Norms

→ $|x^T y| \le \|x\|_2 \|y\|_2$ (Cauchy-Schwarz inequality)

→ $\|A\|_1 = \max_{1\le i\le m} (\|a_j\|_1)$

→ $\|A\|_\infty = $ max. row absolute sum

→ If $A = uv^T$, $\|A\|_2 = \|u\|_2 \|v\|_2$ (when $x = \frac{v}{\|v\|}$)

→ $\|AB\|_p \le \|A\|_p \|B\|_p$

→ $\|A\|_F = \sqrt{\sum_{i=0}^{m}\sum_{j=0}^{n} |a_{ij}|^2} = \sqrt{tr(AA^T)} = \sqrt{tr(A^TA)} = \sqrt{\sum_{i=0}^{n} \|a_i\|_2^2}$

Conditioning & Stability

→ $\hat{\kappa}(A) = \frac{\sigma_1}{\sigma_m} = \sqrt{\frac{\lambda_{max}(A^TA)}{\lambda_{min}(A^TA)}}$

→ $\hat{\kappa}^R(A) = \|A\| \|A^+\|$, where $\|A^+\| = (A^TA)^{-1}A^T$

Backward stability: $\tilde{f}(x) = f(x+\delta x)$

Forward stability: $\|\tilde{f}(x) - f(x)\| = O(\epsilon_m)$

SVD

$A = U\Sigma V^T$, $U \in \mathbb{R}^{m\times m}$, orthogonal; $\Sigma \in \mathbb{R}^{m\times n}$, diagonal; $V^T \in \mathbb{R}^{n\times n}$, orthogonal

→ $\|A\|_2 = \sigma_1$, $\|A\|_F = \sqrt{\sigma_1^2 + \cdots + \sigma_n^2}$

Projection

→ Projector matrix P, must be:
- → $P^2 = P$ (idempotent) → $Pv - v \in Null(P)$
- → Square matrix → Rank-deficient
- → $Px = x \;\forall x \in Range(P)$

→ Orthogonal projector also fulfils $P = P^T$ (symmetric)

→ Let us have $Ax = b$ has no soln., we can solve $A\hat{x} = Pb$ instead. P projects b onto the coln. space of A, which makes this eqn. solvable. Hence, $P = A(A^TA)^{-1}A^T$

→ If P is an orthogonal projector, so is $(I-P)$

→ Projection & SVD: If $A = U\Sigma V^T$, U forms a orthonormal basis of A. Hence, the projection matrix corresponding to A, $P = Q(\cancel{Q^TQ})Q^T = QQ$

Symmetric +ve Definite Matrix (SPD)

→ Properties
- → $A = A^T$
- → $(x, Ay) = (y, Ax)\;\; \forall x, y \in \mathbb{R}^m$
- → $x^TAx > 0, \;\forall x \in \mathbb{R}^m$

→ If A is S.P.D. & $X \in \mathbb{R}^{m\times n}$ is full rank, then X^TAX is also S.P.D.

→ All eigenvalues are +ve for any S.P.D.

→ Cholesky Decomposition: If A is S.P.D, A can be decomposed s.t. $A = R^TR$ (UTM)
→ $n^3/3$ FLOPs

Principal Component Analysis

1) Make the matrix 0-centered by subtracting the mean of the coln. from each element in the coln.

2) Variance of the coln.: $\frac{E(X^2) - \cancel{(E(X))^2}}{m} = \frac{\|a_j\|_2^2}{m}$

3) Hence, total variance, $T = \|a_1\|_2^2 + \cdots + \|a_n\|^2$
$= \|A\|_F^2$
$= \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_n^2$

Biggest contributor to the variance

→ To get the direction with highest variance, we need to find a vector $\hat{w}$ s.t.: $t_1 = A\hat{w}$, $\|t_1\|_2$ is maximised. The solution is $\hat{w} = v_1$.

$t_1 = Av_1 = \sum_{i=1}^{n} u_i\sigma_i v_i^T v_1 = \sigma_1 u_1$ $\left[v_i^Tv_1 = \begin{cases} 0, & i \ne 1 \\ 1, & i = 1\end{cases}\right]$

→ Similarly, $t_2 = \sigma_2 u_2$, where t_2 is the direction along second highest variance.

