Numerical Analysis for Machine Learning

1) $A\vec{x}=\vec{b}$ solvability (4 fundamental spaces)
2) $A\vec{x}=\lambda\vec{x}$ } generalization for any matrix
3) $A\vec{v}=\sigma\vec{u}$ SVD (single value decomposition), used for PCA, Least squares, A^+ pseudo-inverse of A
4) Minimization quadratic functionals
5) Factorization QR factorization

Matrix Vector Multiplication

$$\vec{c}=\underbrace{\begin{bmatrix}1&4\\2&5\\3&6\end{bmatrix}}_{A_1}\underbrace{\begin{bmatrix}x_1\\x_2\end{bmatrix}}_{x}=\begin{bmatrix}x_1+4x_2\\2x_1+5x_2\\3x_1+6x_2\end{bmatrix}=\begin{bmatrix}1\\2\\3\end{bmatrix}x_1+\begin{bmatrix}4\\5\\6\end{bmatrix}x_2$$

$\vec{c}\in C(A_1)$ (column space of A (plane))

$\Downarrow$

$\begin{bmatrix}1\\2\\3\end{bmatrix}\quad\begin{bmatrix}4\\5\\6\end{bmatrix}$ both linearly independent

$A_{2\,(1)}$

$$\begin{bmatrix}1&4&7\\2&5&8\\3&6&9\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}$$

$\vec{a}_1\ \vec{a}_2\ \vec{a}_3$

$\vec{a}_3=-\vec{a}_1+2\vec{a}_2 \Rightarrow C(A_1)=C(A_2)$ (but linearly independent)

$C(A_3)=\mathbb{R}^3$

$$A_4=\begin{bmatrix}1&2&3\\2&4&6\\3&6&9\end{bmatrix}$$

$\rightarrow$ only 1 column vector is linearly independent

$C(A_4)=\begin{bmatrix}1\\2\\3\end{bmatrix}$ (line)

dim $\Rightarrow$ rank of a matrix

$\dim(A_1)=2$
$\dim(A_2)=2$
$\dim(A_3)=3$
$\dim(A_4)=1$

$A\vec{x}=\vec{b}$ ($m\times n$ matrix)

$$\begin{bmatrix}1&4&7\\2&5&8\\3&6&10\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}1\\2\\3\end{bmatrix}x_1+\begin{bmatrix}4\\5\\6\end{bmatrix}x_2+\begin{bmatrix}7\\8\\10\end{bmatrix}x_3=\begin{bmatrix}b_1\\b_2\\b_3\end{bmatrix}$$

linear combination of the columns of A $\in C(A_3)$ $\Rightarrow$ $\vec{b}$ must be in the column space of A: $\vec{b}\in C(A_3)$

Find $C(A)$ $\boxed{\vec{a_1},\vec{a_2},\dots,\vec{a_n}}$ columns of A

- put $\vec{a_1}$ in $C(A)$
- if $\vec{a_2}=\alpha\vec{a_1}\Rightarrow\vec{a_2}\notin C(A)$ else put $\vec{a_2}$ in $C(A)$

$\circlearrowleft$ reiterate

$A_2=CR$ (row-reduced echelon form (rref)) $\rightarrow$ useful when $\text{rank}(A)\le n$

$$\underbrace{\left[\begin{array}{c|c}1&4\\2&5\\3&6\end{array}\right]}_{C\ (3\times2)}\underbrace{\left[\begin{array}{cc|c}1&0&-1\\0&1&2\end{array}\right]}_{R\ (2\times3)}$$

combination of $\vec{a_1}$ and $\vec{a_2}$ to retrieve the respective column of A_2

$$A_1=\begin{bmatrix}1&4\\2&5\\3&6\end{bmatrix}\qquad A_1^T=\left[\begin{array}{ccc}1&2&3\\4&5&6\\ \vec{a_1}&\vec{a_2}&\vec{a_3}\end{array}\right]$$

$\vec{a_3}=2\vec{a_2}-\vec{a_1}$

$\dim(C(A))=\dim(C(A^T))=r\le n$ same rank

Matrix - Matrix Multiplication

$$C=AB=\begin{bmatrix}|&&|\\\vec{a_1}&\cdots&\vec{a_n}\\|&&|\end{bmatrix}\begin{bmatrix}—&\vec{b_1}&—\\&\vdots&\\—&\vec{b_n}&—\end{bmatrix}=\vec{a_1}\vec{b_1}+\dots+\vec{a_n}\vec{b_n}$$

$\vec{a_k}$, $k\in[1,n]$ is a column vector; $\vec{b_k}$, $k\in[1,n]$ is a row vector

matrices of rank 1

$$C=A\Sigma B=\delta_1\vec{a_1}\vec{b_1}+\dots+\delta_n\vec{a_n}\vec{b_n}\dots$$

$$\begin{bmatrix}1&2\\3&4\end{bmatrix}\begin{bmatrix}2&1\\2&3\end{bmatrix}=\begin{bmatrix}2&1\\6&3\end{bmatrix}+\begin{bmatrix}4&6\\8&12\end{bmatrix}=\begin{bmatrix}6&7\\14&15\end{bmatrix}$$

rank 1 matrix

$\vec{a_1}\vec{b_1}$ $\vec{a_2}\vec{b_2}$

FACTORIZATIONS

1) $A = LU$ $(PA = LU)$

2) $A = QR$ { Q is an orthogonal matrix; R is an upper triangular matrix

3) $S = S^T$ (symmetric matrix) ($n \times n$ matrices)

$S = Q \Lambda Q^T$ { Q is an orthogonal matrix; Λ is a diagonal matrix

4) $A = X \Lambda X^{-1}$ { Λ is a diagonal matrix; A is a square matrix

5) $A = U \Sigma V^T$ { U and V are orthogonal matrices; Σ is a pseudo-diagonal matrix

Orthogonal Matrix

$QQ^T = Q^TQ = I$

if also implies that $\|Q\vec{x}\| = \|\vec{x}\| \rightarrow \det(Q) = \pm 1$

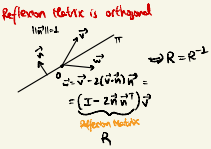
$\|Q\vec{x}\|^2 = (Q\vec{x})^T Q\vec{x} = \vec{x}^T Q^T Q \vec{x} = \vec{x}^T \vec{x} = \|\vec{x}\|^2$

Rotation Matrix is orthogonal

$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$

Reflection Matrix is orthogonal

$\|\vec{n}\| = 1$



$\Rightarrow R = R^{-1}$

$= \vec{v} - 2(\vec{v}\vec{n}^T)\vec{n} = (I - 2\vec{n}\vec{n}^T)\vec{v}$

Reflection Matrix

R

$S = (Q\Lambda)Q^T = \vec{Q}Q^T = \lambda_1 \vec{q}_1 \vec{q}_1^T + \dots + \lambda_n \vec{q}_n \vec{q}_n^T$ (eigenvalues / eigenfrequencies)

$\vec{Q} = \vec{q_1}\vec{\lambda_1} + \dots + \vec{q_n}\vec{\lambda_n}$

Spectral decomposition

$\Rightarrow S\vec{q_1} = \lambda_1 \vec{q_1}$

$A \in \mathbb{R}^{m \times n}$, $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$, $\Sigma \in \mathbb{R}^{m \times n}$

supposing that $m > n$

$\Sigma = m$ { $\sigma_1 \dots \sigma_n$; 0 ; n

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

$A \in \mathbb{R}^{m \times n}$

$\text{rank}(A) = r$

1) $C(A) \subset \mathbb{R}^m$, $\dim(C(A)) = r$

2) $C(A^T) \subset \mathbb{R}^n$, $\dim(C(A^T)) = r$

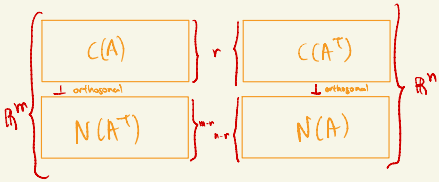
3) $N(A) \subset \mathbb{R}^n$, $\dim(N(A)) = ?$

4) $N(A^T) \subset \mathbb{R}^m$, $\dim(N(A^T)) = ?$

$A\vec{x} = \vec{0}$ $\quad N(A) = \{\vec{x} \in \mathbb{R}^n : A\vec{x} = \vec{0}\}$

$\text{Ker}(A) = N(A)$ Null space / Kernel

$A^T\vec{x} = \vec{0}$ $\quad N(A^T) = \{\vec{x} \in \mathbb{R}^m : A^T\vec{x} = \vec{0}\}$



1) $\vec{x} = \vec{0} \in N(A)$

2) if $\vec{x}, \vec{y} \in N(A) \Rightarrow A(\vec{x} + \vec{y}) = \vec{0}$

3) if $\vec{x} \in N(A) \Rightarrow \alpha\vec{x}, \alpha \in \mathbb{R} \Rightarrow A(\alpha\vec{x}) = \vec{0}$

Ex. $A = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$ (n×m)

$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} \begin{bmatrix} x_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow$ $x_1 + 4x_2 + 7x_3 = 0$, $2x_1 + 5x_2 + 8x_3 = 0$, $3x_1 + 6x_2 + 9x_3 = 0$

$x_1 = -4x_2 - 7x_3$

$-3x_2 - 6x_3 = 0$

$-6x_2 - 12x_3 = 0$

rank = 2 $\Rightarrow$ 1 degree of freedom (dof)

$m - r = 3 - 2 = 1$

but $\Rightarrow A = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 10 \end{bmatrix}$ $\Rightarrow$ If A is full-rank, rank = 3, 0 dof $N(A) = \{\vec{0}\}$

$A \in \mathbb{R}^{m \times n}$ $\quad \text{rank}(A) = r < n$ $\quad A = CR$

$A = [A_1 \; A_2]$ $\quad A_1 \in \mathbb{R}^{m \times r}$ $\quad A_2 \in \mathbb{R}^{m \times (n-r)}$

$A = [A_1 \; A_1 B]$, $B \in (\mathbb{R}^r)^{r \times (n-r)}$

$K = \begin{bmatrix} -B \\ I_{n-r} \end{bmatrix}$, $K \in \mathbb{R}^{n \times (n-r)}$

$AK = [A_1 \; A_1 B] \begin{bmatrix} -B \\ I_{n-r} \end{bmatrix} = -A_1 B + A_1 B = \vec{0}$ ($m \times (n-r)$)

$AK = 0 \Rightarrow A\vec{k_i} = \vec{0} \rightsquigarrow \vec{k_i} \in N(A)$

$A = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix} \sim \begin{bmatrix} 1 & 4 & 7 \\ 0 & -3 & -6 \\ 0 & -6 & -12 \end{bmatrix} \sim \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix}$

A_1 $\quad B$

$K\vec{u} = \vec{0} \Rightarrow \vec{u} = \vec{0}$

$\begin{bmatrix} -B \\ I \end{bmatrix} \vec{u} = \vec{0} \Rightarrow \begin{bmatrix} -B\vec{u} \\ \vec{u} \end{bmatrix} = \begin{bmatrix} \vec{0} \\ \vec{0} \end{bmatrix} \begin{matrix} \} r \\ \} n-r \end{matrix} \Rightarrow \vec{u} = \vec{0}$

$A\vec{x} = \vec{0}$ (*) each $\vec{x}$ that satisfy (*) must be a linear combination of the columns of K

$(A_1\vec{x} = \vec{0} \in \mathbb{R}^m \Rightarrow \vec{x} = \vec{0} \in \mathbb{R}^r)$

$A\vec{v} = \vec{0} \in \mathbb{R}^m$

$= [A_1 \; A_1 B] \begin{bmatrix} \vec{u_1} \\ \vec{u_2} \end{bmatrix} = [A_1\vec{u_1} + A_1 B\vec{u_2}] = A_1[\vec{u_1} + B\vec{u_2}] = \vec{0}$

$\vec{u_1} + B\vec{u_2} = \vec{0} \Rightarrow \vec{u_1} = -B\vec{u_2}$

$\vec{u} = \begin{bmatrix} -B\vec{u_2} \\ \vec{u_2} \end{bmatrix} = \begin{bmatrix} -B \\ I \end{bmatrix} \vec{u_2} = K\vec{u_2}$ $\quad \dim(N(A)) = n - r$

EIGENVALUES & VECTORS

- A general square matrix $n \times n$
  + symmetry
  + positive definite (spd)

$A\vec{x}_i = \lambda_i \vec{x}_i \quad i = 1, \dots, n$

$X^{-1} A X = \Lambda$ (eigenvector; diagonal matrix of eigenvalues)

A^2 ? $\Rightarrow A^2 \vec{x}_i = A(A\vec{x}_i) = A(\lambda_i \vec{x}_i) = \lambda_i (\lambda_i \vec{x}_i) = \lambda_i^2 \vec{x}_i$

$e^{A\vec{x}} = e^{\lambda \vec{x}}$ $\quad \frac{d\vec{u}}{dt} = A\vec{u}$

Power method (find max eigenvalue): $\vec{w}^{(n)} = A \vec{w}^{(n-1)} = A^n w^{(0)}$

If A is full rank then any vector $\vec{v} \in \mathbb{R}^n$ can be written as a linear combination of the eigen vectors $(\vec{x}_i)$

Similar Matrices

$A, B \in \mathbb{R}^{n \times n}$ $\quad B = M^{-1} A M$ (if M is invertible)

$M^{-1} A M \vec{v} = \lambda \vec{v} \Rightarrow A M \vec{v} = \lambda M \vec{v} \Rightarrow A \vec{w} = \lambda \vec{w}$ (same eigenvalues)

