Linear Algebra

Analytic Geometry

Norms

- A norm on a vector space V is a function which assigns each vector x its length such that for all $\lambda \in \mathbb{R}$ and x, y in V the following hold:
  - Absolutely homogeneous: $\|\lambda \boldsymbol{x}\| = |\lambda| \|\boldsymbol{x}\|$
  - Triangle Inequality (the sum of the lengths of any two sides must be greater than or equal to the length of the remaining side): $\|\boldsymbol{x} + \boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\|$
  - Positive Definite: $\|x\| \geqslant 0 \text{ and } \|x\| = 0 \iff x = 0$
- ℓ_1 norm (Manhattan norm): $\|x\|_1 := \sum_{i=1}^{n} |x_i|$

- ℓ_2 norm (euclidean norm): $\|x\|_2 := \sqrt{\sum_{i=1}^n x_i^2} = \sqrt{x^\top x}$
  - euclidean distance of the vector
- Frobenius norm of a matrix:
  $\|X\|_F = \sqrt{\sum_{i=1}^M \sum_{j=1}^N X_{ij}^2} = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} = \sqrt{\operatorname{trace}(AA^H)} = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2(A)}$ for sigma, the singular values of A

Inner Products

- Dot product: $x \cdot y = \boldsymbol{x}^\top \boldsymbol{y} = \sum_{i=1}^n x_i y_i$
- General Inner products
  - bilinear mapping Ω is a mapping with 2 arguments and linear in each argument:
    $\Omega(\lambda \boldsymbol{x} + \psi \boldsymbol{y}, \boldsymbol{z}) = \lambda \Omega(\boldsymbol{x}, \boldsymbol{z}) + \psi \Omega(\boldsymbol{y}, \boldsymbol{z})$
  - For $\Omega : V \times V \to \mathbf{R}$, mapping is symmetric if $\Omega(x, y) = \Omega(y, x)$
  - Mapping is positive definite if $\forall x \in V \backslash \{0\} : \Omega(x, x) > 0, \quad \Omega(0, 0) = 0$
  - A positive definite, symmetric bilinear mapping is called an inner product on V, denoted $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$
- Given a basis B, can write x, y in terms of that basis. Then
  $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \left\langle \sum_{i=1}^n \psi_i \boldsymbol{b}_i, \sum_{j=1}^n \lambda_j \boldsymbol{b}_j \right\rangle = \sum_{i=1}^n \sum_{j=1}^n \psi_i \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle \lambda_j = \hat{\boldsymbol{x}}^\top \boldsymbol{A} \hat{\boldsymbol{y}}$ where $A_{ij} := \langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle$ and $\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}$ are the coordinates wrt the basis. The inner product is uniquely determine by A.

Outer Products

- In matrix multiplication, can be done by taking the columns of A times rows of B to get AB
- one column u times one row v^T produces a matrix. While inner product $v^T u$ produces a scalar, outer product produces $uv^T = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} 3 & 4 & 6 \end{bmatrix} = \begin{bmatrix} 6 & 8 & 12 \\ 6 & 8 & 12 \\ 3 & 4 & 6 \end{bmatrix}$ a rank 1 matrix.
- The column space of the outer product is one-dimensional - the line in the direction of u. The row space is the line through v
- $(uv^T)^T = vu^T$

Lengths and Distances

- Inner products and norms are closely related in the sense that any inner product induces a norm in a natural way, such that we can compute lengths of vectors using the inner product. However, not every norm is induced by an inner product. The Manhattan norm (3.3) is an example of a norm without a corresponding inner product.
- Length $\|x\| := \sqrt{\langle x, x \rangle} = \sqrt{x \cdot x}$
- Schwarz Inequality: $|\boldsymbol{v} \cdot \boldsymbol{w}| \leq \|\boldsymbol{v}\| \|\boldsymbol{w}\|$
- Triangle Inequality: $\|\boldsymbol{v} + \boldsymbol{w}\| \leq \|\boldsymbol{v}\| + \|\boldsymbol{w}\|$
- Distance between x and y: $d(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\| = \sqrt{\langle \boldsymbol{x} - \boldsymbol{y}, \boldsymbol{x} - \boldsymbol{y} \rangle}$. If we use the dot product as the inner product, then we get the euclidean distance.
- A metric d satisfies: symmetric, positive definite, triangle inequality ($d(\boldsymbol{x}, \boldsymbol{z}) \leqslant d(\boldsymbol{x}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{z})$).

