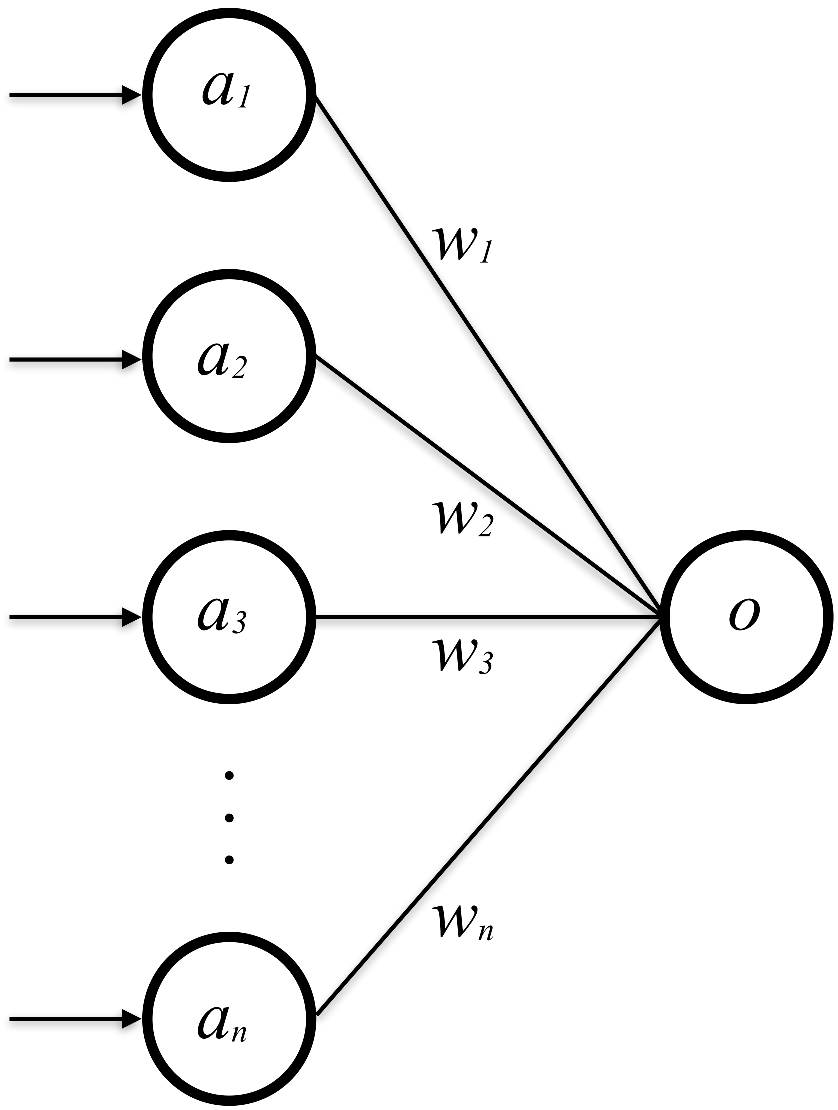
math of a simple neural network model



$$o = f\left(\sum_{i=1}^{n} a_i w_i\right)$$

if modeling actual neurons, the weights would reflect the strength (and type) of synaptic connections