- Power Method (PM) on A (max eigenvalue)
- Inverse PM on A^{-1} (min eigenvalue)
- PM with shift $(A - \alpha I)^{-1}$ $\alpha \in \mathbb{R}$ (closest eigenvalue to α)

Deflation method $\lambda_1, \vec{x}_1$

$$\begin{bmatrix} \lambda_1 & \vec{b}_k^T \\ 0 & A_2 \end{bmatrix}$$

QR Factorization

$A \in \mathbb{R}^{m \times n}$ $\quad m \geq n$ independent columns

$A = QR$ (m×n)(n×n) $\quad$ Q has orthogonal columns; R is upper triangular

How can we compute Q? Gram-Schmidt orthogonalization procedure

$A \to \dots \to Q$

$A = \begin{bmatrix} | & & | \\ \vec{a}^1 & \dots & \vec{a}^n \\ | & & | \end{bmatrix}$

$\vec{q}_i^T \vec{q}_j = \begin{cases} 1 & i = j \\ 0 & \text{otherwise} \end{cases}$

$\vec{q}_1 = \frac{\vec{a}^1}{\|\vec{a}^1\|}$

$\vec{q}_2 = \frac{\vec{a}^2 - (\vec{q}_1^T \vec{a}^2)\vec{q}_1}{\|\vec{a}^2 - (\vec{q}_1^T \vec{a}^2)\vec{q}_1\|}$ ($\vec{a}^2$, $(\vec{q}_1^T \vec{a}^2)\vec{q}_1$)

$\vec{q}_3 = \frac{\vec{a}^3 - (\vec{q}_1^T \vec{a}^3)\vec{q}_1 - (\vec{q}_2^T \vec{a}^3)\vec{q}_2}{\|\vec{a}^3 - (\vec{q}_1^T \vec{a}^3)\vec{q}_1 - (\vec{q}_2^T \vec{a}^3)\vec{q}_2\|}$

Ex: $A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} | & | & | \\ \vec{q}^1 & \vec{q}^2 & \vec{q}^3 \\ | & | & | \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ 0 & r_{22} & r_{23} \\ 0 & 0 & r_{33} \end{bmatrix}$

QR iteration

$A = A^{(0)} = Q^{(0)} R^{(0)}$

$A^{(1)} = Q^{(0)T} A^{(0)} Q^{(0)}$ $\quad A^{(1)} = Q^{(1)} R^{(1)}$

$A^{(2)} = \dots$ $\quad A^{(n)}$ = upper triangular $\Rightarrow$ diagonal values are the eigenvalues

---

Symmetric matrix $S \in \mathbb{R}^{n \times n}$ $\quad S = S^T$

$S = Q \Lambda Q^T$ (Q orthogonal matrix; λ_i are real)

i) $S\vec{x} = \lambda \vec{x}$, $\quad S\vec{y} = 0\vec{y}$

$\vec{x} \in C(S) = C(S^T)$ $\quad \vec{y} \in N(S)$ (implies first) $\Rightarrow \vec{x} \perp \vec{y}$

ii) $S\vec{x} = \lambda \vec{x}$, $\quad S\vec{v} = \alpha \vec{v}$

$(S - \alpha I)\vec{x} = \lambda \vec{x} - \alpha \vec{x} = (S - \alpha I)\vec{v} = 0\vec{v}$

$\Rightarrow (S - \alpha I)\vec{x} = (\lambda - \alpha)\vec{x}$ (shifting the eigenvalues)

$\vec{x} \in C(S - \alpha I) = (N(S - \alpha I))^\perp$ $\quad \vec{v} \in N(S - \alpha I)$

$\vec{x}^T S \vec{x} = \lambda \vec{x}^T \vec{x}$ (ca) (cc)

$(a + ib)(a - ib) = (a^2 + b^2)$

$\lambda = \frac{\vec{x}^T S \vec{x}}{\vec{x}^T \vec{x}}$ Rayleigh's quotient

---

(Symmetric) Positive-definite (spd)

i) $\lambda_i > 0 \quad i = 1, \dots, n$

ii) $\vec{x}^T S \vec{x} > 0$, $\forall \vec{x} \in \mathbb{R}^n$, $\vec{x}^T S \vec{x} = 0 \Rightarrow \vec{x} = \vec{0}$

iii) Leading determinants are positive

iv) $S = B^T B$ Cholesky factorization (R upper triangular)

v) All pivot elements are positive

$\lambda > 0$

① $S\vec{x} = \lambda \vec{x}$

$\vec{x}^T S \vec{x} = \lambda \vec{x}^T \vec{x} = \lambda \|\vec{x}\|^2 \geq 0$

② $\vec{v} = (c_1 \vec{x}_1 + c_2 \vec{x}_2 + \dots + c_n \vec{x}_n)$

$(c_1 \vec{x}_1 + c_2 \vec{x}_2 + \dots + c_n \vec{x}_n)^T S (c_1 \vec{x}_1 + c_2 \vec{x}_2 + \dots + c_n \vec{x}_n)$

i) $c_1 \vec{x}_1^T S \vec{x}_1 = c_1^2 \lambda_1 \vec{x}_1^T \vec{x}_1 = c_1^2 \lambda_1 \|\vec{x}_1\|^2 > 0$

ii) $c_1 c_2 \vec{x}_1^T S \vec{x}_2 = c_1 c_2 \lambda_1 \vec{x}_1^T \vec{x}_2 = 0$

③ $S = B^T B$

$\vec{v}^T (B^T B) \vec{v} = (\vec{v}^T B^T)(B\vec{v}) =$

$= (B\vec{v})^T (B\vec{v}) = \|B\vec{v}\|^2 \geq 0$

SVD (Single Value Decomposition)

↳ works for every matrix (always possible)

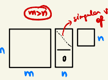
Applications

- **Least-squares approximation**
  finding the regression line representing the trend of data points
  need to find ↳ slope, intersection of y-axis
  Via SVD, we can introduce a pseudo-inverse matrix
- **Low-rank approximation**
  (Eckart-Young)
  based on the usage of the SVD

$A \in \mathbb{R}^{m \times n}$ $m \Rightarrow$ # samples
$rank(A) = r < n$ $n \Rightarrow$ # features

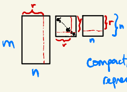
SVD ⇒ $A = U \Sigma V^T$, where { U $m \times m$ orthogonal; V^T $n \times n$ orthogonal; Σ almost diagonal $n \times n$ }

(m>n) ← FULL SVD



↳ generally $\sigma_1 > \sigma_2 > \sigma_3 > \dots > \sigma_n$

ECONOMY SVD



compact/reduced representation
$(m \times n)(n \times n)(n \times n)$
or (if $r < n$)
$(m \times r)(r \times r)(r \times n)$

$AV = U\Sigma$
$A\vec{v}_i = \sigma_i \vec{u}_i$

⊗ since $rank(A) = r$, then
$\sigma_1 \dots \sigma_r > 0$
$\sigma_{r+1} \dots \sigma_n = 0$

$\Rightarrow$ $AV = U\Sigma$:
$A\vec{v}_1 = \sigma_1 \vec{u}_1$, ..., $A\vec{v}_r = \sigma_r \vec{u}_r$ } r: $\vec{u}_i \in C(A), \forall_i: i \le r$
$A\vec{v}_{r+1} = \vec{0}$, ..., $A\vec{v}_n = \vec{0}$ } n−r: $\vec{v}_i \in N(A), \forall_i: r < i \le n$

$rank(A) = r \le n$
$A = U\Sigma V^T =$
$= \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$ ($\sigma_i > \sigma_{i+1}$, so the first term is the leading term)
$\tilde{r} < r$ ⇝ $\sum_{i=1}^{\tilde{r}} \sigma_i \vec{u}_i \vec{v}_i^T$
↳ $\tilde{r}$ approximation of A

$\|A - \tilde{A}\| \le \|A - B\|, \forall B \in \mathbb{R}^{m \times n}$

Proof of existence of SVD

$A \in \mathbb{R}^{m \times n}$ $rank(A) = r$

- symmetric i) $(A^TA) = A^TA$
- positive definite ii) $\vec{x}^T(A^TA)\vec{x} = (\vec{x}^TA^T)A\vec{x} =$
$= (A\vec{x})^T A x = \|A\vec{x}\|^2 \ge 0$

$A^TA = V\Lambda V^T =$
$= \sum_{i=1}^{n} \lambda_i \vec{v}_i \vec{v}_i^T =$
↖ σ_i^2

$rank(A^TA) = ?$
- $A\vec{x} = \vec{0} \Rightarrow \vec{x} \in N(A)$ ↘ consequence
$A^T(A\vec{x}) = A^T\vec{0} = \vec{0} \Rightarrow \vec{x} \in N(A^TA)$
- $A^TA\vec{x} = \vec{0} \Rightarrow \vec{x} \in N(A^TA)$
$\vec{x}^TA^TA\vec{x} = \|A\vec{x}\|^2 = 0$ // $x \in N(A^TA) \Rightarrow x \in N(A)$

$rank(A^TA) = rank(A)$

$(\sigma_i^2, \vec{v}_i)$
$A^TA = V\Lambda V^T$
$A^TAV = V\Lambda$ } eigenvalue eigenvector relationship
$A^TA\vec{v}_i = \sigma_i^2 \vec{v}_i$

$\vec{u}_i = \frac{A\vec{v}_i}{\sigma_i}$
$A\vec{v}_i = \sigma_i \vec{u}_i$
orthonormal; orthonormal & eigenvectors of A

i) $\vec{u}_i$ are unitary vectors ($\|\vec{u}\| = 1$)
$\vec{u}_i^T\vec{u}_i = \left(\frac{A\vec{v}_i}{\sigma_i}\right)^T\left(\frac{A\vec{v}_i}{\sigma_i}\right) =$
$= \frac{\vec{v}_i^TA^TA\vec{v}_i}{\sigma_i^2} = \frac{\vec{v}_i^T\sigma_i^2\vec{v}_i}{\sigma_i^2} = \vec{v}_i^T\vec{v}_i = 1$

ii) $\vec{u}_i \perp \vec{u}_j$
$\vec{u}_i^T\vec{u}_j = \left(\frac{A\vec{v}_i}{\sigma_i}\right)^T\left(\frac{A\vec{v}_j}{\sigma_j}\right) =$
$= \frac{\vec{v}_i^TA^TA\vec{v}_j}{\sigma_i\sigma_j} = \frac{\vec{v}_i^T\sigma_j^2\vec{v}_j}{\sigma_i\sigma_j} = \frac{\sigma_j}{\sigma_i}\vec{v}_i^T\vec{v}_j = \emptyset$ (orthogonal)

iii) $\vec{u}_i$ are eigenvectors of AA^T with eigenvalues σ_i^2
$AA^T\vec{u}_i = AA^T\left(\frac{A\vec{v}_i}{\sigma_i}\right) = \frac{AA^TA\vec{v}_i}{\sigma_i} =$
$= \frac{A\sigma_i^2\vec{v}_i}{\sigma_i} = \sigma_i^2\left(\frac{A\vec{v}_i}{\sigma_i}\right) = \sigma_i^2\vec{u}_i$
↳ β_i ⇒ eigenvalue

Ex. 1

$A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}$
rank(A) = 2

$X = A^TA = \begin{bmatrix} 25 & 7 \\ 7 & 25 \end{bmatrix}$ eigs(x) = { $\lambda_1 = 18$, $\vec{v}_1 = \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$; $\lambda_2 = 32$, $\vec{v}_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$ }

$Y = AA^T = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$ $\vec{u}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ $\vec{u}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$U = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ $V = \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}$

$\Sigma = \begin{bmatrix} \sqrt{18} & 0 \\ 0 & \sqrt{32} \end{bmatrix}$

$A \in \mathbb{R}^{n \times n}$ $A = A^T$

$A = QS$
Q: orthogonal; S: symmetric positive semi-definite ↗ polar decomposition

$A = U\Sigma V^T = (UV^T)(V\Sigma V^T)$
with $Q = UV^T$, $S = V\Sigma V^T$

Properties of the SVD

i) If A is orthogonal then $\sigma_i = 1$ $\boxed{A^TA = I}$
ii) All eigenvalues of a square matrix are $\le \sigma_1$
$\|A\vec{x}\| = \|U\Sigma V^T\vec{x}\| = \|\Sigma V^T\vec{x}\| \le \sigma_1$
orthogonal matrix doesn't affect the length of the vector $\boxed{\|V^T\vec{x}\| = \sigma_1\|\vec{x}\|}$
$\Rightarrow \|A\vec{x}\| \le \sigma_1\|\vec{x}\|$
$\|A\vec{v}_i\| = \|\lambda_i\vec{v}_i\| = |\lambda_i|\|\vec{v}_i\|$
$|\lambda|\|\vec{v}_i\| \le \sigma_1\|\vec{v}_i\| \Rightarrow \boxed{|\lambda_i| \le \sigma_1}$ $\forall_i$

$A \in \mathbb{R}^{m \times n}$ $m >> n$
AA^T } which is better? $AA^T (m \times m)$
A^TA } $A^TA (n \times n)$ → method of snapshots

Ex. 2

$A = \begin{bmatrix} 4 & 3 \\ 8 & 6 \end{bmatrix}$ rank(A) = 1

$X = A^TA = \begin{bmatrix} 80 & 60 \\ 60 & 45 \end{bmatrix}$
eigs(x) = { $\lambda_1 = 125$, $\vec{v}_1 = \begin{bmatrix} -4/5 \\ -3/5 \end{bmatrix}$; $\lambda_2 = 0$, $\vec{v}_2 = ?$ ⇒ $\vec{v}_2 \perp \vec{v}_1$ }
$\vec{v}_2 = \begin{bmatrix} 3/5 \\ -4/5 \end{bmatrix}$