Angles

- Inner products capture the geometry of a vector space by defining the angle ω between two vectors.
- From Cauchy-Schwarz Inequality, $-1 \leqslant \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\| \|\boldsymbol{y}\|} \leqslant 1$ for $x, y \neq 0$. Then there exists a unique

$\omega \in [0, \pi],\ \cos\omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|}$. Using the dot product as the inner product, this translates to $\cos\omega = \frac{\langle \boldsymbol{x}, \boldsymbol{y} \rangle}{\sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle \langle \boldsymbol{y}, \boldsymbol{y} \rangle}} = \frac{\boldsymbol{x}^\top \boldsymbol{y}}{\sqrt{\boldsymbol{x}^\top \boldsymbol{x} \boldsymbol{y}^\top \boldsymbol{y}}}$

- Orthogonality: $\langle x, y \rangle = 0 \implies x \perp y$. Orthonormal when $||x|| = 1 = ||y||$. Can be orthogonal wrt one inner product but not another

Inner Products of Functions

- $\langle u, v \rangle := \int_a^b u(x)v(x)dx$ for limits $a, b < \infty$. If this evaluates to 0, functions u and v are orthogonal.
- Unlike inner products on finite-dimensional vectors, inner products on functions may diverge

Vectors and Matrices

- Linear combination: $cv + dw = c\begin{bmatrix}1\\1\end{bmatrix} + d\begin{bmatrix}2\\3\end{bmatrix} = \begin{bmatrix}c+2d\\c+3d\end{bmatrix}$
- Dot product: $\boldsymbol{v} \cdot \boldsymbol{w} = v_1 w_1 + v_2 w_2$.
- Dot product = 0 indicates orthogonality: $v \cdot w = 0 \iff v \perp w$

Matrix x Vector Multiplication

- Row multiplication: $Ax = \begin{bmatrix}(\text{row } I) & \cdot x\\(\text{row } 2) & \cdot x\\(\text{row } 3) & x\end{bmatrix}$
  - Example: $A\boldsymbol{x} = \begin{bmatrix}1 & 0 & 0\\-1 & 1 & 0\\0 & -1 & 1\end{bmatrix}\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix} = \begin{bmatrix}(1,0,0)\cdot(x_1,x_2,x_3)\\(-1,1,0)\cdot(x_1,x_2,x_3)\\(0,-1,1)\cdot(x_1,x_2,x_3)\end{bmatrix}$
- Column multiplication: $Ax = x(\text{column } 1) + y(\text{column } 2) + z(\text{column } 3)$
  - Example: $Ax = \begin{bmatrix}2 & 5\\1 & 3\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} = x_1\begin{bmatrix}2\\1\end{bmatrix} + x_2\begin{bmatrix}5\\3\end{bmatrix}$

Matrix Multiplication

- Matrix multiplication is associative and distributive but not commutative: (AB)C = A(BC)
- To multiply AB, if A has n columns, B must have n rows. Left columns = Right rows

1. Dot product each row in A with column in B
   - The entry in row i and column j of AB is (row i of A) · (column j of B)
2. Matrix A times every column of B
   - Each column of AB is a combination of the columns of A.
   - $A\begin{bmatrix}b_1 \cdots b_p\end{bmatrix} = \begin{bmatrix}Ab_1 \cdots Ab_p\end{bmatrix}$
3. Every row of A times matrix B
   - Each row of AB is a combination of the rows of B.
   - $\begin{bmatrix}\text{row } i \quad \text{of } A\end{bmatrix}\begin{bmatrix}1 & 2 & 3\\4 & 5 & 6\\7 & 8 & 9\end{bmatrix} = \begin{bmatrix}\text{row } i \quad \text{of } AB\end{bmatrix}$
4. Multiply columns 1 ton of A times rows 1 ton of B. Add those matrices.
   - Column 1 of A multiplies row 1 of B. Columns 2 and 3 multiply rows 2 and 3.