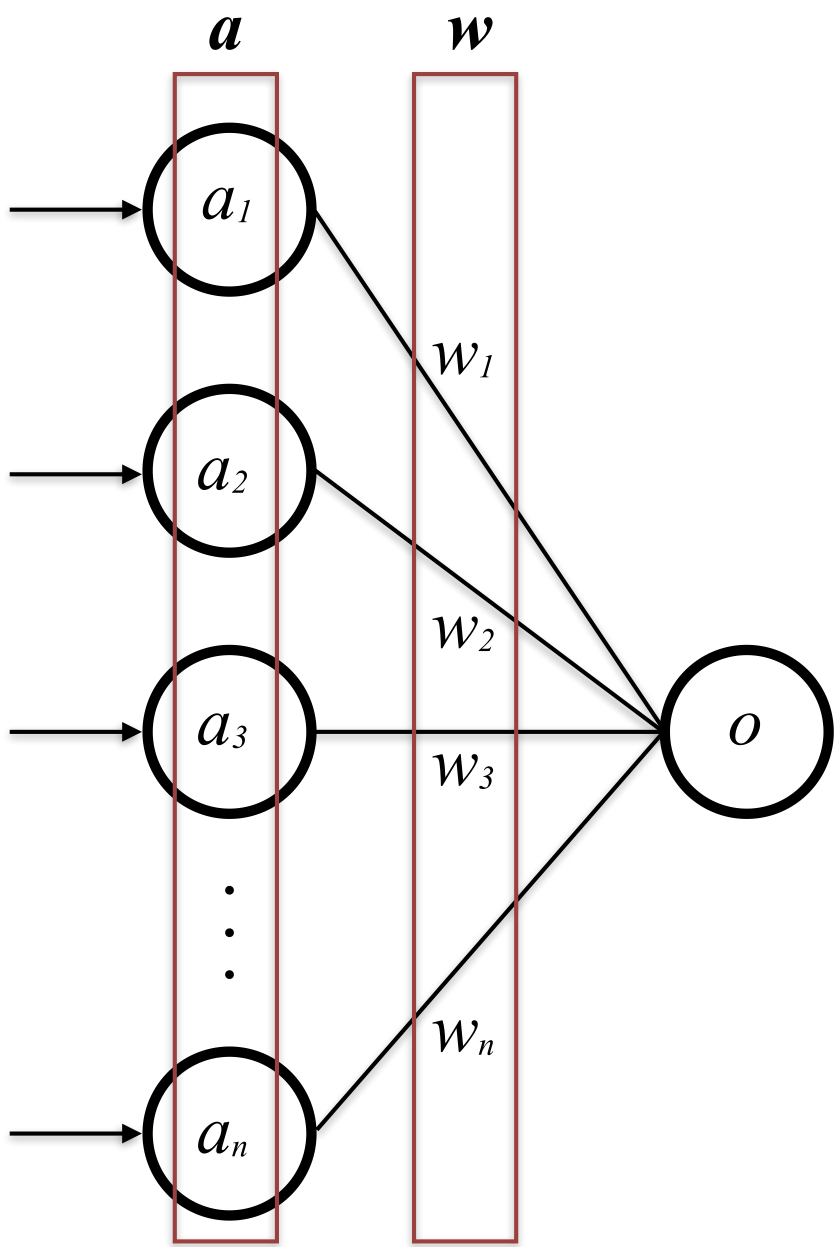
math of a simple neural network model



$$o = \sum_{i=1}^{n} a_i w_i$$

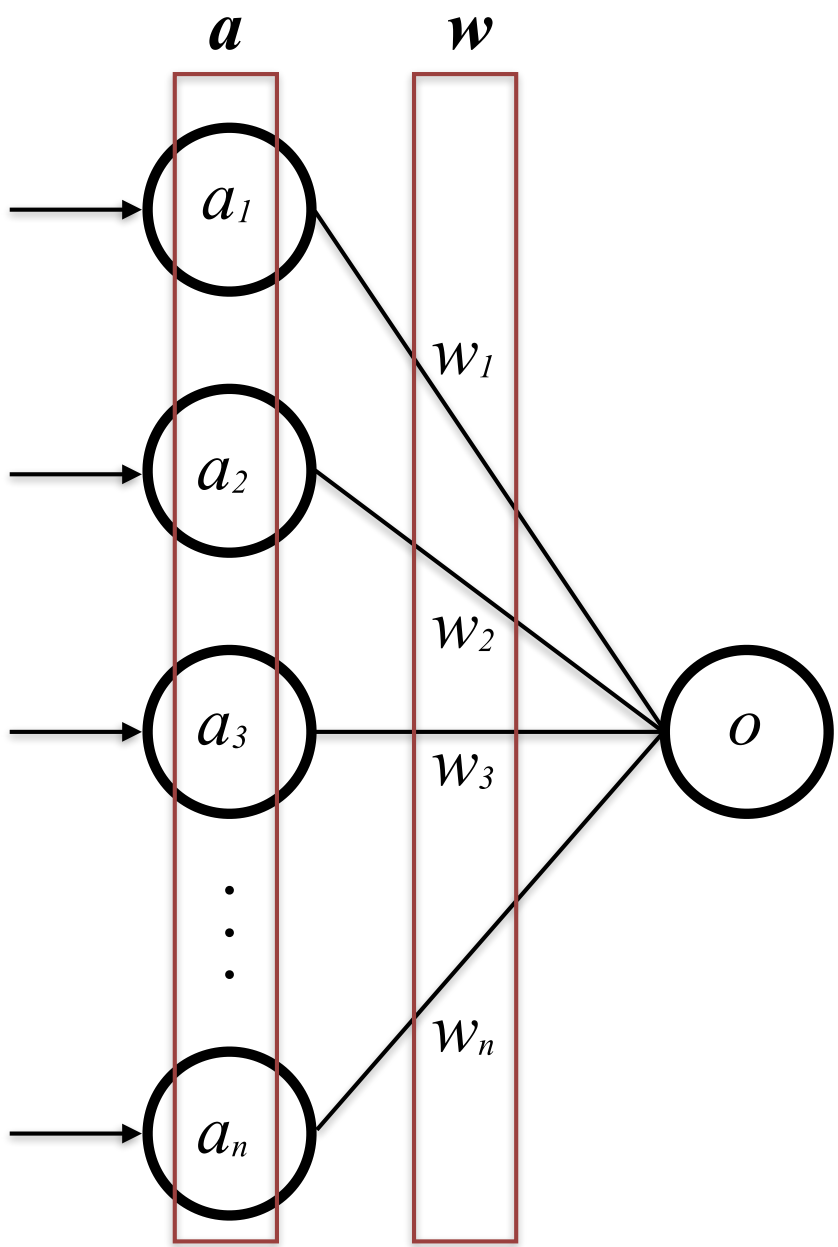
linear model (most interesting neural networks are nonlinear)

math of a simple neural network model