$\vec{u}_1 = \frac{A\vec{v}_1}{\sigma_1} = \frac{1}{\sqrt{125}}\begin{bmatrix} 4 & 3 \\ 8 & 6 \end{bmatrix}\begin{bmatrix} -4/5 \\ -3/5 \end{bmatrix} =$
$= \frac{1}{\sqrt{125}}\begin{bmatrix} -5 \\ -10 \end{bmatrix}$

$\vec{u}_2 = \frac{A\vec{u}_1}{\sigma_2}$ ⇝ $A = U\Sigma V^T = \hat{U}\hat{\Sigma}\hat{V}^T$
$\sigma_2 = 0$

$A = \frac{1}{\sqrt{125}}\begin{bmatrix} -5 \\ -10 \end{bmatrix}\begin{bmatrix} \sqrt{125} \end{bmatrix}\begin{bmatrix} -\frac{4}{5} & -\frac{3}{5} \end{bmatrix} = \begin{bmatrix} 4 & 3 \\ 8 & 6 \end{bmatrix}$
with $\hat{U}$, $\hat{\Sigma}$, $\hat{V}^T$

Matrix norms

for a given matrix $A \in \mathbb{R}^{n \times m}$

i) $\|\cdot\|_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2} = tr(AA^T)^{1/2} = tr(A^TA)^{1/2}$

Frobenius norm

Ex.: $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$ $\quad \|A\|_F = \sqrt{1^2+2^2+3^2+4^2+5^2+6^2}$

$AA^T = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} = \begin{bmatrix} 1^2+2^2 & \cdot & \cdot \\ \cdot & 3^2+4^2 & \cdot \\ \cdot & \cdot & 5^2+6^2 \end{bmatrix} \sim tr(AA^T) = 1^2+2^2+3^2+4^2+5^2+6^2$

$tr(AA^T)^{1/2} = \sqrt{tr(AA^T)} = \|A\|_F$ //

$tr(AB) = tr(BA)$

$\|AU\|_F = tr(\underbrace{U^TA^T}_{A}\underbrace{AU}_{B})^{1/2} =$

U is orthogonal

$= tr(\underbrace{UU^T}_{I}A^TA)^{1/2} = tr(A^TA)^{1/2} = \|A\|_F$

$$\boxed{\|A\|_F = \left(\sum_{i=1}^{r} \sigma_i^2 \right)^{1/2}}$$

$\|A\|_F = \|U\Sigma V^T\|_F = \|\Sigma V^T\|_F = \|\Sigma\|_F = tr(\Sigma\Sigma^T)^{1/2} = \left(\sum_{i=1}^{r} \sigma_i^2 \right)^{1/2}$ // (orthogonal)

ii) p-norms $\qquad \vec{p} \in \mathbb{R}^n \Rightarrow \|\vec{p}\|_p = \left(\sum_{i=1}^{n} |p_i|^p \right)^{1/p}$

$\|A\|_p = \sup_{\vec{x} \in \mathbb{R}^n} \frac{\|A\vec{x}\|_p}{\|\vec{x}\|_p} = \sup_{\substack{x \in \mathbb{R}^n \\ \|\vec{x}\|=1}} \|A\vec{x}\|_p$

$p=2 \rightsquigarrow$ operator norm

$\|A\|_2 = \sigma_1$

Theorem for $\|\cdot\|_F$, $\|\cdot\|_2$, we have:

$\|A - A_k\| \leq \|A - B\|$, $\forall B$ of rank k

$A_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^T \qquad \|A - A_k\| = \begin{cases} \sigma_{k+1} & \|\cdot\|_2 \\ \left(\sum_{i=k+1}^{r} \sigma_i^2 \right)^{1/2} & \|\cdot\|_F \end{cases}$

Weyl inequality $\quad \sigma_{i+j-1}(X+Y) \leq \sigma_i(X) + \sigma_j(Y)$

$X = A - B \qquad Y = B$ (rank k)

$\boxed{\sigma_{i+k}(A) \leq \sigma_i(A-B)} + \sigma_{k+1}(B)$ $\quad$ 0 → position $k+1$ of a rank k matrix is $\emptyset$

$\|A - A_k\|_F^2 = \left(\sum_{i=k+1}^{r} \sigma_i^2(A) \right) = \left(\sum_{i=1}^{r-k} \sigma_{i+k}^2(A) \right) \leq \left(\sum_{i=1}^{r-k} \sigma_i^2(A-B) \right) \leq \left(\sum_{i=1}^{\min(m,n)} \sigma_i^2(A-B) \right) = \|A-B\|_F^2$

$A = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$

$A_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^T$

$\|\cdot\|_2$

B $(n \times d)$, $rank(B) = k$

$N(B) \subset \mathbb{R}^d \qquad \dim(N(B)) = d - k$

$V_{k+1} = \begin{bmatrix} | & & | \\ \vec{v}_1 & \cdots & \vec{v}_{k+1} \\ | & & | \end{bmatrix}$ First $k+1$ columns of V

$C(V_{k+1}) \subset \mathbb{R}^d \qquad \dim(C(V_{k+1})) = k+1$

$\dim(N(B)) + \dim(C(V_{k+1})) = d - k + k + 1 = \underline{d+1}$ ↳ since both $N(B)$ and $C(V_{k+1}) \subset \mathbb{R}^d$, they intersect

$\vec{w} \in N(B) \cap C(V_{k+1}) \qquad \|\vec{w}\|_2 = 1$

$\vec{w} = \sum_{i=1}^{k+1} c_i v_i = \qquad \sum_{i=1}^{k+1} c_i^2 = 1$

$= V_{k+1}\vec{c}$

$\|A - B\|_2^2 = \sup_{\|\vec{w}\|=1} \|(A-B)\vec{w}\|_2^2 \geq \|(A-B)\vec{w}\|_2^2$

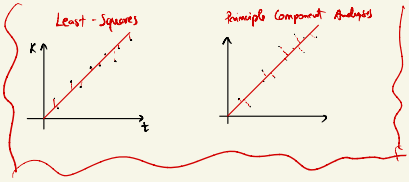
$B\vec{w} = \vec{0} \Rightarrow \|A-B\|_2^2 \geq \|A\vec{w}\|_2^2 = \vec{w}^TA^TA\vec{w} = w^TV\Sigma^TU^TU\Sigma V^T\vec{w} = \vec{w}^TV\Sigma^T\Sigma V^T\vec{w} =$

$= \vec{c}^T V_{k+1}^T V_{k+1} \Sigma^T\Sigma V_{k+1}^T V_{k+1} \vec{c} = \vec{c}^T\Sigma^T\Sigma\vec{c} = \sum_{i=1}^{k+1} c_i^2\sigma_i^2 \geq \sigma_{k+1}^2 \underbrace{\sum_{i=1}^{k+1} c_i^2}_{=1} = \sigma_{k+1}^2$

↳ $\|A-B\|_2^2 \geq \sigma_{k+1}^2 = \|A - A_k\|_2^2 \Rightarrow \|A-B\|_2^2 \geq \|A-A_k\|_2^2$, $\forall_B$ of rank(k)

PCA

↳ Project the dataset in a new space where
- ↳ variance is maximized
- ↳ covariance is minimized
- ↳ dimensionality reduction of the dataset



i) Center the matrix

$\bar{A}$ is mean centered with respect to columns

$H = I_n - \frac{1}{n}\vec{1}_n\vec{1}_n^T$ ↳ centering matrix ↳ ones

$\bar{A} = HA$

ii) Build the covariance matrix

$S = \frac{\bar{A}^T\bar{A}}{n-1}$ ⇒ S is spd, thus $SV = VD$ holds

$SV = VD \Rightarrow S = VDV^T, D = V^TSV$ → vectors in V are the principal components

$\bar{A} = U\Sigma V^T$

$S = \frac{1}{n-1}\bar{A}^T\bar{A} = \frac{1}{n-1}V\Sigma^T\underbrace{U^TU}_{I}\Sigma V^T = \frac{1}{n-1}V\Sigma^T\Sigma V^T = \frac{1}{n-1}V\Sigma^2V^T$

$D = \frac{1}{n-1}\Sigma^2 \rightsquigarrow \lambda_i = \frac{\sigma_i^2}{n-1}$

Image Compression

$A = A_{true} + \gamma A_{noise}$ ← Gaussian noise with 0 mean, unit variance

τ threshold $\sigma_i > \tau$

When γ is known:

i) $A \in \mathbb{R}^{m\times n}$ $\tau = \frac{4}{\sqrt{3}}\sqrt{n}\,\gamma$

ii) $A \in \mathbb{R}^{m\times n}$ $\lambda(\beta) = \left(2(\beta+1) + \frac{8\beta}{(\beta+1)(\beta^2+14\beta+1)}\right)^{1/2}$

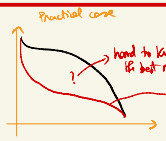
ii.a) $m \ll n$ $\beta = \frac{m}{n}$

ii.b) $m \ll n$ $\beta = \frac{n}{m}$

$\tau = \lambda(\beta)\sqrt{n}\,\gamma$

When γ is not known:

$\tau = \omega(\beta)\,\tilde{\sigma}_{med}$ $\omega(\beta) = \lambda(\beta)/\mu_\beta$, where $\mu_\beta = \int_{(1-\beta)^2}^{u_\beta} \frac{\left(\left((1+\sqrt{\beta})^2 - t\right)\left(t - (1-\sqrt{\beta})^2\right)\right)^{1/2}}{2\pi t}\,dt$



$\mathbb{E}[\|A - QB\|_2] \le \left(1 + \sqrt{\frac{k}{p-1}} + \frac{e\sqrt{k+p}}{p}\sqrt{n-k}\right)\sigma_{k+1}$

Least Squares Approximation (LS)

training set

$X \in \mathbb{R}^{n \times p}$ $\quad n \Rightarrow$ # samples, $p \Rightarrow$ # features

$\text{rank}(X) = p$ (features are independent)

$\vec{y} \in \mathbb{R}^n \Rightarrow$ "labels" of each sample

$X = \begin{bmatrix} \vec{x}_1^T \\ \vec{x}_2^T \\ \vdots \\ \vec{x}_n^T \end{bmatrix}$ (p features)

$\vec{x}_j \Rightarrow j^{th}$ column of matrix X (feature column)

Goal

$\vec{x} \rightsquigarrow \tilde{y}$

$y = \vec{x}^T \cdot \vec{\omega}$ ← weight vector

$\vec{y} \neq X\vec{\omega}$, instead $\hat{\vec{y}} = X\vec{\omega}$ ← model, approximation

$\Rightarrow \hat{\vec{y}} \in C(X)$, $\vec{y} \notin C(X)$

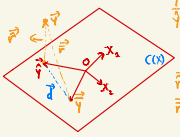
$r_i(\vec{\omega}) = y_i - \hat{y}_i(\vec{\omega}) \rightsquigarrow \vec{r}(\vec{\omega}) = \vec{y} - \hat{\vec{y}}(\vec{\omega})$

(y_i: true value, $\hat{y}_i$: predicted value)

Minimize $\|\vec{r}(\vec{\omega})\|_2^2 \Rightarrow \vec{\omega} = \underset{\vec{\omega}}{\text{argmin}} \|\vec{r}(\vec{\omega})\|_2^2$

Approaches to minimize $\|\vec{r}(\vec{\omega})\|_2^2$

1) Geometric Approach



$\hat{\vec{y}} \in C(X)$

$\vec{y} \in C(X)$

$\vec{r} = \vec{y} - \hat{\vec{y}}$

$\vec{r} = \vec{y} - \vec{d}$

$\rightsquigarrow \|\vec{r}\|_2^2 = \|\vec{r}\|_2^2 + \|\vec{d}\|_2^2$

$\|\vec{r}\|_2^2 > \|\vec{r}\|_2^2$

$\vec{r} \perp C(X)$

$X_j^T \vec{r} = 0$, $j \in [1, p]$

$\Downarrow$

$X^T \vec{r} = \vec{0}$

$X^T(\vec{y} - X\hat{\vec{\omega}}) = \vec{0} \Rightarrow X^T\vec{y} = X^TX\hat{\vec{\omega}} \rightsquigarrow \hat{\vec{\omega}} = (X^TX)^{-1}X^T\vec{y}$

(X^TX: full rank, then invertible; $(X^TX)^{-1}$: computationally demanding, SVD facilitates this calculation)

Rem. $y_i \in \{-1, 1\}$, $i = 1, \dots, n$ $\quad \hat{\vec{\omega}} = (X^TX)^{-1}X^T\vec{y}$

$\hat{\vec{y}} = X\hat{\vec{\omega}} \rightsquigarrow \hat{\vec{y}} = \text{sign}(\hat{\vec{y}})$ ↝ classification problems tend to require pre-processing of the results

2) Optimization

$\hat{\vec{\omega}} = \underset{\vec{\omega}}{\text{argmin}} \|\vec{r}(\vec{\omega})\|_2^2 = \quad \vec{r}(\vec{\omega}) = \vec{y} - X\vec{\omega}$

$= \underset{\vec{\omega}}{\text{argmin}} (\vec{y} - X\vec{\omega})^T(\vec{y} - X\vec{\omega}) =$

$= \underset{\vec{\omega}}{\text{argmin}} \; \vec{y}^T\vec{y} - (X\vec{\omega})^T\vec{y} - \vec{y}^TX\vec{\omega} + (X\vec{\omega})^TX\vec{\omega} =$