    - $\begin{bmatrix} \text{col } 1 & \text{col } 2 & \text{col } 3 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} \text{row } 1 & \cdots \\ \text{row } 2 & \cdots \\ \text{row } 3 & \cdots \end{bmatrix} = (\text{col } 1)(\text{row } 1) + (\text{col } 2)(\text{row } 2) + (\text{col } 3)(\text{row } 3)$
    - $AB = \begin{bmatrix} a \\ c \end{bmatrix} [E \quad F] + \begin{bmatrix} b \\ d \end{bmatrix} [G \quad H] = \begin{bmatrix} \boldsymbol{aE} + bG & \boldsymbol{aF} + bH \\ \boldsymbol{cE} + dG & \boldsymbol{cF} + dH \end{bmatrix}$
- Block multiplication - if blocks of A are the right size to multiply blocks of B, can divide into smaller matrices and multiply
    - $\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} = \begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} \\ A_{21}B_{11} + A_{22}B_{21} \end{bmatrix}$

Inverses

- $A^{-1}A = I = AA^{-1}$ if the inverse exists r = m = n
- For square matrices, a left inverse = right inverse
- Easiest test for invertibility: If singular, det = 0, inverse does not exist
- $(\boldsymbol{AB})^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$

Left Inverse

- Matrix with full column rank r = n, N(A) = {0}, independent columns, 0 or 1 solutions to Ax = b
- $(A^TA)^{-1}A^TA = I$ so the left inverse is $(A^TA)^{-1}A^T$ (Moore-Penrose Pseudoinverse)
- AA_{left}^{-1} is then a projection onto the column space
- Used in least squares since we sub in $\boldsymbol{x} = \left(\boldsymbol{A}^\top \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{b}$ for $\boldsymbol{x} = A^{-1}\boldsymbol{b}$

Right Inverse

- Matrix with full row rank r = m, $N(A^T) = \{0\}$ independent rows, infinite solutions to Ax = b, n - m free variables
- $AA^T(AA^T)^{-1} = I$ so the right inverse is $A^T(AA^T)^{-1}$
- $A_{right}^{-1}A$ is then a projection onto the row space

Pseudoinverse

- $y = A^+(Ay)$
- If x, y both in the row space, then $Ax \neq Ay$ - there is a one to one mapping from x to Ax.
- Pseudoinverse often used in stats since least squares matrices often are not of full rank
- Finding pseudoinverse
    - SVD: $A = U\Sigma V^T$
    - Here $\Sigma = \begin{bmatrix} \sigma_1 & 0 & \dots \\ \dots & \sigma_r & \dots \\ 0 & 0 & 0 \end{bmatrix}$ has rank r, n x m
    - $\Sigma^+ = \begin{bmatrix} 1/\sigma_1 & 0 & \dots \\ \dots & 1/\sigma_r & \dots \\ 0 & 0 & 0 \end{bmatrix}$ n x m, rank r
    - Then $A^+ = V\Sigma^+U^T$
- Note that $\Sigma\Sigma^+$ is m x m with diagonal ones and is a projection onto row space, $\Sigma^+\Sigma$ an n x n matrix projecting onto column space.