$$\boldsymbol{o} = \sum_{i=1}^{n} a_i w_i = \boldsymbol{a} \cdot \boldsymbol{w}$$

math of a simple neural network model



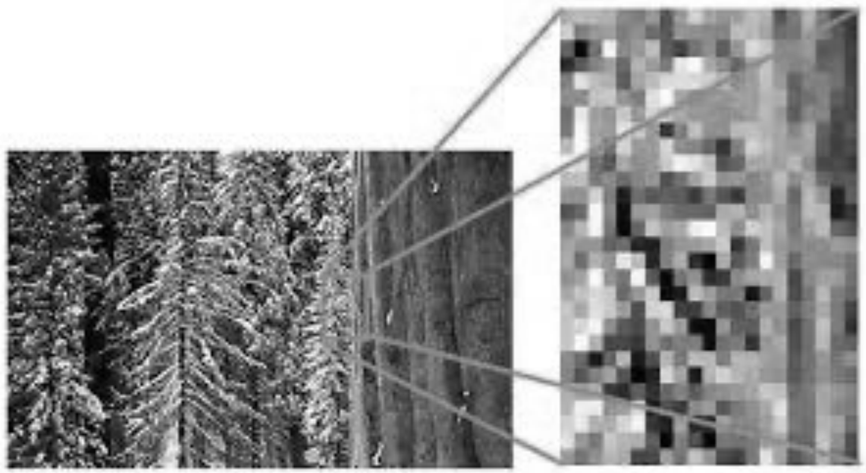
$$o = \sum_{i=1}^{n} a_i w_i = \boldsymbol{a} \cdot \boldsymbol{w}$$