$= \underset{\vec{\omega}}{\text{argmin}} \; \vec{y}^T\vec{y} - 2\vec{\omega}^TX^T\vec{y} + \vec{\omega}^TX^TX\vec{\omega} = F(\vec{\omega})$

X is full rank

X^TX is positive definite

$\nabla_{\vec{\omega}} F(\vec{\omega}) = \vec{\emptyset}$

$\nabla_{\vec{\omega}} F(\vec{\omega}) = -2X^T\vec{y} + 2X^TX\vec{\omega} = \vec{0} \Rightarrow X^T\vec{y} = X^TX\vec{\omega}$

thus $\hat{\vec{\omega}} = (X^TX)^{-1}X^T\vec{y}$

From the deductions, we have

$\hat{\vec{y}} = \underbrace{X}_{(n \times p)} \underbrace{(X^TX)^{-1}X^T}_{(p \times n)} \vec{y}$ $\quad P_X = P_X^2$, thus a projection matrix $+ \; P_X = P_X^T$

Considering a matrix U, which is orthogonal and contains a basis for $C(X)$

↳ $U^TU = I$

$\hat{\vec{y}} = X\hat{\vec{\omega}} = U\hat{\vec{\omega}} \rightsquigarrow C(X) = C(U)$ ↳ only U^TU, not UU^T

(change of basis to a orthonormal basis)

$\hat{\vec{\omega}} = \underset{\vec{\omega}}{\text{argmin}} \|\vec{y} - U\vec{\omega}\|_2^2 \Rightarrow \hat{\vec{y}} = \underbrace{U(U^TU)^{-1}U^T}_{P_U}\vec{y} =$ but U is orthogonal

$= UIU^T\vec{y} = UU^T\vec{y}$

$\hat{\vec{\omega}} = U^T\vec{y}$

$X = \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 0 & 0 \end{bmatrix}$ $\quad \vec{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$ $\quad \vec{x}_2 = \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix}$

$\vec{u}_1 = \frac{\vec{x}_1}{\|\vec{x}_1\|} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}$

$\vec{x}_2' = \vec{x}_2 - (\vec{x}_2^T\vec{u}_1)\vec{u}_1$ ← remove the component of $\vec{x}_2$ in $\vec{u}_1$ (projection)

$= \vec{x}_2 - (\vec{u}_1^T\vec{x}_2)\vec{x}_2$

$\vec{u}_2 = \frac{\vec{x}_2'}{\|\vec{x}_2'\|} = \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}$

$\rightsquigarrow U = \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \\ 0 & 0 \end{bmatrix} \rightsquigarrow U^TU = I$, $UU^T \neq I$

Drawbacks

↳ U does no meaning

$X = U\Sigma V^T$

$\hat{\vec{\omega}} = (X^TX)^{-1}X^T\vec{y} =$

$= (V\Sigma^TU^TU\Sigma V^T)^{-1}V\Sigma^TU^T\vec{y} =$

$= (V\underbrace{\Sigma^T\Sigma}_{A}\underbrace{V^T}_{B})^{-1}V\Sigma^TU^T\vec{y} =$ $\quad (AB)^{-1} = B^{-1}A^{-1}$

$= V(V\Sigma^T\Sigma)^{-1}V\Sigma^TU^T\vec{y} =$ $\quad V^TV = I \Rightarrow V^T = V^{-1}$, thus $(V^T)^{-1} = V$

$= V(\Sigma^T\Sigma)^{-1}\Sigma^TU^T\vec{y} =$

$= V(\Sigma^T\Sigma)^{-1}\Sigma^TU^T\vec{y}$ (*)

Σ^+ ↝ pseudo-inverse of Σ $\rightsquigarrow \hat{\vec{y}} = X(X^TX)^{-1}X^T\vec{y} = X\underbrace{X^+}\vec{y}$

(*) $\hat{\vec{\omega}} = V\Sigma^+U^T\vec{y}$

$U^{(n \times n)}\,\Sigma^{(n \times p)}\,V^{T\,(p \times p)}$ $\quad (n \geq p)$ → assumption that you have more samples than features

$\Sigma = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_p \\ & 0 & \end{bmatrix}$

$\Sigma^+ = (\Sigma^T\Sigma)^{-1}\Sigma^T$

$\begin{matrix} p \\ p \end{matrix}\begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_p \end{bmatrix}\begin{matrix} n \\ p \end{matrix}\begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_p \\ 0 & & 0 \end{bmatrix} = \begin{matrix} p \\ p \end{matrix}\begin{bmatrix} \sigma_1^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_p^2 \end{bmatrix}$

$(\Sigma^T\Sigma)^{-1} \Rightarrow \begin{matrix} p \\ p \end{matrix}\begin{bmatrix} 1/\sigma_1^2 & & 0 \\ & \ddots & \\ 0 & & 1/\sigma_p^2 \end{bmatrix}$ $\quad (\Sigma^T\Sigma)^{-1}\Sigma^T = \begin{matrix} p \\ p \end{matrix}\left[\begin{array}{ccc|c} 1/\sigma_1^2 & & 0 & \\ & \ddots & & 0 \\ 0 & & 1/\sigma_p^2 & \end{array}\right]\begin{matrix} p \\ p \end{matrix}\left[\begin{array}{ccc|c} \sigma_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & \sigma_p & \end{array}\right]\begin{matrix} n \\ p \end{matrix} =$

$\Sigma^+ = \begin{matrix} p \\ p \end{matrix}\left[\begin{array}{ccc|c} 1/\sigma_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & 1/\sigma_p & \end{array}\right]$ $\quad (p \times n)$

Assuming that $p \geq n$ in $X^{(n \times p)}$, X has n LI rows

$X = [\ \ ]_{n \times p} \Rightarrow$ ∞ solutions for $\vec{w}$

$\vec{y} = X\vec{w}$

$\vec{w} = V \Sigma^{\dagger} U^T \vec{y} \rightarrow \Sigma^{\dagger} = \Sigma^T (\Sigma\Sigma^T)^{-1}$

$\hookrightarrow \Sigma^{\dagger} = \begin{bmatrix} 1/\sigma_1 & & 0 \\ & \ddots & \\ 0 & & 1/\sigma_n \end{bmatrix}$

$\vec{w}$, $\hat{w}$ $\quad \vec{y} = X\vec{w} = X\hat{w}$

$\|\vec{w}\|_2^2 = \|\vec{w} - \vec{s}\|_2^2 + \|\vec{s}\|_2^2 - 2(\vec{w} - \vec{s})^T \vec{s} \Rightarrow \|\vec{w}\|_2^2 \geq \|\vec{s}\|_2^2$

$\|\vec{w} - \vec{s}\|_2^2 \geq 0$

for any two vectors $\vec{w}$ and $\vec{s}$, $\vec{s}$ will always have the minimum norm

$(\vec{w} - \vec{s})^T \vec{s} = (\vec{w} - \vec{s})^T X^T (XX^T)^{-1} \vec{y} =$

$= (X(\vec{w} - \vec{s}))^T (XX^T)^{-1} \vec{y} =$

$= (X\vec{w} - X\vec{s})^T (XX^T)^{-1} \vec{y} = \vec{0}$ ($X\vec{s} = \vec{y}$)

$X = \begin{bmatrix} 1 & 0 & 0.01 \\ 0 & 1 & 0 \end{bmatrix} \quad \vec{y} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$\vec{w_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ ↳ choosing 1st column $\quad \vec{w_2} = \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix} \in$ amplifies errors ↳ choosing 3rd column

> reliable

Matrix Completion

$X \in \mathbb{R}^{n \times p}$ $\quad \text{rank}(X) = r << \min(n, p)$ ⇝ there exists a low-rank representation of rank r

X_{ij} for $(i,j) \in \Omega$ $\quad X = \begin{bmatrix} z_1 & & & \\ & z_2 & & \\ & & \ddots & \\ z_3 & z_4 & & z_5 \end{bmatrix}$ $z_i \Rightarrow$ known values

"Ideal" estimator — Non-convex problem

$\hat{X} = \underset{Z \in \mathbb{R}^{n \times p}}{\text{argmin}} \ \text{rank}(Z)$, subject to $\hat{X}_{ij} = X_{ij}$, $(i,j) \in \Omega$

$\hat{X} = X$, with $p = 1$

"Practical" estimator — $\|Z\|_* = \sum_{i=1}^{\min(n,p)} \sigma_i$ (nuclear norm) — convex-problem

$\hat{X} = \underset{Z \in \mathbb{R}^{n \times p}}{\text{argmin}} \ \|Z\|_*$, subject to $\hat{X}_{ij} = X_{ij}$, $(i,j) \in \Omega$

↳ the result is very close to the ideal one

SVT (Singular Value Thresholding) — non-monotone algorithm

in practice applied in parallel (SVD is heavy)

- Initialize $\hat{X} = \text{zeros}(n, p)$
- Set $\hat{X}_{ij} = X_{ij}$ for $(i,j) \in \Omega$
- For $k = 1, 2, \dots, N$
  - $\hat{X}_{old} = \hat{X}$
  - $[U, \Sigma, V^T] = \text{SVD}(\hat{X})$
  - $\Sigma \rightarrow \hat{\Sigma}$ $\begin{cases} \sigma_i \text{ if } \sigma_i > \tau \\ 0 \text{ otherwise} \end{cases}$ (reducing the rank of the matrix)
  - $\hat{X} = U \hat{\Sigma} V^T$
  - $\hat{X}_{ij} = X_{ij}$, $(i,j) \in \Omega$
  - if $\|\hat{X} - \hat{X}_{old}\|_2 > \varepsilon$ ↳ Nonsense return

$\Sigma = \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ 0 & & & 0 \end{bmatrix} \Rightarrow \hat{\Sigma}$, where $\hat{\sigma}_i$: $\begin{cases} \sigma_i \text{ if } \sigma_i > \tau \rightarrow \text{thresholding value} \\ 0 \text{ otherwise} \end{cases}$

(x_i, y_i) , $i = 1 \dots n$

$\vec{x}_i \in \mathbb{R}^p$ $\quad y_i \in \mathbb{R}$ $\quad \vec{y} \in \mathbb{R}^n$

$X \in \mathbb{R}^{n \times p}$

Hipothesis: X has p linearly independent colums

Thus $\sigma_1 \dots \sigma_p > 0$

$\vec{y} = X\vec{\omega}_{LS}$ $\quad \vec{y} \approx X\vec{\omega}_{LS}$ $\quad \vec{y} = X\vec{\omega}^* + \vec{\varepsilon}$ (noise; $\vec{\omega}^*$: weight vector of the underlying reality)

$\vec{\omega}_{LS} = (X^TX)^{-1}X^T\vec{y} =$

$= (X^TX)^{-1}X^T(X\vec{\omega}^* + \vec{\varepsilon}) =$

$= (X^TX)^{-1}(X^TX\vec{\omega}^* + X^T\vec{\varepsilon}) =$

$= (X^TX)^{-1}(X^TX)\vec{\omega}^* + (X^TX)^{-1}X^T\vec{\varepsilon} =$

$= \vec{\omega}^* + (X^TX)^{-1}X^T\vec{\varepsilon}$

$X = U\Sigma V^T$

$(X^TX)^{-1}X^T = V\Sigma^+U^T$

$\Sigma^+ = \begin{bmatrix} 1/\sigma_1 & & & \\ & 1/\sigma_2 & & \\ & & \ddots & \\ & & & 1/\sigma_p \end{bmatrix} \Big| \; 0$

$\sigma_p \simeq 10^{-9}$ (singular, same direction is not explained independently)

$(X^TX)^{-1}X^T\vec{\varepsilon} = V\Sigma^+U^T\vec{\varepsilon}$ → magnifies the error by factor 10^9, $\vec{\omega}_{LS} \neq \vec{\omega}^*$ very different

Ridge Regression

$\vec{\omega}_R = \underset{\vec{\omega}}{\operatorname{argmin}} \{ \|\vec{y} - X\vec{\omega}\|_2^2 + \lambda\|\vec{\omega}\|_2^2 \}$ (least squares + regularization term)

$f(\vec{\omega}) = \|y - X\vec{\omega}\|_2^2 + \lambda\|\vec{\omega}\|_2^2 = \vec{y}^T\vec{y} - 2\vec{\omega}^TX^T\vec{y} + \vec{\omega}^TX^TX\vec{\omega} + \lambda\vec{\omega}^T\vec{\omega}$

$\nabla f(\vec{\omega}) = -2X^T\vec{y} + 2X^TX\vec{\omega} + 2\lambda\vec{\omega} = 0$

$X^T\vec{y} = (X^TX + \lambda I)\vec{\omega}$

$\vec{\omega}_R = (X^TX + \lambda I)^{-1}X^T\vec{y}$

$\vec{\omega}_R = (X^TX + \lambda I)^{-1}X^T\vec{y} =$

$= (X^TX + \lambda I)^{-1}X^T(X\vec{\omega}^* + \vec{\varepsilon}) =$

$= (X^TX + \lambda I)^{-1}X^TX\vec{\omega}^* + (X^TX + \lambda I)^{-1}X^T\vec{\varepsilon}$

if $\vec{\varepsilon} \approx \vec{0}$, $\vec{\omega}_R$ is a little bit worse than $\vec{\omega}_{LS}$

However, if not, $\vec{\omega}_R$ will regularize small singular values, which $\vec{\omega}_{LS}$ does not

$X = U\Sigma V^T$

$\vec{\omega}_R = (V\Sigma^TU^TU\Sigma V^T + \lambda VV^T)^{-1}V\Sigma^TU^T\vec{y} =$