Permutations

- A matrix that executes row exchanges - this may be needed to make a matrix invertible. PA = LU
- P = identity matrix with reordered rows. n! possible matrices / reorderings
- For invertible P, $P^{-1} = P^T$

Transposes

- Switch columns and rows - $(A^T)_{ij} = A_{ji}$
- For symmetric matrices, $A^T = A$.
- Note $A^T A$ is always symmetric, square.
- $(\boldsymbol{A} + \boldsymbol{B})^\top = \boldsymbol{A}^\top + \boldsymbol{B}^\top$
  $(\boldsymbol{A}\boldsymbol{B})^\top = \boldsymbol{B}^\top \boldsymbol{A}^\top$

Echelon Form

- Staircase from top corner separating values from 0's, ie. U or L
- R = reduced row echelon form. $R = \begin{bmatrix} I & F \\ 0 & 0 \end{bmatrix}$
  - Take matrix to echelon form, then make 0's above and below all pivots
  - Normalize each row to make the pivots equal to 1.
  - Matrix forms I in the pivot rows and columns. The R form above has r pivot columns that make up I and n-r free columns that make up F

Elimination and Factorization

Gaussian Elimination

- Pick pivot in the first row - pivot must not be zero. If we have a zero pivot, can try to exchange rows to produce a non-zero pivot.
- Reduce the numbers below pivot to zero using linear combinations of rows
- For well-behaved matrix, will reduce to U, an upper triangular matrix.
- Back Substitution
  - Perform elimination on augmented matrix, eg. $\left[\begin{array}{cc|c} 2 & 5 & 1 \\ 1 & 3 & 2 \end{array}\right]$
  - Use upper triangular equations to solve for variables in reverse order $x_1, \ldots, x_n$. At each step have one additional unknown in each equation
- Row Echelon Form: Any equation system in row-echelon form always has a "staircase" structure.
  - All rows that contain only zeros are at the bottom of the matrix
  - Looking at nonzero rows only, the first nonzero number from the left (also called the pivot or the leading coefficient) is always strictly to the right of the pivot of the row above it.
- Reduced Row Echelon Form
  - In row exchlon form
  - Every pivot is 1
  - The pivot is the only nonzero entry in its column

- The key idea for finding the solutions of Ax = 0 is to look at the nonpivot columns, which we will need to express as a (linear) combination of the pivot columns. The reduced row echelon form makes this relatively straightforward, and we express the non-pivot columns in terms of sums and multiples of the pivot columns that are on their left
- If we bring the augmented equation system into reduced row-echelon form, we can read out the inverse on the right-hand side of the equation system.
- Elimination in Matrix Form - all elimination steps could be combined into a single matrix E, that transforms A into U, ie. EA = U
- The pivot columns indicate the linearly independent columns - the others are linearly dependent

Factorization A= LU

- From elimination we have EA = U for some unknown E. For $L = E^{-1}$, then get A = LU
- L adds back to U what E removed from A. If there are no row exchanges, L is just made of the column multipliers

Vector Spaces

- All vector spaces contain the origin.
- Vectors form a subspace if all linear combinations of those vectors are also in the subspace. Vector space must be closed under linear combinations.
- All subspaces of $\mathbb{R}^3$: $\mathbb{R}^3$, plane through the origin, line through the origin, zero vector only
- Rank of A = # of pivots from elimination

Groups

- Consider a set G and an operation ⊗ : G × G → G group defined on G.
- Then G := (G, ⊗) is called a group if the following hold:
  - Closure of G under ⊗: ∀x, y ∈ G : x ⊗ y ∈ G
  - Associativity: ∀x, y, z ∈ G : (x ⊗ y) ⊗ z = x ⊗ (y ⊗ z)
  - Neutral element: ∃e ∈ G ∀x ∈ G : x ⊗ e = x and e ⊗ x = x 4.
  - Inverse element: ∀x ∈ G ∃y ∈ G : x ⊗ y = e and y ⊗ x = e. We often write x −1 to denote the inverse element of x
- Vector spaces are groups

Solving Ax = 0

- Elimination does not change solutions, so null space is unchanged by elimination
- Can solve Ux = 0 instead then after elimination. Left with pivot columns and free columns, the columns without pivots in which any value can be assigned to the x's corresponding to that number column
- Pivot variables can be found through back substitution, free columns we choose values freely. Set values or free columns to 1 and 0 - this forms our special solution
- The null space contains all the combinations of the special solutions. There is one special solution per free variable, and the number of free variables is n - r