*if the activations and weights were normalized, o would then be the cosine angle between **w** and **a***

neural networks are pattern recognizers

Matrices

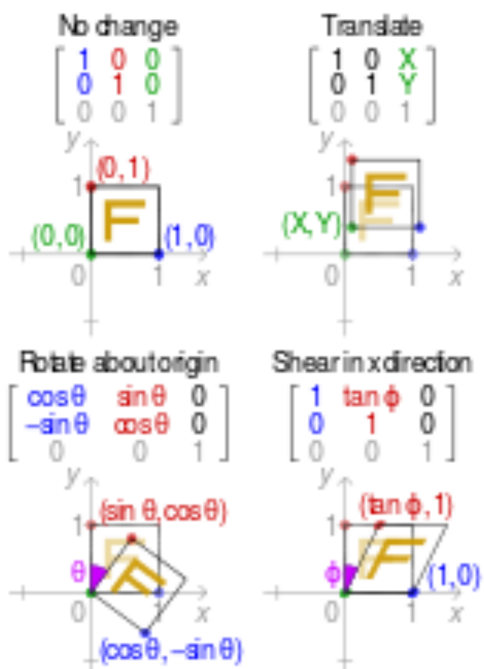
matrices



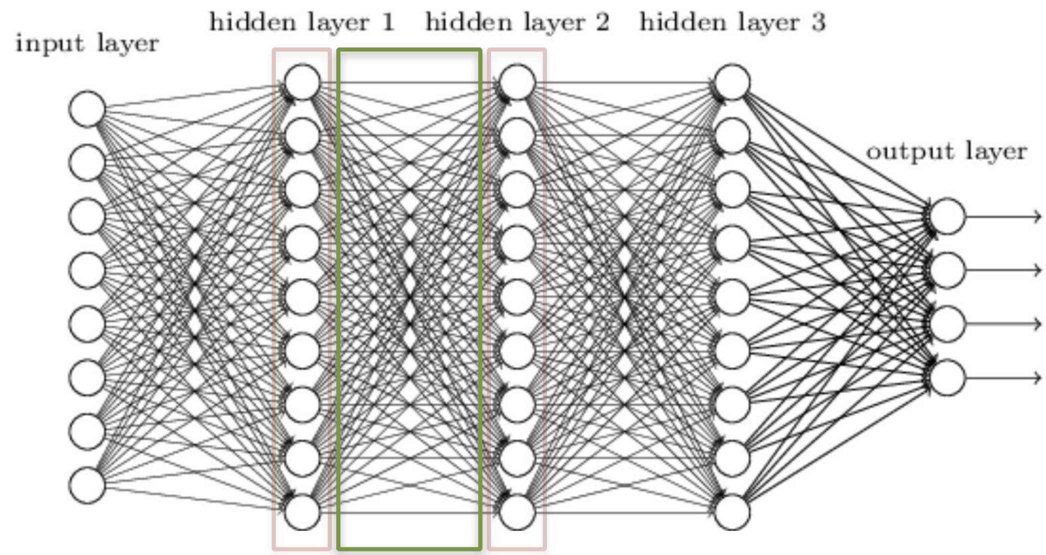
pixels of an image

$$
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix}
=
\begin{bmatrix} 1 & w_1 & x_1 \\ 1 & w_2 & x_2 \\ 1 & w_3 & x_3 \\ 1 & w_4 & x_4 \\ 1 & w_5 & x_5 \\ 1 & w_6 & x_6 \\ 1 & w_7 & x_7 \end{bmatrix}
\begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix}
+
\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \\ \varepsilon_7 \end{bmatrix}
$$

design of a statistical model



transformation of a vector



full set of weights between
layers of a neural network model

matrix addition, subtraction, scalar multiplication

same as element-wise addition, subtraction, and scalar multiplication with numpy arrays

mathematical operations on matrices

$$\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$$

$$c(\mathbf{A}+\mathbf{B}) = c\mathbf{A} + c\mathbf{B}$$

Convince yourself in Python:

```
A = np.array([[2, 1], [2, 3]])
B = np.array([[1, 2], [0, 1]])
c = 2
print((A + B) - (B + A))
print(c*(A+B) - (c*A + c*B))
```

solving systems of linear equations

$$
\begin{aligned}
x + 2y &= 2 \\
x + y &= 3
\end{aligned}
$$

How would you view these equations in Python?

solving systems of linear equations

```
y = 1 - .5x
y = 3 - x
```

How would you view these equations in Python?

solving systems of linear equations

```
y = 1 - .5x
y = 3 - x
```

How would you view these equations in Python?

```
x = np.arange(-6, 6, .01)
y1 = 1 - .5*x
y2 = 3 - x
plt.plot(x, y1, x, y2)
```