$= \left[V(\Sigma^TU^TU\Sigma + \lambda I)V^T\right]^{-1}V\Sigma^TU^T\vec{y} =$ ($V^{-1} = V^T$)

$= V(\Sigma^T\Sigma + \lambda I)^{-1}V^TV\Sigma^TU^T\vec{y} =$

$= V(\Sigma^T\Sigma + \lambda I)^{-1}\Sigma^TU^T\vec{y}$

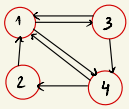
$\Sigma^+ = \begin{bmatrix} 1/\sigma_1 & & & \\ & 1/\sigma_2 & & \\ & & \ddots & \\ & & & 1/\sigma_p \end{bmatrix} \Big| \; 0$

$(\Sigma^T\Sigma + \lambda I)\Sigma^+ = \begin{bmatrix} \frac{\sigma_1}{\sigma_1^2 + \lambda} & & & \\ & \frac{\sigma_2}{\sigma_2^2 + \lambda} & & \\ & & \ddots & \\ & & & \frac{\sigma_p}{\sigma_p^2 + \lambda} \end{bmatrix} \Big| \; 0$

if $\sigma_i \gg \lambda$, diagonal terms are $\propto \frac{1}{\sigma_i}$

if $\sigma_i \approx 0$, diagonal terms are $\simeq 0$ → λ prevents exploding terms regularizes

PAGE-RANK



Idea: Surf the web randomly (random walk on the graph)

Already visited $\vec{\pi}$, π_i: prob to be at site i

$\vec{\pi}$

$\vec{A} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$

$\vec{A}_{ij} = \begin{cases} 1 & \text{link } j \to i \\ 0 & \text{otherwise} \end{cases}$

↓ probability matrix

$A = \begin{bmatrix} 0 & 1 & 1/3 & 1/2 \\ 0 & 0 & 1/3 & 1/2 \\ 1/2 & 0 & 0 & 0 \\ 1/2 & 0 & 1/3 & 0 \end{bmatrix}$

$A\vec{e}_3 = \begin{bmatrix} 1/3 \\ 1/3 \\ 0 \\ 1/3 \end{bmatrix}$ $\quad \vec{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$

$\vec{\pi} = \begin{bmatrix} \pi_1 \\ \pi_2 \\ \pi_3 \\ \pi_4 \end{bmatrix}$

↓ look for $A\vec{\pi} = \vec{\pi}$

$A > 0$, Perron-Frobenius → $\lambda = 1 > \lambda_2 > \lambda_3 > \dots > \lambda_n$

Power-Method

$\vec{\pi}^{(0)}$ $\quad \|\vec{\pi}^{(0)}\| = 1$

for $k = 1, 2, \dots$

$\vec{\pi}^{(k)} = \dfrac{A\vec{\pi}^{(k-1)}}{\|A\vec{\pi}^{(k-1)}\|}$

stop criterion $\|\vec{\pi}^{(k)} - \vec{\pi}^{(k-1)}\| < \varepsilon$

$\vec{\pi}^{(0)}$ $\quad \vec{z}^{(1)} = A\vec{\pi}^{(0)}$ $\quad \vec{\pi}^{(1)} = \dfrac{\vec{z}^{(1)}}{\|\vec{z}^{(1)}\|}$ $\quad \vec{\pi}^{(2)} = \dfrac{A\vec{\pi}^{(1)}}{\|A\vec{\pi}^{(1)}\|} = \dfrac{A^2\vec{\pi}^{(0)}}{\|A^2\vec{\pi}^{(0)}\|}$, thus $\vec{\pi}^{(k)} = \dfrac{A^k\vec{\pi}^{(0)}}{\|A^k\vec{\pi}^{(0)}\|}$

$\vec{\pi}^{(k)} = A^k\vec{\pi}^{(0)} = V\Lambda^kV^T(\alpha_1\vec{v}_1 + \dots + \alpha_n\vec{v}_n) =$

$= V\Lambda^k(\alpha_1\vec{e}_1 + \dots + \alpha_n\vec{e}_n) = V(\alpha_1\lambda_1^k\vec{e}_1 \dots + \alpha_n\lambda_n^k\vec{e}_n) =$

$= \alpha_1\lambda_1^k\vec{v}_1 + \dots + \alpha_n\lambda_n^k\vec{v}_n = \alpha_1\lambda_1^k\left(\vec{v}_1 + \frac{\alpha_2}{\alpha_1}\left(\frac{\lambda_2}{\lambda_1}\right)^k\vec{v}_2 + \dots + \frac{\alpha_n}{\alpha_1}\left(\frac{\lambda_n}{\lambda_1}\right)^k\vec{v}_n\right) =$

(λ_1 since $\lambda_1 > \lambda_2$, $\left(\frac{\lambda_2}{\lambda_1}\right)^k \to 0$ when $k \to \infty$)

$= \alpha_1\lambda_1^k\vec{v}_1 \Rightarrow$ converges

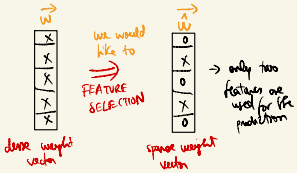
A can be diagonalized, $\vec{v}_i$ $\quad i = 1, \dots, n$

$\hookrightarrow A = V\Lambda V^T$ $\quad A\vec{v}_i = \lambda_i\vec{v}_i$

$\vec{\pi}^{(0)} = \sum_{i=1}^{n}\alpha_i\vec{v}_i$

$A^k\vec{\pi}^{(0)} = \alpha_1\lambda_1^k\left(\vec{v}_1 + \sum_{i=2}^{n}\frac{\alpha_i}{\alpha_1}\left(\frac{\lambda_i}{\lambda_1}\right)^k\vec{v}_i\right)$

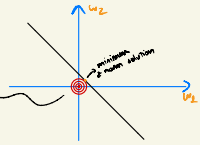
$\vec{y} = X\vec{w}$ For LS and RR, $\vec{w} \in \mathbb{R}^p$ and is dense



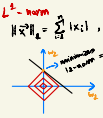
$\vec{y} = X\vec{w}$ $X \in \mathbb{R}^{n \times p}$, $p \gg n$

Example:
2 features, 1 sample ($n=1, p=2$)

$X = [2\ 3]$ $y = [1]$ $\rightsquigarrow$ $1 = [2\ 3]\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \Rightarrow \boxed{1 = 2w_1 + 3w_2}$ line (infinite solutions)



L^2 - norm
$\|\vec{x}\|_2 = \sum_{i=1}^{n} |x_i|^2$, $\vec{x} \in \mathbb{R}^n$



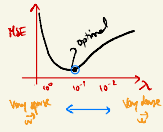
$\Rightarrow$ more sparse solution

$F(\omega) = \|X\vec{w} - \vec{y}\|_2^2 + \lambda\|\vec{w}\|_1$

LASSO (Least Absolute and Selection Operator)

L^1-norm promotes sparsity and may prevent overfitting

also easier interpretability



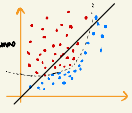
Elastic Net

$F(\vec{w}) = \|X\vec{w} - \vec{y}\|_2^2 + \lambda_1\|\vec{w}\|_1 + \lambda_2\|\vec{w}\|_2^2$

$\vec{y} = X\vec{w}$

$X \in \mathbb{R}^{n \times p}$, p LI columns ($c_i \neq 0$, $i = 1, \dots, p$)

$\vec{y} \in \mathbb{R}^n$



$\vec{w}_R = V(\Sigma^T\Sigma + \lambda I)^{-1}\Sigma^T U^T \vec{y}$ Ridge Regression

$(\Sigma^T\Sigma + \lambda I)^{-1}\Sigma^T = \Sigma^T(\Sigma\Sigma^T + \lambda I)^{-1}$

$\vec{w}_R = V\Sigma^T(\Sigma\Sigma^T + \lambda I)^{-1}U^T\vec{y} =$

$= V\Sigma^T U^T U(\Sigma\Sigma^T + \lambda I)^{-1}U^T\vec{y} =$

$= X^T\alpha$ $\alpha \in \mathbb{R}^n$

$\vec{w}_R = X^T\alpha = \sum_{i=1}^{n} \alpha_i \vec{x}_i$ $\hat{y}_i = \vec{x}_i^T\vec{w} = \boxed{w_1 x_{i1} + w_2 x_{i2}}$

adding features

$\hat{y}_i = w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i1}^2 + w_4 x_{i2}^2 + w_5 x_{i1}x_{i2}$

Using $\varphi(\vec{x}_i)$, m feature map

$\varphi(\vec{x}_i) = \begin{bmatrix} x_{i1} \\ x_{i2} \\ x_{i1}^2 \\ x_{i2}^2 \\ x_{i1}x_{i2} \end{bmatrix}$ $\mathbb{R}^d$ $d > p$ $\hat{y}_i = \varphi(\vec{x}_i)^T\vec{w}$

Kernel method: aim of the method is to avoid the necessity of computing $\Sigma\Sigma^T$ when φ is used

$\varphi \begin{bmatrix} -\varphi(\vec{x}_1)^T- \\ -\varphi(\vec{x}_2)^T- \end{bmatrix}$ $\varphi \in \mathbb{R}^{n \times d}$

$\vec{w}_R = \varphi^T\vec{\alpha}$ $\vec{\alpha} = U(\Sigma\Sigma^T + \lambda I)^{-1}U^T\vec{y} =$

$= (XX^T + \lambda I)^{-1}\vec{y}$

$\vec{\alpha} = (\varphi\varphi^T + \lambda I)^{-1}\vec{y}$ $\rightarrow$ how can we reduce the cost of computing $\vec{\alpha}$

$K \in \mathbb{R}^{n \times n}$

$K_{ij} = \varphi(\vec{x}_i^T)\,\varphi(\vec{y}_j^T) = K(x_i, y_j)$

$\alpha = (K + \lambda I)^{-1}\vec{y}$ $\rightarrow$ How to predict a label for a new feature vector

$\hat{y} = \varphi(\vec{x}^T)\,\vec{w}_R^T\,\varphi(\vec{x}) =$

$= (\varphi^T\alpha)\,\varphi(\vec{x}^T) = \alpha^T\varphi\,\varphi(\vec{x}^T) = \sum_{i=1}^{n} \alpha_i\,\varphi(x_i)^T\varphi(\vec{x}^T) = \sum_{i=1}^{n} \alpha_i K(x_i, x)$

Kernel trick K allows to create non-linear models by achieving the feature map by achieving the feature map without having to extract the computationally heavy features

Kernel Function

$K(\vec{x}_i, \vec{y}_j) = \vec{x}_i^T\vec{y}_j$

$\vec{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix}$ $\varphi(\vec{x}_i) = \begin{bmatrix} x_{i1}^2 \\ \sqrt{2}x_{i1}x_{i2} \\ x_{i2}^2 \end{bmatrix}$ Maps this feature map in a faster and cheaper way

$\varphi(\vec{x}_i)^T\varphi(\vec{x}_j^T) = x_{i1}^2x_{j1}^2 + 2x_{i1}x_{i2}x_{j1}x_{j2} + x_{i2}^2x_{j2}^2$ (*)

$K(\vec{x}_i, \vec{y}_j) = (\vec{x}_i^T\vec{y}_j^T)^2 =$

$= (x_{i1}y_{j1} + x_{i2}y_{j2})^2 =$ (**)

i) Polynomials of degree q
$K(\vec{x}_i^T, \vec{y}_j^T) = (\vec{x}_i^T\vec{y}_j^T)^q$

ii) Polynomials of degree $\leq q$
$K(\vec{x}_i^T, \vec{y}_j^T) = (\vec{x}_i^T\vec{y}_j^T + 1)^q$

iii) Gaussian function
$K(\vec{x}_i, \vec{y}_j^T) = \exp\left(-\frac{\|\vec{x}_i^T - \vec{y}_j^T\|_2^2}{2\sigma}\right)$

$X \in \mathbb{R}^{n \times p}$ (already centered)
n # samples
p # features

$C = \frac{1}{n-1} X^T X$ ⇒ s.p.d, thus factorizable as

$C = V \Lambda V^T$ — V: eigenvectors, principal directions; Λ: eigenvalues, measures of variance along each direction

$\Phi = XV$ → principal components

$X = U \Sigma V^T$

$C = \frac{1}{n-1} V \Sigma^T U^T U \Sigma V^T = \frac{1}{n-1} V \Sigma^2 V^T$

$\lambda_i = \frac{1}{n-1} \sigma_i^2$

$\Phi = XV = U \Sigma V^T V = U \Sigma$

$X \in \mathbb{R}^{p \times n}$ ⇒ you have to interchange the role of U and V
p # features
n # features

$C = \frac{1}{n-1} X X^T = \frac{1}{n-1} U \Sigma^2 U^T$

$\Phi = X^T U$

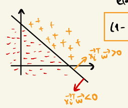
$\hat{\vec{w}}_R = \underset{\vec{w}}{\operatorname{argmin}} \|y - \Phi \vec{w}\|_2^2 + \lambda \|\vec{w}\|_2^2$

$(\vec{x_i}, y_i)$ ⇒ binary classification problem $y_i \in \{-1, 1\}$, $x_i \in \mathbb{R}^p$

$\hat{\vec{w}}_{LS} = \underset{\vec{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - x_i^T \vec{w})^2 = \underset{\vec{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (1 - y_i x_i^T \vec{w})^2$ if $y_i = 1$

$\hat{\vec{w}}_{LS} = \underset{\vec{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (-1 - x_i^T \vec{w})^2 = \underset{\vec{w}}{\operatorname{argmin}} \sum_{i=1}^{n} (1 + x_i^T \vec{w})^2$ if $y_i = -1$