Solving Ax = b

- Typical approach - augmented matrix -> elimination

- Solvability condition - Ax = b is solvable only when b is in C(A)
- Finding complete solution
    1. Set all free variables to 0, solve Ax= b for the pivot variables. This gives us a particular solution
    2. Find solutions in the null space
    3. Take linear combination of particular solution and null space solutions. $X_{total} = X_p + X_n$. $Ax_p = b$ + $Ax_n = 0$ = $A(x_n + x_p) = b$. With a particular solution, can add anything in the null space and still get b
    4. X_n is combination of the special solutions - $X_{total} = X_p + c_1 X_{SS1} + c_2 X_{SS2} \dots$
- In short: Find a particular solution to Ax = b, find all solutions to Ax = 0, combine these two results to obtain a general solution

Rank

- # of pivot columns
- Dimension of C(A) column space
- The column rank equals the row rank
- Only full column rank matrices are invertible

Full Column Rank

- r = n < m
- No free variables, N(A) = {0}
- Solution to Ax = b - unique if a solution exists -> 0 or 1 solutions
- $R = \begin{bmatrix} I \\ 0 \end{bmatrix}$

Full Row Rank

- r = m < n
- Solution exists for Ax = b for all b due to free variables
- Have n - r = n - m free variables
- $R = \begin{bmatrix} I & F \end{bmatrix}$

Full Row + Column Rank

- r = m = n -> square matrix, always invertible (defines invertibility)
- N(A) = {0}
- Ax = b has 1 solution - see this by combing rules for full row and full col rank
- $R = \begin{bmatrix} I \end{bmatrix}$

Singular Matrix r < m, n

- Ax = b has 0 or ∞ solutions
- $R = \begin{bmatrix} I & F \\ 0 & 0 \end{bmatrix}$

Independence, Span, Basis

- Vectors $x_1, \dots, x_n$ are independent if no combination gives the zero vector (except the zero combination with scalars = 0): $c_1 x_1 + \dots + c_n x_n \neq 0$
- Columns are independent if N(A) = {0} $\iff$ rank = n

- Columns are dependent if some vector is in the null space $\iff$ rank < n . Think of 3 vectors in a plane - these must be dependent
- Span - vectors $v_1, \ldots, v_n$ span a space means a space consists of all combinations of these vectors
- Generating sets are sets of vectors that span vector (sub)spaces, i.e., every vector can be represented as a linear combination of the vectors in the generating set.
- Every linearly independent generating set of V is minimal and is called a basis of V
- Basis - for a vector space, a basis is a sequence of vectors which
    1. are Independent
    2. span the space
    - A basis is not unique, but all bases for a space have the same number of vectors. In $\mathbb{R}^n$ need n vectors to form basis - this is the dimension D of the space
    - Finding a basis: write spanning vectors as columns in matrix, reduce to row-echelon form, the spanning vectors associated with the pivot columns form a basis
- Dimension D = number of vectors needed to form a basis

The Four Subspaces

Column Space

- $C(A) \in \mathbb{R}^m$
- C(A) = all linear combinations of the columns
- We can solve Ax = b when b is in the column space
- dim(C(A)) = # of pivot columns = rank(A) = r
- Basis = pivot columns

Null Space

- $N(A) \in \mathbb{R}^n$
- All solutions x to equation Ax = 0
- Zero vector is always in the null space
- The null space contains all the combinations of the special solutions. There is one special solution per free variable, and the number of free variables is n - r
- Null space matrix N, RN = 0. $N = \begin{bmatrix} -F \\ I \end{bmatrix}$
- dim(N(A)) = # of free variables = n - r
- Basis = special solutions

Row Space

- All linear combination of the rows or the column space of A^T, $C(A^T)$
- $C(A^T) \in \mathbb{R}^n$
- $dim(C(A^T)) = r$