What's the solution to the system of equations?

solving systems of linear equations

$$
\begin{aligned}
x + 2y &= 2 \\
x + y &= 3
\end{aligned}
$$

Rewrite in matrix notation:

$$
\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}
$$

solving systems of linear equations

$$\begin{aligned} 1x + 2y &= 2 \\ 1x + 1y &= 3 \end{aligned}$$

Rewrite in matrix notation:

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

solving systems of linear equations

$$1x + 2y = 2$$
$$1x + 1y = 3$$

Rewrite in matrix notation:

$$\underset{2x2}{\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}} \underset{2x1}{\begin{bmatrix} x \\ y \end{bmatrix}} = \underset{2x1}{\begin{bmatrix} 2 \\ 3 \end{bmatrix}}$$

solving systems of linear equations

$$1x + 2y = 2$$
$$1x + 1y = 3$$

Rewrite in matrix notation:

$$\begin{bmatrix} \boxed{1 \quad 2} \\ 1 \quad 1 \end{bmatrix} \begin{bmatrix} \boxed{\begin{matrix} x \\ y \end{matrix}} \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} \boxed{x + 2y} \\ \\ \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

solving systems of linear equations

$$1x + 2y = 2$$
$$1x + 1y = 3$$

Rewrite in matrix notation:

$$\begin{bmatrix} 1 & 2 \\ \boxed{1 \quad 1} \end{bmatrix} \begin{bmatrix} \boxed{\begin{matrix} x \\ y \end{matrix}} \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} x + 2y \\ \boxed{x + y} \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times \boxed{n}$ matrix $\qquad \boxed{n} \times p$ matrix $\qquad m \times p$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$\boxed{m} \times n$ matrix $\quad n \times \boxed{p}$ matrix $\quad \boxed{m} \times \boxed{p}$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\qquad$ $n \times p$ matrix $\qquad$ $m \times p$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

multiplying matrices

$$\begin{bmatrix} \boxed{a_{11} & a_{12} & \cdots & a_{1n}} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} \boxed{c_{11}} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

dot product

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times \boxed{n}$ matrix $\quad$ $\boxed{n} \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

dot product

multiplying matrices

$$\begin{bmatrix} \boxed{a_{11} \quad a_{12} \quad \cdots \quad a_{1n}} \\ a_{21} \quad a_{22} \quad \cdots \quad a_{2n} \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ a_{m1} \quad a_{m2} \quad \cdots \quad a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & \boxed{c_{12}} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & \boxed{c_{1p}} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ \boxed{c_{21}} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$$m \times n \text{ matrix} \qquad n \times p \text{ matrix} \qquad m \times p \text{ matrix}$$

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & \boxed{c_{22}} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\qquad$ $n \times p$ matrix $\qquad$ $m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \boxed{a_{21} & a_{22} & \cdots & a_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & \boxed{c_{2p}} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad$ $n \times p$ matrix $\quad$ $m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$
\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}
\times
\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix}
=
\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}
$$

$m \times n$ matrix $\qquad$ $n \times p$ matrix $\qquad$ $m \times p$ matrix

$$
c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}
$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & \boxed{c_{m2}} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\qquad$ $n \times p$ matrix $\qquad$ $m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

multiplying matrices

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{bmatrix}$$

$m \times n$ matrix $\quad n \times p$ matrix $\quad m \times p$ matrix

$$c_{ij} = \underbrace{\sum_{k=1}^{n} a_{ik} b_{kj}}_{\mathbf{a_i} \cdot \mathbf{b_j}}$$

solving systems of linear equations

$$
\begin{aligned}
x + 2y &= 2 \\
x + y &= 3
\end{aligned}
$$

Rewrite in matrix notation:

$$
\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}
$$

$$
\begin{bmatrix} x + 2y \\ x + y \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}
$$

solving systems of linear equations

$$\begin{aligned} x_1 + 2x_2 &= 2 \\ x_1 + x_2 &= 3 \end{aligned}$$

Rewrite :

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

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

matrix *vector* *vector*

solving systems of linear equations

$$x_1 + 2x_2 = 2$$
$$x_1 + x_2 = 3$$

Rewrite :

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

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

matrix	*column vector*	*column vector*
2x2	2x1	2x1

mathematical operations on matrices

How about this?

$$\mathbf{AB} \stackrel{?}{=} \mathbf{BA}$$

mathematical operations on matrices

How about this?

$$\mathbf{AB} \stackrel{?}{=} \mathbf{BA}$$

Homework 5

Well, you know it cannot be true in general:

A $m \times n$

B $n \times p$

AB is defined

BA is not (unless $m = n = p$)

Imagine **C** is 2x3 and **D** is 3x4.