Ex $y_i = 1$ $\vec{x_i}^T \vec{w} = 10$, for some $\vec{w}$ classified correctly



$(1 - 1 \cdot 10)^2 = 81$ ⇒ changing the loss function

1) $L_i(y_i, \vec{x_i}^T \vec{w}) = (1 - y_i \vec{x_i}^T \vec{w})^2$ square

2) $L_i(y_i, \vec{x_i}^T, \vec{w}) = \begin{cases} 0 & y_i \vec{x_i}^T \vec{w} > 0 \\ 1 & y_i \vec{x_i}^T \vec{w} < 0 \end{cases}$

3) $L_i(y_i, \vec{x_i}^T, \vec{w}) = (1 - y_i \vec{x_i}^T \vec{w})_+$ $(a)_+ = \begin{cases} a & \text{if } a > 0 \\ 0 & \text{otherwise} \end{cases}$

Derivative calculation

	Pros	Cons
Manual	Exact, Good for proofs	Prone to error; Expensive for complex functions
Numerical Differentiation	Easy to program	FP precision; Computational cost
Automatic Differentiation	Exact; Fast	Implementation

Automatic Differentiation

→ Representing any function as a combination of simple functions (sum, product, trigonometric)

Wengert list $f: \mathbb{R}^n \to \mathbb{R}^m$

$v_{i-n} = x_i$	$i = 1 \dots n$	INPUTS
v_i	$i = 1 \dots l$	INTERMEDIATE
$y_{m-i} = v_{l-i}$	$i = m-1 \dots 0$	OUTPUT

$y = f(x_1, x_2) = \sin((x_1 + x_2) x_2^2)$

$v_{-1} = x_1 = 1$ (suppose) INPUTS
$v_0 = x_2 = 2$

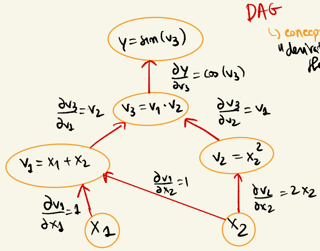
$v_1 = x_1 + x_2 = 3$
$v_2 = x_2^2 = 4$ INTERMEDIATE
$v_3 = v_1 \cdot v_2 = 12$
$v_4 = \sin(v_3) \approx 0.536$

$y = v_4$ OUTPUTS

$\dot{v}_{-1} = \dot{x}_1 = 1$ (derivative w.r.t x_1)
$\dot{v}_0 = \dot{x}_2 = 0$
$\dot{v}_1 = \dot{x}_1 + \dot{x}_2 = 1$
$\dot{v}_2 = 0$
$\dot{v}_3 = \dot{v}_1 v_2 + v_1 \dot{v}_2 = 4$
$\dot{v}_4 = \cos(v_3) \dot{v}_3 \approx 3.37$
$\dot{y} = \dot{v}_4 = 3.37$

← Forward Mode of AD

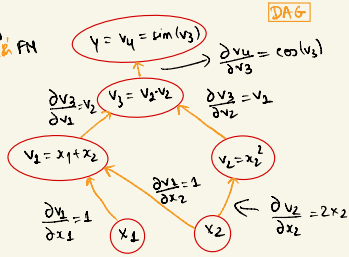
DAG
↳ concept of "derivative on the edge"



$y = f(x_1, x_2) = \sin\left((x_1 + x_2)\, x_2^2\right)$ → two elementary functions

$f: \mathbb{R}^n \to \mathbb{R}^m$ $\left(\begin{matrix} m \gg n \\ m \ll n \end{matrix}\right)$ typical for NN, issue with FM

DAG



AD is a method for evaluating the derivative

$$J = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial y_m}{\partial x_1} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix}$$

$\vec{x}' = \vec{e_i}$

$y_j' = \frac{\partial y_j}{\partial x_i}$, $j = 1, \dots, m$

$v_1 = x_1 = 1$	$v_1' = x_1' = 1$ FM	$v_1' = x_1' = 0$ FM
$v_0 = x_2 = 2$	$v_0' = x_2' = 0$ (standard mode of AD)	$v_0' = x_2' = 1$
$v_2 = x_1 + x_2 = 3$	$v_1' = x_1' + x_2' = 1$	$v_1' = x_1' + x_2' = 1$
$v_2 = x_2^2 = 4$	$v_2' = 2x_2 x_2' = 0$	$v_2' = 2x_2 x_2' = 4$
$v_3 = v_1 \cdot v_2 = 12$	$v_3' = v_1' v_2 + v_1 v_2' = 4$	$v_3' = v_1' v_2' v_2' v_2 = 16$
$v_4 = (y) = \sin(v_3)$	$v_4' = v_3' \cos(v_3)$	$v_4' = y' = \cos(v_3) v_3'$
	$\frac{\partial y}{\partial x_1}$	$\frac{\partial y}{\partial x_2}$

Similarly to Newton's Method

$J \vec{v}$: J function at given iteration

→ Krylov method ex. through → Matrix-free approach

→ able to avoid the explicit construction of the whole matrix though some tricks

→ save memory

→ way forward AD mode is employed

Dual numbers

$\{a + b\varepsilon : a, b \in \mathbb{R}, \varepsilon \neq 0, \varepsilon^2 = 0\}$

Operations

$(a + b\varepsilon) + (c + d\varepsilon) = (a + c) + (b + d)\varepsilon$

$(a + b\varepsilon) + (c + d\varepsilon) = ac + ad\varepsilon + bc\varepsilon + bd\varepsilon^2 = \phi$

$= ac + (ad + bc)\varepsilon$

$f(a + b\varepsilon) = f(a) + f'(a) b\varepsilon + \frac{f''(a)}{2} b^2 \varepsilon^2 \dots = \phi$

$= f(a) + f'(a) b\varepsilon$

if $b = 1$: $f(a + \varepsilon) = f(a) + \varepsilon f'(a)$ ✓

Ex.: $f(x) = g(x) h(x)$

$f(a + \varepsilon) = g(a + \varepsilon) h(a + \varepsilon) =$

$= (g(a) + g'(a)\varepsilon)(h(a) + h'(a)\varepsilon) =$

$= g(a)h(a) + g(a)h'(a)\varepsilon + g'(a)h(a)\varepsilon + g'(a)h'(a)\varepsilon^2 =$ ($\varepsilon^2 \to 0$)

$= g(a)h(a) + \left(g(a)h'(a) + g'(a)h(a)\right)\varepsilon$

$f(x) = g(h(x))$

$f(a + \varepsilon) = g(h(a + \varepsilon)) =$

$= g(h(a) + \varepsilon h'(a)) =$

$= g(h(a)) + g'(h(a)) h'(a) \varepsilon$

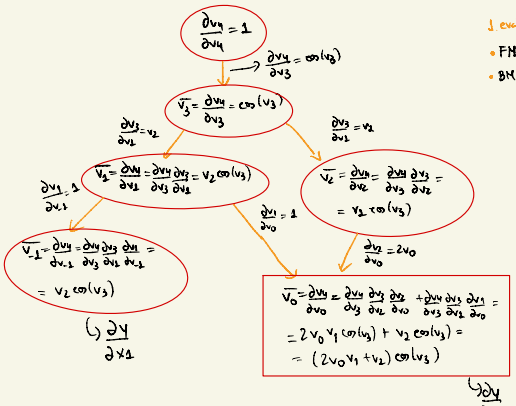
$f(x) = \frac{1}{x}$

$f(x + \varepsilon) = \frac{1}{x + \varepsilon} = \frac{x - \varepsilon}{(x + \varepsilon)(x - \varepsilon)} = \frac{x - \varepsilon}{x^2} = \frac{x}{x^2} - \frac{\varepsilon}{x^2} = \frac{1}{x} - \frac{1}{x^2}\varepsilon$

Backward (reverse) mode

↳ sensitivity of the output w.r.t. the inputs

$\bar{v_i} = \frac{\partial y}{\partial v_i}$



$f: \mathbb{R}^n \to \mathbb{R}^m$, $m \ll n$

↓ evaluation of f costs ops(f)

- FM ⇒ ~ n · c · ops(f) operations
- BM ⇒ ~ m · c · ops(f) operations

c ≈ 2.5 → choose FM or BM according to sizes of m/n

Jacobian compilation

autodiff.org

Griewank book

- FM

If $\dot{\vec{x}} = \vec{v} \Rightarrow J\dot{\vec{r}}$

- BM

if $\bar{\dot{y}} = \bar{r} \Rightarrow J^T \bar{r}$

↳ sensitivity vector

Exercise: Work the tangent dist for BM $f(x_1, x_2) = \sin((x_1+x_2)\, x_2^2)$

$v_{-1} = x_1$
$v_0 = x_2$
$v_1 = v_{-1} + v_0$
$v_2 = v_1^2$
$v_3 = \sin(v_1)$
$v_4 = v_2 \cdot v_3$

$\bar{v}_4 = \frac{\partial v_4}{\partial v_4} = 1$

$\bar{v}_3 = \frac{\partial v_4}{\partial v_3} = \cos(v_3)$

$\bar{v}_2 = \frac{\partial v_4}{\partial v_2} = \frac{\partial v_3}{\partial v_2} = \cos(v_3)\, v_2$

$\bar{v}_1 = \frac{\partial v_4}{\partial v_1} = \frac{\partial v_4}{\partial v_3}\frac{\partial v_3}{\partial v_2} = \cos(v_3)\, v_2$

$\bar{v}_0 = \frac{\partial v_4}{\partial v_0} = \frac{\partial v_4}{\partial v_3}\frac{\partial v_3}{\partial v_1}\frac{\partial v_1}{\partial v_0} + \frac{\partial v_4}{\partial v_2}\frac{\partial v_2}{\partial v_1}\frac{\partial v_1}{\partial v_0} =$

$= \cos(v_3)\cdot v_2 + \cos(v_3)\cdot v_2 \cdot 2v_0 = \cos(v_3)\left(v_2 + 2v_1 v_0\right)$

$\bar{v}_{-1} = \frac{\partial v_4}{\partial v_{-1}} = \frac{\partial v_4}{\partial v_3}\frac{\partial v_3}{\partial v_2}\frac{\partial v_2}{\partial v_{-1}} = \cos(v_3)\, v_2$

Newton's Method

$H(x) \rightsquigarrow (H\vec{v})$ ⟩ Hessian-free

useful to calculate

$f: \mathbb{R}^n \to \mathbb{R} \quad \vec{x} \to y$

reverse-on-forward

i) $\nabla f \cdot \vec{v} = \frac{\partial f}{\partial \vec{x}} \quad \vec{x}^1 = \vec{v}$ (forward step)

ii) Apply the backward mode to the result of i)

$\nabla^2 f \cdot \vec{v} = H\vec{v}$

Convolution

Functions

$f(x)$ $g(x)$

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(t)\, g(x-t)\, dt$$

Vectors

$\vec{c}$ $\vec{d}$

$$[\vec{c} * \vec{d}]_k = \sum_{i+j=k} c_i d_j$$

$c_0 + c_1 x + \dots + c_{n-1} x^{n-1} = c(x)$

$d_0 + d_1 x + \dots + d_{n-1} x^{n-1} = d(x)$

$c(x) d(x) \quad x^k \Rightarrow c_0 d_k + c_1 d_{k-1} + \dots$

sum of indices are always k

Cyclic Convolutions

$$(\vec{c} \circledast \vec{d})_k = \sum_{i+j=k \bmod n} c_i d_j$$

Convolution

Toeplitz matrix

for n×n matrix, the elements are given by a (2n − 1) length sequence

$\{t_k : -(n-1) \leq k \leq (n-1)\}$

$T(i,j) = t_{i-j}$

$$T = \begin{bmatrix} t_0 & t_{-1} & \cdots & t_{-(n-1)} \\ \vdots & t_0 & \ddots & \vdots \\ & & \ddots & t_{-1} \\ t_{n-1} & t_{n-2} & \cdots & t_0 \end{bmatrix}$$

Cyclic Convolution

Circulant Matrix

For n×n matrix, the elements are given by a n-length sequence

$\{c_k : 0 \leq k \leq n-1\}$

$c(i,j) = c_{(i-j) \bmod n}$

$$C = \begin{bmatrix} c_0 & c_{n-1} & \cdots & c_1 \\ \vdots & c_0 & \ddots & \vdots \\ & c_1 & \ddots & \\ c_{n-1} & \cdots & c_1 & c_0 \end{bmatrix}$$

ex.:
$$\begin{bmatrix} 1 & 8 & 5 & 3 \\ 3 & 1 & 8 & 5 \\ 5 & 3 & 1 & 8 \\ 8 & 5 & 3 & 1 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}}_{\vec{c}} = \begin{bmatrix} c_2 \\ c_3 \\ c_4 \\ c_1 \end{bmatrix}$$

$C = c_0 I + c_1 P + c_2 P^2 + c_3 P^3$ (polynomial in P)

$D = d_0 I + d_1 P + d_2 P^2 + d_3 P^3$

$CD = ?$ ⇒ CD is also circulant

$\hookrightarrow CD = (c_0 I + c_1 P + c_2 P^2 + c_3 P^3)(d_0 I + d_1 P + d_2 P^2 + d_3 P^3) =$

Given that it represents a cyclic convolution: $P^4 = I$, $P^5 = P$, $P^6 = P^2$ ⇝ $P^{(n \times n)} \Rightarrow P^n = I$

