2- linear algebra review

vectors =

↳ column vector (v) = $R^{n\times 1}$ $v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \end{bmatrix}$ ↳ row vector (v^T) = $R^{1\times n}$ $v^T = [v_1, v_2 \dots]$

matrices =

↳ $A \in R^{m\times n}$ array of numbers with m rows and n columns

↳ upper left corner of image is $[y,x] = (0,0)$ in python

↳ tensor = multidimensional matrices (color images 3 numbers per pixel = $m\times n\times 3$)

↳ grayscale images 1 number per pixel, so $m\times n$ matrix

vector norm = $\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$ always non-negative

- $p=1$ (manhattan dist) $\|x\|_1 = \sum_{i=1}^{n} |x_i|$
- $p=2$ (euclidean dist) $\|x\|_2 = \left(\sum_{i=1}^{n} |x_i^2|\right)^{\frac{1}{2}}$

when p gets larger, it will converge to largest $|x_i|$ $\|x\|_\infty = \max_i |x_i|$

dot product = $\langle x,y\rangle = x^T y = \sum_{i=1}^{n} x_i y_i = \|x\|.\|y\|.\cos\theta$ (projection)

multiplication = $\underset{2\times 2}{A} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ $\underset{(2\times 3)}{B} = \begin{bmatrix} 5 & 6 & 7 \\ 8 & 9 & 10 \end{bmatrix}$ $A\times B = \begin{bmatrix} 1.5+2.8 & 1.6+2.9 & 1.7+2.10 \\ 3.5+4.8 & 3.6+4.9 & 3.7+4.10 \end{bmatrix}_{2\times 3}$

↳ entry (i,j) = dot product of i'th row of A and j'th column of B

transpose = $\begin{bmatrix} 0 & 1 \\ 2 & 3 \\ 4 & 5 \end{bmatrix}^T = \begin{bmatrix} 0 & 2 & 4 \\ 1 & 3 & 5 \end{bmatrix}$ $(ABC)^T = C^T B^T A^T$

determinant = $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$

↳ $\det(A) = ad - bc$

- $\det(A^T) = \det(A)$
- $\det(AB) = \det(BA)$
- $\det(A) = 0 \iff A$ is singular (not inventible)
- $\det(A^{-1}) = \frac{1}{\det(A)}$

trace = $tr(A)$ sum of diognals • $A = \begin{bmatrix} 1 & 3 \\ 5 & 7 \end{bmatrix} \rightarrow tr(A) = 1+7 = 8$ • $tr(AB) = tr(BA)$

special matrices

↳ identity = $\begin{bmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \end{bmatrix}$ $I \times A = A$ square

↳ diognal = $\begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix}$ square, numbers along the diognal are 0

↳ symmetric = $A^T = A$

↳ skew-symmetric = $A^T = -A$

transformation matrices

↳ scaling = $\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \times \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \end{bmatrix}$

↳ translation = $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+t_x \\ y+t_y \\ 1 \end{bmatrix}$

↳ rotation = $\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$ $\theta \rightarrow$ counter clockwise

- $\begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \rightarrow$ first scales and then translates

↳ homogeneous system = to being able to translate matrix, add new coordinate for constant

$\begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} ax+by+c \\ dx+ey+f \\ 1 \end{bmatrix}$ if $\begin{bmatrix} x \\ y \\ 7 \end{bmatrix} \rightarrow \begin{bmatrix} x/7 \\ y/7 \\ 1 \end{bmatrix}$

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

pseudo-inverse = to find X in $AX = B \rightarrow A^{-1}.A.X = A^{-1}B \rightarrow X = A^{-1}.B$ np.linalg.inv(A) * B

↳ too costly and wrong due to floating point

↳ np.linalg.solve (A,B) = returns closest if no sol, smallest if many

linear independence = if some vector cannot be expressed as a linear combination of others

matrix rank = number of linearly independent rows or columns in the matrix $\begin{bmatrix} 2 & 3 & -1 \\ 0 & 1 & 4 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow rank(A) = 2$ (same)

↳ for transformation matrices, it tells the dimension of the output $\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x+y \\ 2x+2y \end{bmatrix}$ rank=1

↳ full rank matrix = if $m\times m$ matrix's rank is m → else singular

eigenvector (x) and eigenvalue (λ) = when linear transformation A is applied to an eigen vector x, it does not change its direction, it scales it by eigenvalue λ

↳ $Ax = \lambda x$ $x \neq 0$ $Ax = \lambda I x \rightarrow (\lambda I - A)x = 0 \rightarrow \lambda I - A = 0$

- $trace(A) = \sum_i^n \lambda_i$
- $determinant(A) = \prod_i^n \lambda_i$
- $rank(A)$ = # non-zero eigenvalues of A
- eigen values of a diognal matrix = diognal entries

↳ spectrum = set of all eigenvalues of A eigenpair = (λ, x)

↳ spectral radius = magnitude of the largest eigenvalue

↳ the eigenvectors of A are orthanormal

- $D = \begin{bmatrix} \lambda_1 & & \\ & \lambda_2 & \\ & & \ddots \end{bmatrix}$ $V = [v_1\ v_2\ v_3\ \dots]$ $Av = vD$ $A = vDv^{-1}$

matrix calculus

↳ the size of $\nabla_A f(A)$ (gradient) is always same as A

the hessian matrix = $\nabla_x^2 f(x) \in R^{n\times n} = \begin{bmatrix} \frac{d^2 f(x)}{dx_1^2} & \frac{d^2 f(x)}{dx_1 dx_2} & \frac{d^2 f(x)}{dx_1 dx_3} & \cdots \\ \frac{d^2 f(x)}{dx_2 dx_1} & \frac{d^2 f(x)}{dx_2^2} & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$ always symmetric

every entry = $\nabla_x^2 f(x)_{ij} = \frac{d^2 f(x)}{dx_i\, dx_j}$

↳ $\frac{d^2 f(x)}{dx_1 dx_2} = \frac{d^2 f(x)}{dx_2 dx_1}$ order does not matter