Then **CD** is defined but **DC** isn't.

mathematical operations on matrices

How about this? Try it in Python.

$$\mathbf{AB} \stackrel{?}{=} \mathbf{BA}$$

Homework 5

matrix multiplication

```
A@B - B@A
```

vs.

element-wise multiplication

```
A*B - B*A
```

solving systems of linear equations

$$
\begin{aligned}
x + 2y &= 2 \\
x + y &= 3
\end{aligned}
$$

Rewrite in matrix notation:

$$
\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}
$$

back to this

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

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

matrix *vector* *vector*

what do we need to do to solve for x?

back to this

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

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

matrix *vector* *vector*

if we can move A to the other side,
we will have a solution for **x**

mathematical operations on matrices

$$Ax = b$$

how would you solve for x if
A, x, and b were all scalar values?

mathematical operations on matrices

$$\mathbf{Ax} = \mathbf{b}$$

but these are vectors and matrices

mathematical operations on matrices

$$\mathbf{Ax = b}$$

imagine we have some matrix W such that

$$\mathbf{WAx = Wb}$$

mathematical operations on matrices

$$\mathbf{Ax} = \mathbf{b}$$

imagine we have some matrix W such that

$$\underbrace{\mathbf{WA}}\mathbf{x} = \mathbf{Wb} \qquad\qquad \mathbf{WAx} \neq \mathbf{bW}$$

$$\mathbf{1x} = \mathbf{Wb}$$

what is the matrix
equivalent of 1?

mathematical operations on matrices

$$\mathbf{Ax} = \mathbf{b}$$

imagine we have some matrix W such that

$$\mathbf{WAx} = \mathbf{Wb} \qquad \mathbf{WAx} \cancel{=} \mathbf{bW}$$

$$\mathbf{Ix} = \mathbf{Wb}$$

identify
matrix

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

mathematical operations on matrices

W is the inverse of matrix **A**, or $\mathbf{A}^{-1}$

$$\mathbf{WAx} = \mathbf{Wb}$$

$$\mathbf{A}^{-1}\mathbf{Ax} = \mathbf{A}^{-1}\mathbf{b}$$

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

mathematical operations on matrices

$\mathbf{A}^{-1}$ is not the same as a matrix of the inverse of the elements …

demonstrate they are not the same in **Homework 5**

back to this

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

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

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

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

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

back to this

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

```
A = np.array([[1, 2], [2, 1]])
b = np.array([[2],[3]])

x = np.linalg.inv(A) @ b
```

back to this

$$\begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

```
A = np.array([[1, 2], [2, 1]])
b = np.array([[2],[3]])

x = np.linalg.inv(A) @ b
```

this uses a better numerical method

```
x = np.linalg.solve(A, b)
```

larger systems of linear equations

(mathematically the same matrix operation)

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

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

solving for $\mathbf{x}$

not all matrices are invertible

- only square matrices are invertible (but not all square matrices are invertible)

- some square matrices are singular (not invertible)

  e.g., some rows or columns are linear combinations of one another (they are not linearly independent)