Null Transpose Space

- Nullspace of A^T, $N(A^T)$, the left nullspace of A
    - For $A^T y = 0$, $y \in N(A^T) \implies y^T A = 0$
- $N(A^T) \in \mathbb{R}^m$

- $dim(N(A^T)) = m - r$, the number of free columns in A^T

Matrix Spaces

- S = symmetric, U = upper triangular
- $S \cap U =$ symmetric + upper triangular = diagonal. S+U is the linear combinations of the matrices in the two spaces
- $dim(S) + dim(U) = dim(S \cap U) + dim(S + U)$

Affine Subspace

- Let V be a vector space, $x_0 \in V,\ U \subset V$ a subspace. Then the subset $L = \boldsymbol{x}_0 + U := \{\boldsymbol{x}_0 + \boldsymbol{u} : \boldsymbol{u} \in U\} = \{\boldsymbol{v} \in V | \exists \boldsymbol{u} \in U : \boldsymbol{v} = \boldsymbol{x}_0 + \boldsymbol{u}\} \subseteq V$ is called affine subspace or linear manifold of V . U is called direction or direction space, and x0 is called support point.
- Examples of affine subspaces are points, lines, and planes in R3 , which do not (necessarily) go through the origin.

Orthogonality

Vector Orthogonality

- For two vectors $x \cdot y = x^T y = 0 \iff x \perp y$
- $x \perp y \implies ||x||^2 + ||y||^2 = ||x + y||^2$
- Zero vector orthogonal to all other vectors

Subspace Orthogonality

- Subspaces $S \perp T \implies$ all vectors in S perp to all vectors in T
- $C(A^T) \perp N(A)$: Row Space orthogonal to Null Space
  - Ax = 0 for x in N(A)
  - Then by defn x is perpendicular to each row in A using matrix multiplication
  - $(c_1 row_1 + c_2 row_2 \dots)^T x = 0$
- $C(A) \perp N(A^T)$: Column space orthogonal to left nullspace
- Orthogonal complements: a complement contains all vectors perp to the other space
  - Null space and row space are orthogonal complements in $\mathbb{R}^n$
  - For complements U, U^T, we have $U \cap U^\perp = \{\mathbf{0}\}$. Can decompose a vector in the larger space V as a combination of vectors in the complements: $\boldsymbol{x} = \sum_{m=1}^{M} \lambda_m \boldsymbol{b}_m + \sum_{j=1}^{D-M} \psi_j \boldsymbol{b}_j^\perp, \quad \lambda_m, \psi_j \in \mathbb{R}.$
  - The orthogonal complement can also be used to describe a plane U (two-dimensional subspace) in a three-dimensional vector space. More specifically, the vector w with $||w|| = 1$, which is orthogonal to the plane U, is the basis vector of U^T

Projections

- $A^T A$ invertible $\iff$ A has independent columns. Share a rank and null space
- p = projection of b onto a. Since p lies on a, it is a scalar multiple of a: $p = xa$. Left to find x
- $x = \frac{a^T b}{a^T a}$
  - Define e = b - p = orthogonal vector from b to a vector. Therefore $a \perp b - p$

    - Then $a^T(b-xa) = 0 \implies xa^Ta = a^Tb \implies x = \frac{a^Tb}{a^Ta}$
- $p = xa = a\frac{a^Tb}{a^Ta} = \frac{aa^T}{a^Ta}b = Pb$ for
- **Projection Matrix** $P = \frac{aa^T}{a^Ta}$
    - $P^T = P$ - symmetric
    - $P^2 = P$ - projection of projection same as single projection
- Point of projection - Ax = b may have no solutions, ie. b not in C(A). Can instead solve $A\hat{x} = p$, where p is the projection of b onto the column space and $\hat{x}$ is the solution to this altered problem.
- Higher dimensions - $p = A\hat{x}$
    - Key is $e = b - A\hat{x} \perp plane$
    - $A^TA\hat{x} = A^Tb$
    - $\hat{x} = (A^TA)^{-1}A^Tb$
    - projection matrix $P = A(A^TA)^{-1}A^T$. The inverse cannot be distributed bc A not necessarily square - A^TA is square however
- Some implications:
    - If b in C(A), $Pb = b$. Derived from $b = Ax \implies A(A^TA)^{-1}A^TAx = Ax = b$
    - If $b \perp C(A)$, $Pb = 0$. Derived from $A(A^TA)^{-1}A^Tb = A(A^TA)^{-1}(0) = 0$