Eigen Decomposition

→ $A = X\Lambda X^{-1}$, Λ is a diagonal matrix containing eigenvalues and X is a matrix comprising of the respective eigenvectors

→ Eigenvectors are essentially Nullspace of $(A - \lambda I)$. Hence, $(A-\lambda I)$ must be a rank-deficient (singular) matrix, with $\det(A-\lambda I) = 0$

Linear Least Squares

→ $A\hat{x} = Pb$ // get b to coln. space of A

$A\hat{x} = A(A^TA)^{-1}A^Tb \Rightarrow \hat{x} = (A^TA)^{-1}A^Tb$

→ Soln. by Cholesky decomposition:

$A^TA\hat{x} = A^Tb$

$R^TR\hat{x} = A^Tb$

$R^Tw = A^Tb$ // Let $w = R\hat{x}$

↓ Solve for w

$R\hat{x} = w$

↓ Solve for x

$\hat{x}$

→ Soln. by QR factorisation

$Ax = Pb$

$QRx = QQ^Tb$

$Rx = Q^Tb$

↓ Solve for x

x

→ Soln with SVD

$Ax = Pb$

$U\Sigma V^Tx = UU^Tb$

$\Sigma V^Tx = U^Tb$

$\Sigma y = U^Tb$ // Let $V^Tx = y$

↓ Solve for y

$V^Tx = y$

↓ Solve for x

x

Algo	work
Cholesky	$mn^2 + n^3/3$
QR (Householder)	$2mn^2 - \frac{2n^3}{3}$
SVD	$2mn^2 + 11n^3$

Defective Eigenvalues & Matrices

→ An eigenvalue, for which algebraic multiplicity > geometric multiplicity is a defective eigenvalue.

→ Any matrix that has a defective eigenvalue is a defective matrix
→ It does not posses a full set of L.I. eigenvectors.

→ Diagonalisability: If $A \in \mathbb{R}^{m\times n}$ is not defective iff it has eigenvalue decomposition.

Orthogonal diagonalisability

not all non-defective matrices are unitary diagonalisable

→ Unitary diagonalisability: If a non-defective matrix A has eigenvalue decomposition $A = Q\Lambda Q^{-1} = Q\Lambda Q^*$, where Q is a unitary matrix.

Transpose conjugate

$Q \in \mathbb{C}^{m\times m}$ $Q^*Q = QQ^* = I$

→ Symmetric matrices have all real eigenvalues & eigenvectors.

→ Skew symmetric matrices have all imaginary eigenvalues. Skew symmetric matrices are also unitary diagonalisable.

Schur Factorisation

→ $A = QTQ^*$, where Q is unitary, and T is U.T.M.

→ SVD of T = SVD of A.

→ Every sq. matrix has a Schur factorisation.

* If A is real, A can be decomposed to UTU^T, where U & T are real, and T is quasi U.T.M.

→ Schur factorisation need not be unique.

QR-Factorisation

→ Gram-Schmidt orthogonalisation:

$v_j = a_j - (q_1^Ta_j)q_1 - \cdots - (q_{j-1}^Ta_j)q_{j-1}$

$q_j = \frac{v_j}{\|v_j\|_2}$

Hence $q_n = \frac{a_n - \sum_{i=0}^{n-1} r_{in}q_i}{r_{nn}}$, where $r_{ij} = \begin{cases} q_i^Ta_j, & i \ne j \\ \|a_j - \sum_{i=1}^{j-1} r_{ij}q_i\|_2, & i = j\end{cases}$

→ Gram Schmidt with projector: $q_n = \frac{P_n a_n}{\|P_n a_n\|}$, where

$P_n = I - \sum_{i=0}^{n-1} q_iq_i^T = I - \hat{Q}_{j-1}\hat{Q}_{j-1}^T$

→ Modified Gram-Schmidt:

Algo: for $j = 1 \to n$:

$v_j^{(1)} = a_j$

$v_j^{(2)} = P_{\perp q_1}a_j = (I - q_1q_1^T)a_j = v_j^{(1)} - q_1q_1^Tv_j^{(1)}$