- matrices are singular if their determinant is zero (may be numerically close to zero on a computer)

solving systems of linear equations

$$\begin{aligned} 2x - y &= 2 \\ 6x - 3y &= 12 \end{aligned}$$

solving systems of linear equations

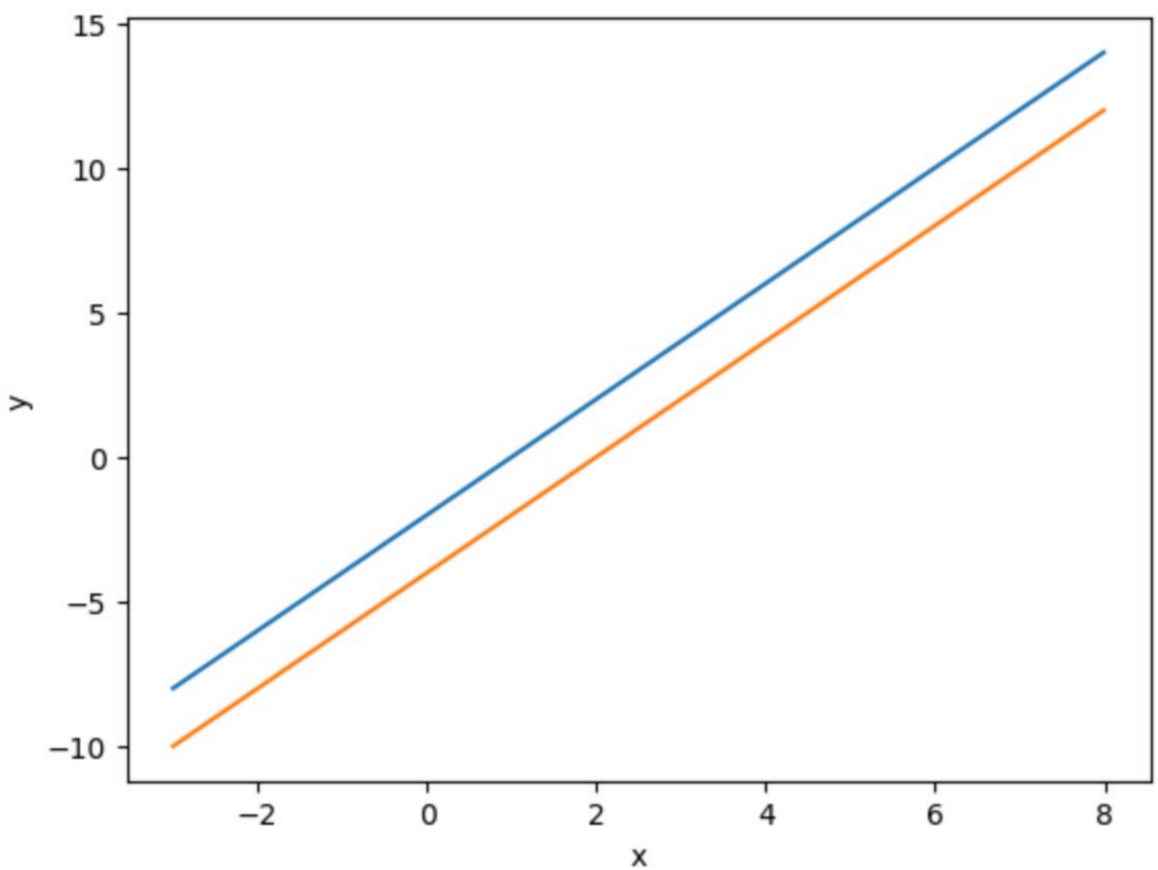
```
y = -2 + 2x
y = -(12/3) + (6/3)x
```

How would you view these equations in Python?

solving systems of linear equations

$$y = -2 + 2x$$
$$y = -4 + 2x$$

How would you view these equations in Python?



linear regression (multiple regression)

$$\begin{array}{cccccc} \text{n x 1} & & \text{n x k} & \text{k x 1} & & \text{n x 1} \\ \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} & = & \begin{bmatrix} 1 & x_{11} & \cdots & x_{1k} \\ 1 & x_{12} & \cdots & x_{2k} \\ \vdots & \vdots & & \vdots \\ 1 & x_{n2} & \cdots & x_{nk} \end{bmatrix} & \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} & + & \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \end{array}$$

$$\mathbf{y} = \mathbf{X}\,\mathbf{b}$$

solving for **b**

simple linear regression

$$
\underset{n \times 1}{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}} = \underset{n \times 2}{\begin{bmatrix} 1 & x_{11} \\ 1 & x_{12} \\ 1 & \vdots \\ 1 & x_{1n} \end{bmatrix}} \underset{2 \times 1}{\begin{bmatrix} b_0 \\ b_1 \end{bmatrix}} + \underset{n \times 1}{\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}}
$$