Example

$(1,2,1) * (3,1,2)$

⇓

$(1 + 2x + x^2)(3 + x + 2x^2) = 3 + 7x + 7x^2 + 5x^3 + 2x^4$

```
      1 2 1
    × 2 1 3
    -------
      3 6 3
    1 2 1 *
  2 4 2 * *
  ---------
  2 5 7 7 3
```

⇓

but $x^3 = 1$, $x^4 = x$

Thus

$(1,2,1) * (3,1,2) = (8, 9, 7)$

$3 + 5x^3{:}5$, $+x^2$, $+x$, $2x^4{:}2x$

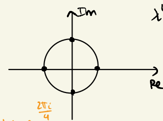
$\vec{c} = (1,2,1)$ $\vec{d} = (3,1,2)$

$$C = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix} \quad C\vec{d} = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 1 \\ 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3+1+4 \\ 6+1+2 \\ 3+2+2 \end{bmatrix} = \begin{bmatrix} 8 \\ 9 \\ 7 \end{bmatrix}$$

Eigenvalues and Eigenvectors of C

$$P = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad P - \lambda I = \begin{bmatrix} -\lambda & 1 & 0 & 0 \\ 0 & -\lambda & 1 & 0 \\ 0 & 0 & -\lambda & 1 \\ 1 & 0 & 0 & -\lambda \end{bmatrix} \quad \det(P - \lambda I) = \lambda^4 - 1$$

The eigenvals of P are the fourth roots of 1



$\lambda^4 - 1 = 0$

⇓

$\lambda_1 = 1$, $\lambda_2 = i$, $\lambda_3 = -1$, $\lambda_4 = -i$

$w = e^{\frac{2\pi i}{n}}$ → $w = e^{\frac{2\pi i}{4}}$

$\lambda_1 = w^0$, $\lambda_2 = w^1$, $\lambda_3 = w^2$, $\lambda_4 = w^3$

$P_n \to \lambda^n - 1 = 0$

$\hookrightarrow \lambda_1 = w^0$, $\lambda_2 = w^1$, ⋮, $\lambda_n = w^n$

$P^T P = I \Rightarrow$ P is orthogonal

$C = c_0 I + c_1 P + c_2 P^2 + c_3 P^3$

$\lambda_k, \vec{v}_k \rightsquigarrow \vec{v}_k$ is an eigenvector of P

$C\vec{v}_k = \lambda_k \vec{v}_k \Leftrightarrow (c_0 I + c_1 P + c_2 P^2 + c_3 P^3)\vec{v}_k = (c_0 + c_1 \lambda_k + c_2 \lambda_k^2 + c_3 \lambda_k^3)\vec{v}_k$

$\lambda_1 = 1$: $\vec{v}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ $\lambda_2 = i$: $\vec{v}_2 = \begin{bmatrix} 1 \\ i \\ i^2 \\ i^3 \end{bmatrix}$ $\lambda_3 = -1$: $\vec{v}_3 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$ $\lambda_4 = -i$: $\vec{v}_4 = \begin{bmatrix} 1 \\ -i \\ (-i)^2 \\ (-i)^3 \end{bmatrix}$

$$F = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & w & w^2 & w^3 \\ 1 & w^2 & w^4 & w^6 \\ 1 & w^3 & w^6 & w^9 \end{bmatrix}$$

$\rightsquigarrow w = e^{\frac{2\pi i}{4}}$

⇝ Fourier Matrix ⇝ Matrix involved in the Discrete Fourier Transform

↳ normalization

$$\begin{bmatrix} \lambda_0(c) \\ \vdots \\ \lambda_{n-1}(c) \end{bmatrix} = \begin{bmatrix} c_0 + c_1 \cdots c_{n-1} \\ \vdots \\ c_0 + c_1 w^{n-1} + \dots + c_{n-1} w^{(n-1)(n-1)} \end{bmatrix} = F \begin{bmatrix} c_0 \\ \vdots \\ c_{n-1} \end{bmatrix} = F\vec{c}$$

$\vec{\lambda}_c = F\vec{c}$

↳ vector of eigenvalues of C

Two matrices sharing the same eigenvectors, the eigenvalues of the products of the matrix is the product of the eigenvalues

$A\vec{v} = \lambda\vec{v}$ $B\vec{v} = \gamma\vec{v}$

$AB\vec{v} = \gamma A\vec{v} = \gamma\lambda\vec{v}$

C and D, both circulant

$\vec{c}$ $\vec{d}$

- CD in the first row you have the cyclic convolution of $\vec{c}$ and $\vec{d}$ ↳ $\vec{c} \circledast \vec{d}$
- $F(\vec{c} \circledast \vec{d})$ is the vector of eigenvalues of CD

$\lambda(C) = F\vec{c}$ $\lambda(D) = F\vec{d}$

$\lambda(CD) = \lambda(C) \cdot \lambda(D) = F\vec{c} .* F\vec{d} = \vec{\lambda}_{CD} = ((F\vec{c})_1 \cdot (F\vec{d})_1, (F\vec{c})_2 \cdot (F\vec{d})_2, \dots)$

↳ dot product ↳ component wise product

$\vec{F}(\vec{c} \circledast \vec{d}) = F\vec{c} .* F\vec{d}$

Convolution rule

- FFT nlogn operations

$F(\vec{c} \circledast \vec{d}) = n^2 + n\log n$

$F\vec{c} .* F\vec{d} = 2n\log n + n$ ←

Optimization

$w^* = \arg\min_{w} J(w)$

Gradient Descent

$\Delta J \approx \nabla J \cdot \Delta w$ — increment $\Delta w = -\eta \nabla J$, $\eta > 0$

$\Rightarrow \Delta J \approx -\eta \|\nabla J\|^2$

$w^{(k+1)} = w^{(k)} - \eta \nabla J(w^{(k)})$

$x^{(k+1)} = x^{(k)} - \eta \nabla f(x^{(k)})$, where f is the function to be minimized → x^* is the minimum; η = learning rate, assuming η is fixed for now, $x^{(0)}$ is given

↳ if it is too small: slow convergence
↳ if it is too big: overshooting (moving too far from the minimum)

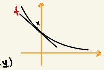
Convergence of Gradient Descent

Def: $f: \text{dom}(f) \to \mathbb{R}$ is convex if

i) dom(f) is convex

ii) $\forall x, y \in \text{dom}(f)$, $\lambda \in [0,1]$

$f(\lambda x + (1-\lambda) y) \leq \lambda f(x) + (1-\lambda) f(y)$



Lemma: dom(f) is open and f is differentiable, then f is convex iff dom(f) is convex and $f(y) \geq f(x) + \nabla f(x)^T (y - x)$, $\forall x, y \in \text{dom}(f)$ — above tangent hyperplane at y

Remembering Least Squares

$\|v - w\|^2 = (v - w)^T (v - w) =$

$= \|v\|^2 + \|w\|^2 - 2 v^T w \Leftrightarrow$

$\Leftrightarrow 2 v^T w = \|v\|^2 + \|w\|^2 - \|v - w\|^2$

Set $x = x^{(k)}$, $y = x^*$

$f(x^{(k)}) - f(x^*) \leq \nabla f(x^{(k)})^T (x^{(k)} - x^*) = \frac{1}{2\eta} 2 (x^{(k)} - x^{(k+1)})^T (x^{(k)} - x^*) = \frac{1}{2\eta} \left(\|x^{(k)} - x^{(k+1)}\|^2 + \|x^{(k)} - x^*\|^2 - \|x^{(k+1)} - x^*\|^2 \right) =$

(difference between the evaluation of f at iteration k and the minimum; $v^T = x^{(k)} - x^{(k+1)}$, $w = x^{(k)} - x^*$)

$g^{(k)} = \nabla f(x^{(k)})$ — using GD relationship

↳ $x^{(k+1)} - x^{(k)} = -\eta g^{(k)} \Rightarrow g^{(k)} = \frac{x^{(k)} - x^{(k+1)}}{\eta} \Leftrightarrow \eta g^{(k)} = x^{(k)} - x^{(k+1)}$

$= \frac{1}{2\eta} \left(\eta^2 \|g^{(k)}\|^2 + \|x^{(k)} - x^*\|^2 - \|x^{(k+1)} - x^*\|^2 \right) =$

$= \frac{\eta}{2} \|g^{(k)}\|^2 + \frac{1}{2\eta} \left(\|x^{(k)} - x^*\|^2 - \|x^{(k+1)} - x^*\|^2 \right)$ — with a summation, all intermediate values cancel out (telescopic sum)

$\sum_{k=0}^{N-1} (f(x^{(k)}) - f(x^*)) \leq \frac{\eta}{2} \sum_{k=0}^{N-1} \|g^{(k)}\|^2 + \frac{1}{2\eta} \|x^{(0)} - x^*\|^2$ → (neglect $-\|x^{(N)} - x^*\|^2$ since comparison is $\leq$)

$x^{(0)}$ is an initial guess (initialization of the weights in a NN)

Lipschitz convex functions

Assumption: all the gradients are bounded ⇒ function is Lipschitz continuous

$\|g^{(k)}\| \leq L$, $\forall k \in \mathbb{R}$, $L \in \mathbb{R}$

Theorem: $f: \mathbb{R}^d \to \mathbb{R}$, convex, differentiable, minimum is $x^* \in \mathbb{R}^d$

$\|x^{(0)} - x^*\| \leq R$ and $\|\nabla f(x)\| \leq L$

Then choosing $\eta = \frac{R}{L\sqrt{N}}$

$\frac{1}{N} \sum_{k=0}^{N-1} (f(x^{(k)}) - f(x^*)) \leq \frac{RL}{\sqrt{N}}$

$\leq \frac{\eta}{2} N L^2 + \frac{1}{2\eta} R^2$, with $\ell(\eta)$ the right-hand side

$\ell'(\eta) = \frac{NL^2}{2} - \frac{R^2}{2\eta^2} = 0$

$NL^2 = \frac{R^2}{\eta^2} \Leftrightarrow \eta^2 = \frac{R^2}{NL^2} \Leftrightarrow \eta = \frac{R}{L\sqrt{N}}$

$\frac{\eta}{2} NL^2 + \frac{1}{2\eta} R^2 =$

$= \frac{RNL^2}{2L\sqrt{N}} + \frac{R^2 L\sqrt{N}}{2R} = \frac{RNL}{2\sqrt{N}} + \frac{RL\sqrt{N}}{2} = \frac{RNL\sqrt{N}}{2N} + \frac{RLN\sqrt{N}}{2N} = \frac{2RL\sqrt{N}}{2} = RL\sqrt{N}$

$\frac{RL}{\sqrt{N}} \leq \varepsilon \Leftrightarrow \frac{R^2 L^2}{N} \leq \varepsilon^2 \Leftrightarrow \frac{R^2 L^2}{\varepsilon^2} \leq N$ → $N = O\left(\frac{1}{\varepsilon^2}\right)$

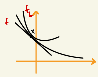
↓ does not depend directly on the dimensionality (L might depend very mildly with d)

Smooth Convex functions

Def: f: dom$(f) \to \mathbb{R}$, differentiable, $X \subseteq$ dom(f) is convex, $L \in \mathbb{R}^+$

function f is smooth with parameter L over X iff

$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L}{2} \|y - x\|^2$, $\forall x, y \in X$ — below tangent hyperplane at x



Lemma: dom(f) is open and convex and f: dom$(f) \to \mathbb{R}$ differentiable. Let $L \in \mathbb{R}^+$ the following statements are equivalent

i) f is smooth with parameter L

ii) g defined by $g(x) = \frac{L}{2} x^T x - f(x)$ is convex over dom$(g) :=$ dom(f)

1) If $L = 0$

$f(y) = f(x) + \nabla f(x)^T (y - x)$

2) $f(x) = x^2$

$f(y) = y^2 = x^2 + 2x(y - x) + (y - x)^2$

↳ $\frac{L}{2} = 1 \Leftrightarrow L = 2$

3) $f(x) = x^T Q x + b^T x + c$ | Q is a $d \times d$ matrix, sym.; $b \in \mathbb{R}^d$, $c \in \mathbb{R}$

is smooth with $L = 2\|Q\|$ ↳ spectral norm

$\|Q\| = \max_{x \in \mathbb{R}^d, x \neq 0} \frac{\|Ax\|}{\|x\|}$

4) $f(x) = x^4$, convex

Not smooth

↳ generally true if function has higher asymptotic behaviour than quadratic

Lemma: $f: \mathbb{R}^d \to \mathbb{R}$ convex, differentiable

The following statements are equivalent

i) f is smooth with parameter L

ii) $\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$, $\forall x, y \in \mathbb{R}^d$ — gradients are Lipschitz continuous

Let $f: \mathbb{R}^d \to \mathbb{R}$ differentiable, smooth L. Picking

$\eta = \frac{1}{L}$ GD satisfies $f(x^{(k+1)}) \leq f(x^{(k)}) - \frac{1}{2L} \|\nabla f(x^{(k)})\|^2$, $k \geq 0$

$x^{(k+1)} = x^{(k)} - \frac{1}{L} \nabla f(x^{(k)})$

$f(x^{(k+1)}) \leq f(x^{(k)}) + \nabla f(x^{(k)})^T (x^{(k+1)} - x^{(k)}) + \frac{L}{2} \|x^{(k+1)} - x^{(k)}\|^2$

$= f(x^{(k)}) + \nabla f(x^{(k)})^T \left(-\frac{1}{L} \nabla f(x^{(k)})\right) + \dots =$

$= f(x^{(k)}) - \frac{1}{L} \|\nabla f(x^{(k)})\|^2 + \frac{1}{2L} \|\nabla f(x^{(k)})\|^2 =$

$= f(x^{(k)}) - \frac{1}{2L} \|\nabla f(x^{(k)})\|^2$

• Smoothness L
• $\eta = \frac{1}{L}$

$\Rightarrow f(x^{(N)}) - f(x^*) \leq \frac{L}{2N} \|x^{(0)} - x^*\|^2$, $N > 0$

$\|g^{(k)}\|^2 = \|\nabla f(x^{(k)})\|^2$

$\frac{1}{2L} \sum_{k=0}^{N-1} \|\nabla f(x^{(k)})\|^2 \leq \sum_{k=0}^{N-1} (f(x^{(k)}) - f(x^{(k+1)})) = f(x^{(0)}) - f(x^{(N)})$

$\|x^{(0)} - x^*\| \leq R$

$\frac{LR^2}{2N} < \varepsilon \Rightarrow \frac{LR^2}{2\varepsilon} < N$ → $N = O\left(\frac{1}{\varepsilon}\right)$

Convergence Results for GD

- Lipschitz-convex functions $O\left(\frac{1}{\varepsilon^2}\right)$
- Smooth-functions $O\left(\frac{1}{\varepsilon}\right)$
- Smooth & strongly convex function $O\left(\log\left(\frac{1}{\varepsilon}\right)\right)$
- Accelerated GD $O\left(\frac{1}{\sqrt{\varepsilon}}\right)$

Accelerated GD

Aim: minimizing convex function $f: \mathbb{R}^d \to \mathbb{R}$ (with gradient ∇f)

"first order method" → uses only f and ∇f

What is the best first order method?