$v_j^{(3)} = P_{\perp q_2}a_j = v_j^{(2)} - q_2q_2^Tv_j^{(2)}$

⋮

$v_j^{(j)} = v_j^{(j-1)} - q_{(j-1)}q_{(j-1)}^Tv_j^{(j-1)}$

$q_j = \frac{v_j^{(j)}}{\|v_j^{(j)}\|_2}$

Algo	FLOPs	Stability	Error
GS	$2mn^2$	Unstable	$O(\hat{\kappa}(A)^2\epsilon_m)$
MGS	"	Backward stable	$O(\hat{\kappa}(A)\cdot\epsilon_m)$
Householder	$2mn^2 - 2/3n^3$	"	$O(\epsilon_m)$

Householder Triangularisation

→ Perform transformations to convert A to R:

→ Let $x = \begin{bmatrix} - \\ - \\ \vdots \\ - \end{bmatrix}$, F should be s.t. $Fx = \begin{bmatrix} \circ \\ 0 \\ \vdots \\ 0 \end{bmatrix}$, $\circ = \|x\|\cdot e_1$

Hence, $y = Fx = (I - 2uu^T)x$, where $u = \frac{v}{\|v\|}$, where $v = \|x\|_2e_1 - x$

Algo: for $k = 1 \to n$:

$x = A(k:m, k)$ // row k to row m in the k^{th} coln.

$v_k = sgn(x_1)\cdot\|x\|_2e_1 + x$

$v_k = \frac{v_k}{\|v_k\|}$

$A(k:m, k:n) -= 2v_kv_k^TA(k:m, k:n)$

Analysis of Algos. (per iteration)

→ Power iteration: $O(m^2)$ due to matrix-vector multiplication

→ Inverse power iteration: $O(m^3)$ due to soln. of linear system of eqns.
→ Can be reduced to $O(m^2)$ by solving $(A-\mu I)^{-1}$ once

→ Rayleigh quotient iteration: $O(m^3)$, but lesser iterations are reqd.
→ Can be reduced to $O(m^2)$ by reducing A to tridiagonal / upper Hessenberg.

Eigen solvers

→ Rayleigh quotient: $\alpha = \dfrac{x^T A x}{x^T x}$

Power Iterations

→ Finds the eigenvector corresponding to largest eigenvector (by magnitude).

Algo: Initialise $v^{(0)}$ to a random unit vector

for $k = 1 \to \infty$,
$w = A v^{(k-1)}$
$v^{(k)} = \dfrac{w}{\|w\|}$
$\lambda^{(k)} = (v^{(k)})^T A (v^{(k)})$

Results:

$$\| v^{(k)} - (\pm q_1) \|_2 = O\left(\left| \frac{\lambda_2}{\lambda_1} \right| \right)^k$$

$$| \lambda^{(k)} - \lambda_1 | = O\left(\left| \frac{\lambda_2}{\lambda_1} \right|^{2k} \right)$$

If k is even, $v^{(k)} \to q_1$, otherwise, $v^{(k)} \to -q_1$

* Convergence is slow if $\lambda_2 \approx \lambda_1$

Inverse Power Iterations

Algo: Initialise μ = some value near λ_J,
$v^{(0)}$ = " random unit vector

Converges to closest eigenvalue of μ. If λ_J is the closest to μ and λ_K is second-most closest to μ,

for $k = 1 \to \infty$:
$w = (A - \mu I)^{-1} v^{(k-1)}$ // Solve by finding soln to system of lin eqns.
$v^{(k)} = \dfrac{w}{\|w\|}$
$\lambda^{(k)} = (v^{(k)})^T A (v^{(k)})$

$$\| v^{(k)} - (\pm q_J) \| = O\left(\left| \frac{\mu - \lambda_J}{\mu - \lambda_K} \right|^k \right)$$

$$| \lambda^{(k)} - \lambda_J | = O\left(\left| \frac{\mu - \lambda_J}{\mu - \lambda_K} \right|^{2k} \right)$$

Multiple Eigen values

→ Subspace / simultaneous iterations
→ Tone multiple vectors which are L.I.. Provided we have an ∞ precision computer, these will converge to different eigen vectors.
→ Assumption #1: The first n eigenvalues are distinct & well-separated
→ " #2: If $Q_1 = [q_1, \dots q_n]$, where $\{q_1, \dots q_n\}$ are eigen vectors of A, $Q_1^T v^{(0)}$ is non-singular, and all principal submatrices of $Q_1^T v^{(0)}$ are also singular.
→ Orthogonalise at each step to prevent loss of orthogonality, and hence make the algo. stable.