$$\mathbf{y} = \mathbf{X} \quad \mathbf{b}$$

solving for **b**

simple linear regression

$$\underset{n \times 1}{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}} = \underset{n \times 2}{\begin{bmatrix} 1 & x_{11} \\ 1 & x_{12} \\ 1 & \vdots \\ 1 & x_{1n} \end{bmatrix}} \underset{2 \times 1}{\begin{bmatrix} b_0 \\ b_1 \end{bmatrix}}$$

$$\mathbf{y} = \mathbf{X} \mathbf{b}$$

can we do the same trick as before?

simple linear regression

```
X = np.array([[1, 60],
              [1, 61],
              [1, 62],
              [1, 63],
              [1, 65]])
y = np.array([[3.1],
              [3.6],
              [3.8],
              [4],
              [4.1]])
```

simple linear regression

$$\mathbf{y} = \mathbf{X}\ \mathbf{b}$$

Can we just do this?

$$\mathbf{X}^{-1}\ \mathbf{y} = \mathbf{X}^{-1}\ \mathbf{X}\ \mathbf{b}$$

Try is.

simple linear regression

$$\mathbf{y} = \mathbf{X}\,\mathbf{b}$$

Can we just do this?

$$\mathbf{X}^{-1}\,\mathbf{y} = \mathbf{X}^{-1}\,\mathbf{X}\,\mathbf{b}$$

Try is.

No. Matrix must be square.

simple linear regression

$$\mathbf{y} = \mathbf{X}\ \mathbf{b}$$

We can do this:

$$\mathbf{X}^{\mathbf{T}}\ \mathbf{y} = \mathbf{X}^{\mathbf{T}}\ \mathbf{X}\ \mathbf{b}$$

What will $\mathbf{X}^{\mathbf{T}}\ \mathbf{X}$ give you? What size is it?

simple linear regression

$$\mathbf{y} = \mathbf{X} \mathbf{b}$$

We can do this:

$$\mathbf{X}^T \mathbf{y} = \mathbf{X}^T \mathbf{X} \mathbf{b}$$

What could we do now?

simple linear regression

$$\mathbf{y} = \mathbf{X}\ \mathbf{b}$$

We can do this:

$$\mathbf{X}^T\ \mathbf{y} = \mathbf{X}^T\ \mathbf{X}\ \mathbf{b}$$

What could we do now?

$$(\mathbf{X}^T\ \mathbf{X})^{-1}\ \mathbf{X}^T\ \mathbf{y} = (\mathbf{X}^T\ \mathbf{X})^{-1}\ \mathbf{X}^T\ \mathbf{X}\ \mathbf{b}$$

What will this give you?

simple linear regression

$$\mathbf{y} = \mathbf{X}\ \mathbf{b}$$

We can do this:

$$\mathbf{X}^T\mathbf{y} = \mathbf{X}^T\mathbf{X}\ \mathbf{b}$$

What could we do now?

$$(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\ \mathbf{b}$$

What will this give you?

$$(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{b}$$

$$\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

in practice, inverting large matrices
is computationally very expensive,
and often numerically unstable,
so different algorithms are used

simple linear regression

```
X = np.array([[1, 60], [1, 61], [1, 62], [1, 63], [1, 65]])
y = np.array([[3.1], [3.6], [3.8], [4], [4.1]])

Xt = np.transpose(X)
b = np.linalg.inv(Xt @ X) @ Xt @ y

yhat = b[0] + b[1] * X[:,1]
plt.plot(X[:,1], y[:,0], 'ro', X[:,1], yhat, 'g-')
```

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

in practice, inverting large matrices
is computationally very expensive,
and often numerically unstable,
so different algorithms are used

"real" linear regression (in statistics)

```
from sklearn.linear_model import LinearRegression

model = LinearRegression()
```

uses algorithm that does not require inverting large matrices

```
Xdat = (X[:,1]).reshape((len(X[:,1]),1))
model.fit(Xdat, y)

print(model.intercept_, model.coef_)
```