Linear Algebra:

- l_2 norm:

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

- linearly dependent - vector is linear combination of other vectors.

$$x_n = \sum_{i=1}^{n-1} \alpha_i x_i.$$

- column rank - largest number of columns that are linearly independent set
- rank - assumed col $\overset{m}{\text{rank}}$ = row $\overset{n}{\text{rank}}$ for a matrix $A \in R^{m \times n}$
- inverse of a square matrix, A is invertible if this exists:

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

- orthogonal vectors

$$x^T y = 0$$

normalized vectors

$$\|x\|_2 = 1$$

- orthogonal matrices - square matrix with columns orthogonal to each other and normalized (orthonormal columns)

$$U^T U = I = UU^T$$

↑

orthogonal square matrix with orthonormal columns

fully?

also $U^T = U^{-1}$ inverse of ^square orthogonal matrix is its transpose.

- span of a set of vectors is all vectors expressed as linear combination.
- projection of a vector onto the span of matrices

$$\text{Proj}(y; \{x_1 \dots x_n\}) = \underset{v \in \text{span}(\{x_1 \dots x_n\})}{\text{argmin}} \underbrace{\|y - v\|_2}_{\text{euclidean norm}}$$

↑

vector y projected onto span of $x_1 \dots x_n$

- range is span of cols of matrix A
- nullspace of a matrix is all vectors = 0 when multiplied by A
- determinant of a square matrix is a function. Absolute

value of determinant of square matrix A is volume of set S^2.

If 2×2 matrix → volume is area in 2D

◦ determinant of identity is 1 det I = 1, volume of a unit hypercube = 1

· gradient ($\mathbb{R}^n - \mathbb{R}^1$) ⟶ $\nabla f(x,y) = \begin{bmatrix} \partial f/\partial x \\ \partial f/\partial y \end{bmatrix}$

· jacobian ($\mathbb{R}^n - \mathbb{R}^m$) vector to vector of same or diff shape

◦ $\underset{\mathbb{R}^2}{f(x,y)} = \underset{\mathbb{R}^2}{\begin{bmatrix} 2x + y^3 \\ e^y - 13x \end{bmatrix}}$

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix} = \begin{bmatrix} 2 & 3y^2 \\ -13 & e^y \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \nabla f_1^T \\ \nabla f_1^T \end{bmatrix}$$

· jacobian chain rule

$$f(x,y) = \begin{bmatrix} \sin(x^2+y) \\ \ln(y^3) \end{bmatrix}$$

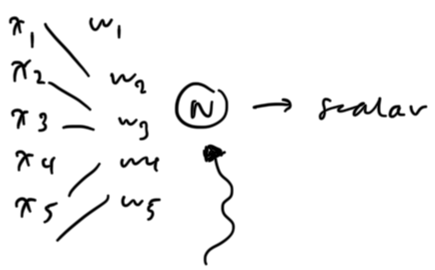
$$\frac{\partial f}{\partial x} = \frac{\partial f}{\cancel{\partial g}} \cdot \frac{\cancel{\partial g}}{\partial x} \quad \text{where} \quad f(x) = \begin{bmatrix} \sin(g) \\ \ln(g) \end{bmatrix}, \; g(x) = \begin{bmatrix} x^2+y \\ y^3 \end{bmatrix}$$

$$= \begin{bmatrix} \cos g_1 & 0 \\ 0 & \frac{1}{g_2} \end{bmatrix} \cdot \begin{bmatrix} 2x & 1 \\ 0 & 3y^2 \end{bmatrix}$$

$$= \begin{bmatrix} 2x \cos g_1 & \cos g_1 \\ 0 & \frac{3y^2}{g^2} \end{bmatrix}$$

* if intermediate can't be chain rule-d then put 1

$$= \begin{bmatrix} 2x\cos(x^2+y) & \cos(x^2+y) \\ 0 & \frac{3}{y} \end{bmatrix}$$

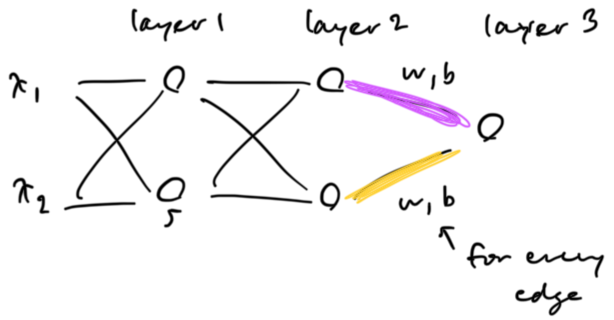


$$\sigma\Big(\underbrace{\sum_{i=1}^{N} x_i w_i}_{x^T w}\Big) + b$$

Relu

vector to

scalar function.



w^l_{jk}

w^3_{11} — going into layer 3

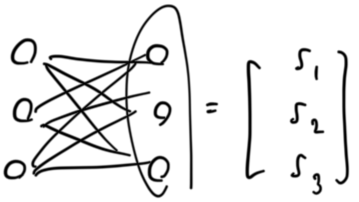
node in layer l

node in layer l−1

w^3_{12}

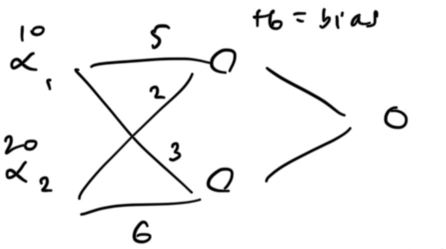
j is going to k is coming from

b^l_j $b^2_1 = 5$



each node passes weights and biases to next node. output vector with scalars is passed to next layer.

- weight matrix: w^l for entire layer l (feed forward)



↑ backprop improves weights after fully formed network.

$\sigma(w^l \cdot a^{l-1} + b^l)$

$\sigma(z^1)$

OR

$a_0 =$

$\sigma(w^1 a^0 + b^1) =$

$\sigma(w^2 a^1 + b^2) =$ etc.

k — 2 to 1!

$$w^1 = j\begin{bmatrix} \overset{1,1}{5} & \overset{1,2}{2} \\ \overset{2,1}{3} & \overset{2,2}{6} \end{bmatrix}\begin{bmatrix} 10 \\ 20 \end{bmatrix} = \begin{bmatrix} 90 \\ 150 \end{bmatrix} + 6$$

$(2,2) \times (2,1) = (2 \times 1)$

$z^1 = \begin{bmatrix} 96 \\ 156 \end{bmatrix}$ is output vector, now put into activation function that's non-linear

Formulas for all ML topics:

- Unparametrized - ex knn, no assumptions on size of input data.
- Regularization - applying penalties to parameters of a model.
  - $L(w) = \frac{1}{2} \sum_{n=1}^{N} (y_n \underbrace{-w^T \phi_n^2}_{\text{squaring weight}}) + \frac{\lambda}{2} w^T w$ ← larger values of λ produce

↑