→ Works for large, sparse matrices

Algo: Initialise $\hat{Q}^{(0)} \in R^{m \times n}$
for $k = 1 \to \infty$:
$Z^{(k)} = A \hat{Q}^{(k-1)}$
$\hat{Q}^{(k)}, \hat{R}^{(k)} = Z$ // QR factorisation

Rayleigh Quotient Iteration

Algo: Initialise $v^{(0)}$ to some random unit vector
$\lambda^{(0)} = (v^{(0)})^T A (v^{(0)})$
for $k = 1 \to \infty$:
$w = (A - \lambda^{(k-1)} I)^{-1} v^{(k-1)}$ // again linear system of eqns
$q^{(k)} = \dfrac{w}{\|w\|}$
$\lambda^{(k)} = (q^{(k)})^T A (q^{(k)})$

→ Very fast convergence:

$$\| v^{(k+1)} - (\pm q_J) \| = O\left(\| v^{(k)} - (\pm q_J) \|^3 \right)$$

$$| \lambda^{(k+1)} - \lambda_J | = O\left(| \lambda^{(k)} - \lambda_J |^3 \right)$$

→ Pure QR algorithm (dense matrices)

Algo: $A^{(0)} = A$
for $k = 1 \to \infty$
$Q^{(k)}, R^{(k)} = A^{(k-1)}$ // orthogonalise
$A^{(k)} = R^{(k)} Q^{(k)}$

→ As $k \to \infty$, $A^{(k)}$ approaches Schur form.
→ Can also be considered as simultaneous inverse iteration applied to a flipped identity matrix.
→ Modified QR (most used by eigensolvers)

Full Algo: Define $A^{(0)}$ s.t. $(Q^{(0)})^T A^{(0)} (Q^{(0)}) = A$ // tridiagonalisation of A
for $k = 1 \to \infty$:
Pick a shift $\mu^{(k)}$ // many methods for picking, e.g. $\mu^{(k)} = A^{(k-1)}_{mm}$
$Q^{(k)} R^{(k)} = A^{(k-1)} - \mu^{(k)} I$ // shifted QR factorisation
$A^{(k)} = R^{(k)} Q^{(k)} + \mu^{(k)} I$
If any off-diagonal entries are close to 0, set $A_{j,j+1} = A_{j+1,j} = 0$
Split $A^{(k)}$ into A_1 & A_2 s.t. $A^{(k)} = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}$
Apply QR algo (from tridiagonalisation) on A_1 & A_2.

→ Krylov subspace method (fully iterative):
→ Krylov subspace is a subspace rich in eigenvectors. This is the set of vectors $\{b, Ab, A^2 b, \dots\}$.
→ This seq. is similar to power iterations
→ For this to be an actual subspace, $b, Ab, A^2 b$ etc. must be L.I. They are confirmed to be L.I. if A is full-rank.
→ This method is computationally unstable

→ Arnoldi Iteration (To construct Krylov subspace)

Algo: b = arbitrary vector
$q_1 = \dfrac{b}{\|b\|}$
for $n = 1 \to \infty$
$v = A q_n$
for $j = 1 \to n$
$h_{jn} = q_j^T v$
$v = v - h_{jn} q_j$
$h_{(n+1)n} = \|v\|$
$q_{n+1} = \dfrac{v}{h_{(n+1)n}}$

→ At the end of the iterations, we have:
→ constructed a subspace rich in eigenvalues of A
→ Projected A onto the subspace, to obtain H_n
→ Hence, H_n is a projection of A onto K_n
→ Eigenvalues of H_n are Arnoldi eigenvalue estimates, a.k.a. Ritz values

→ Arnoldi iterations can be viewed as polynomial approximation
→ Arnoldi approximation problem: Find $p^n \in P^n$ s.t. $\| p^n(A) b \|_2$ is minimum. P^n is the set of monic polynomials of deg. n.
→ Soln. to this problem is actually $P_{H_n}(z) = \det(zI - H_n)$
→ As $n \to \infty$, the soln. approaches eigen values of A.