Nemirovski & Yudin (1979): every 1st order method needs in the worst-case $O\left(\frac{1}{\sqrt{\varepsilon}}\right)$ steps.

Nesterov (1983): Accelerated GD (AGD)

AGD

Let $f: \mathbb{R}^d \to \mathbb{R}$ convex, differentiable and smooth with parameter L

AGD reads:

- choose $z^{(0)} = y^{(0)} = x^{(0)}$ — initial guess
- for $k \geq 0$ set: (iterations)

$y^{(k+1)} = x^{(k)} - \frac{1}{L} \nabla f(x^{(k)})$ — "normal step"

$z^{(k+1)} = z^{(k)} - \frac{k+1}{2L} \nabla f(x^{(k)})$ — "aggressive step"

$x^{(k+1)} = \frac{k+1}{k+3} y^{(k+1)} + \frac{2}{k+3} z^{(k+1)}$ — "average"

with this, it is possible to achieve $O\left(\frac{1}{\sqrt{\varepsilon}}\right)$

Convergence of AGD

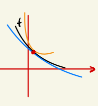
Theorem: Let $f: \mathbb{R}^d \to \mathbb{R}$ convex, differentiable with a global minimum x^*. Suppose f is smooth with parameter L. AGD yields

$$f(y^{(N)}) - f(x^*) \leq \frac{2L\|z^{(0)} - x^*\|^2}{N(N+1)} \quad N > 0$$

Smooth and Strongly Convex Functions

Def: Let f: dom$(f) \to \mathbb{R}$ be a convex and differentiable function, $X \subseteq$ dom(f) convex and $\mu \in \mathbb{R}^+$. Function f is called strongly convex with parameter μ over X if:

$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{\mu}{2} \|y - x\|^2$, $\forall x, y \in X$



Convergence

Theorem: Let $f: \mathbb{R}^d \to \mathbb{R}$ convex and differentiable. Suppose that f is smooth with parameter L and strongly convex with parameter $\mu > 0$. Choose $\eta = \frac{1}{L}$

Then:

i) $\|x^{(k+1)} - x^*\|^2 \leq \left(1 - \frac{\mu}{L}\right) \|x^{(k)} - x^*\|^2 \quad k \geq 0$ — squared distances to x^* are geometrically decreasing

ii) $f(y^{(N)}) - f(x^*) \leq \frac{L}{2} \left(1 - \frac{\mu}{L}\right)^N \|x^{(0)} - x^*\|^2 \quad N > 0$ — Absolute error after N it. is exponentially small in N

Recalling that $\ln(1 + x) \leq x$, we have:

$$N \geq \frac{L}{\mu} \ln\left(\frac{R^2 L}{2\varepsilon}\right) \quad \text{(Exercise)}$$

SGD

- Cost function : $J(\underline{w}) = \frac{1}{N}\sum_{i=1}^{N} J_i(\underline{w})$
- Algorithm : sample i_K randomly

$$\underline{w}^{(K+1)} = \underline{w}^{(K)} - \eta \nabla J_{i_K}(\underline{w}^{(K)})$$

Theorem: The following statements are equivalent to the condition that a diff function is μ-strongly convex

i) $f(\underline{y}) \geq f(\underline{x}) + \nabla f(\underline{x})^T(\underline{y}-\underline{x}) + \frac{\mu}{2}\|\underline{y}-\underline{x}\|^2$

ii) $g(\underline{x}) = f(\underline{x}) - \frac{\mu}{2}\|\underline{x}\|^2$ is convex

iii) $(\nabla f(\underline{x}) - \nabla f(\underline{y}))^T(\underline{y}-\underline{x}) \geq \mu\|\underline{y}-\underline{x}\|^2$

Simple convergence results for SGD

- $\|\underline{w}^{K+1} - \underline{w}^*\|$
- $J(\underline{w}^{(K+1)}) - J(\underline{w}^*)$

} measurements to prove convergence | convergence results in expectation

Thm: Assume $\underline{w}^*$ is the minimizer of J and

i) ∇J is L-Lipschitz $\rightarrow J(\underline{y}) \leq J(\underline{x}) + \nabla J(\underline{x})^T(\underline{y}-\underline{x}) + \frac{L}{2}\|\underline{y}-\underline{x}\|^2$

ii) J is μ-strongly convex $\rightarrow J(\underline{y}) \geq J(\underline{x}) + \nabla J(\underline{x})^T(\underline{y}-\underline{x}) + \frac{\mu}{2}\|\underline{y}-\underline{x}\|^2$

iii) $\|\nabla J_i(\underline{x})\| \leq c \quad , \forall \underline{w}, \forall i$

iv) $0 < \eta < \frac{1}{2\mu}$

v) $\mathbb{E}[g_K] = \nabla J(\underline{w}^{(K)})$

A) $\mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2\right] \leq (1-2\eta\mu)^K \|\underline{w}^{(0)} - \underline{w}^*\|^2 + \boxed{\frac{\eta c^2}{2\mu}}$ — noise component of the method

B) $\mathbb{E}\left[J(\underline{w}^{(K)})\right] - J(\underline{w}^*) \leq (1-2\mu\eta)^K\left[J(\underline{w}^{(0)}) - J(\underline{w}^*)\right] + \boxed{\frac{L\eta c^2}{4\mu}}$

A)

$$\|\underline{w}^{(K+1)} - \underline{w}^*\| = \|(\underline{w}^{(K)} - \eta\nabla J_{i_K}(\underline{w}^{(K)})) - \underline{w}^*\|^2 =$$

$$= \|\underline{w}^{(K)} - \underline{w}^*\|^2 - 2\eta\nabla J_{i_K}(\underline{w}^{(K)})^T(\underline{w}^{(K)} - \underline{w}^*) + \eta^2\|\nabla J_{i_K}(\underline{w}^{(K)})\|^2$$

$$\leq \|\underline{w}^{(K)} - \underline{w}^*\|^2 - 2\eta\nabla J_{i_K}(\underline{w}^{(K)})^T(\underline{w}^{(K)} - \underline{w}^*) + \eta^2 c^2$$

$$\mathbb{E}\left[\|\underline{w}^{(K+1)} - \underline{w}^*\|^2\right] \leq \mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2 - 2\eta\nabla J(\underline{w}^{(K)})^T(\underline{w}^{(K)} - \underline{w}^*) + \eta^2 c^2\right] =$$

$$\leq \mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2\right] - 2\eta\mu\,\mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2\right] + \eta^2 c^2 =$$

$$= (1-2\eta\mu)\,\mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2\right] + \eta^2 c^2$$

(iterate K times)

$$\mathbb{E}\left[\|\underline{w}^{(K)} - \underline{w}^*\|^2\right] \leq (1-\eta\mu)^K\|\underline{w}^{(0)} - \underline{w}^*\|^2 + \underbrace{\sum_{i=0}^{K}(1-2\eta\mu)^i\eta^2 c^2}_{\text{given that } \eta \text{ is } >0 \text{ and that } \forall i,\ c<\infty}$$

$$\sum_{i=0}^{\infty} r^i = \frac{1}{1-r}, \ |r|<1 \qquad r = 1-2\eta\mu \Rightarrow \frac{1}{1-r} = \frac{1}{1-(1-2\eta\mu)} = \frac{1}{2\eta\mu}$$

$$= (1-\eta\mu)^K\|\underline{w}^{(0)} - \underline{w}^*\|^2 + \frac{\eta^2 c^2}{2\eta\mu} =$$

$$= (1-\eta\mu)^K\|\underline{w}^{(0)} - \underline{w}^*\|^2 + \frac{\eta c^2}{2\mu}$$

B)

$$J(\underline{w}^{(K+1)}) \leq J(\underline{w}^{(K)}) + \nabla J(\underline{w}^{(K)})^T\left[\underline{w}^{(K+1)} - \underline{w}^{(K)}\right] + \frac{L}{2}\|\underline{w}^{(K+1)} - \underline{w}^{(K)}\|^2$$

$$= J(\underline{w}^{(K)}) + \nabla J(\underline{w}^{(K)})^T\left(\underline{w}^{(K)} - \eta\nabla J_{i_K}(\underline{w}^{(K)}) - \underline{w}^{(K)}\right) + \frac{L}{2}\|\eta\nabla J_{i_K}(\underline{w}^{(K)})\|^2$$

$$\leq J(\underline{w}^{(K)}) - \eta\nabla J(\underline{w}^{(K)})^T\nabla J_{i_K}(\underline{w}^{(K)}) + \frac{Lc^2\eta^2}{2}$$

$$\mathbb{E}\left[J(\underline{w}^{(K+1)}) - J(\underline{w}^*)\right] \leq J(\underline{w}^{(K)}) - J(\underline{w}^*) - \eta\|\nabla J(\underline{w}^{(K)})\|^2 + \frac{Lc^2\eta^2}{2}$$

$$\leq J(\underline{w}^{(K)}) - J(\underline{w}^*) - 2\eta\mu\left(J(\underline{w}^{(K)}) - J(\underline{w}^*)\right) + \frac{Lc^2\eta^2}{2}$$

$$= (1-2\eta\mu)\left(J(\underline{w}^{(K)}) - J(\underline{w}^*)\right) + \frac{Lc^2\eta^2}{2}$$

(iterate K times)

$$\mathbb{E}\left[J(\underline{w}^{(K)}) - J(\underline{w}^*)\right] \leq (1-2\eta\mu)^K\left(J(\underline{w}^{(0)}) - J(\underline{w}^*)\right) + \underbrace{\sum_{i}^{K\to\infty}(1-2\eta\mu)^i\frac{Lc^2\eta^2}{2}}_{\frac{1}{2\eta\mu}\cdot\frac{Lc^2\eta^2}{2} = \frac{Lc^2\eta}{4\mu}}$$

$$f(\underline{y}) \geq f(\underline{x}) + \nabla f(\underline{x})^T(\underline{y}-\underline{x}) + \frac{\mu}{2}\|\underline{y}-\underline{x}\|^2$$

LHS is minimized when $\underline{y} = \underline{w}^*$

RHS is minimized when $\underline{y} = \underline{x} - \frac{1}{\mu}\nabla J(\underline{x})$

$$J(\underline{w}^*) \geq J(\underline{w}) - \frac{1}{2\mu}\|\nabla J(\underline{w})\|^2$$

Line search Procedure

- $\underline{w}^{(0)}$
- $\underline{p}_K$, s.t. $\underline{p}_K^T\nabla J(\underline{w}^{(K)}) < 0$

$$\underline{p}_K = -\frac{\nabla J(\underline{w}^{(K)})}{\|\nabla J(\underline{w}^{(K)})\|}$$ ↓ best choice

- $\underline{w}^{(K+1)} = \underline{w}^{(K)} - \eta_K\underline{p}_K$

$$J(\underline{x}) = x_1 - x_2 + 2x_1x_2 + 2x_1^2 + x_2^2$$

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \qquad \nabla J(\underline{x}) = \begin{bmatrix} 1+2x_2+4x_1 \\ -1+2x_1+2x_2 \end{bmatrix}$$

$$\eta_K = \text{argmin}\, J\left(\underline{w}^{(K)} - \eta\nabla J(\underline{w}^{(K)})\right)$$

$$\underline{x}^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \nabla J(\underline{x}^{(0)}) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$J\left(\underline{x}^{(0)} - \eta\nabla J(\underline{x}^{(0)})\right) =$$

$$= J(-\eta, \eta) = \eta^2 - 2\eta$$

$$0 = \frac{dJ}{d\eta} = 2\eta - 2 = 2(\eta - 1) \Rightarrow \eta = 1 //$$

ADAGRAD

η { for rare features, η is big ; for features appearing often, η is small

$$\eta_{K,i} \propto \frac{1}{\sum_{i}^{K}\nabla J_i}$$ ← feature i

problem as the summation grows, η get small and the optimization gets stuck

ADADELTA / RMSProp } Averaging the gradients

ADAM

Universal Approx. Theorem

add that theory on error bounds is not available yet

Comparison PINN vs FEM on background

Low data ⟺ much physics
Some data ⟺ some physics
Lots data ⟺ large physics

} comparison: plain vs PINN: can solve any of them (maybe PINN more)