Least Squares

- Given number of non-collinear points, have some line $Ax = b$ with errors $||e||^2 = e_1^2 + e_2^2 + \dots$
- We take points on the line $p_1, p_2, p_3, \dots$ instead of original points $b_1, b_2, b_3, \dots$
- Use $A^TA\hat{x} = A^Tb$ to derive normal equations. The partial derivatives of $||Ax - b||^2$ are zero when $A^TA\hat{x} = A^Tb$
- If A has independent columns, then A^TA is invertible is crucial to making this work.
    - If $A^TAx = 0$, then x can only be the zero vector

Orthogonal Matrices

- Orthonormal vectors: $q_i^Tq_j = \begin{cases} 0 & if\ i = j \\ 1 & if\ i \neq j \end{cases}$
- $Q = [\, q_1 q_2 \dots q_n \,]$
- $Q^TQ = I$. For square Q, this implies $Q^T = Q^{-1}$
- Q has orthonormal columns, to project onto its column space $P = Q(Q^TQ)^{-1}Q^T = QQ^T$
    - If P is square, then projection is onto whole space $QQ^T = I$
- Transformations by orthogonal matrices are special because the length of a vector x is not changed when transforming it using an orthogonal matrix A.

Orthonormal Basis

- The basis vectors are orthogonal to each other and where the length of each basis vector is 1.
- $\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle = 0 \quad \text{for } i \neq j$ and $\langle \boldsymbol{b}_i, \boldsymbol{b}_i \rangle = 1$
- Gram-Schmidt is the process for forming an orthonormal basis.

Gram-Schmidt

- Process to find orthonormal projection - method iteratively constructs and orthogonal basis from any basis of V
    - High-dimensional data quite often possesses the property that only a few dimensions contain most information, and most other dimensions are not essential to describe key properties of the data. Compression causes loss of information, so we need to find the most informative dimensions in the data
    - The idea is to find the vector in the subspace spanned by the columns of A that is closest to b, i.e., we compute the orthogonal projection of b onto the subspace spanned by the columns of A -> the least-squares solution
- For independent vectors a, b, c, let A, B, C be orthogonal, then $q_1 = \frac{A}{||A||}$, $q_2 = \frac{B}{||B||}, q_3 = \frac{C}{||C||}$
- Let a = A, then need to change b to be orthogonal to a. Requires B = e, the error vector
- $B = b - \frac{A^T b}{A^T A} A$, then $A^T B = 0 \implies A \perp B$
- To make C, need a third vector orthogonal to both A, B by subtracting off the components in the a, b directions
- $C = c - \frac{A^T c}{A^T A} A - \frac{B^T c}{B^T B} B$
- A = QR
    - Basic expression of G-S
    - $A = [\, a_1 a_2 \,] = [\, q_1 q_2 \,] R$ for R upper triangular

Projection onto Affine Subspaces

- Given affine space $L = x_0 + U$ with b_1, b_2 basis vectors for U
- Transform $L - x_0 = U$, now can use projection onto a vector subspace. Projection equals $\pi_L(\boldsymbol{x}) = \boldsymbol{x}_0 + \pi_U\left(\boldsymbol{x} - \boldsymbol{x}_0\right)$

Determinants

- A number associated with every **square** matrix. Test for invertibility when $det(A) \neq 0$
- The determinant det(A) is the signed volume of an n-dimensional parallelepiped formed by columns of the matrix A.
- If A invertible, $\det\left(\boldsymbol{A}^{-1}\right) = \frac{1}{\det(\boldsymbol{A})}$
- Similar matrices possess the same determinant - for a linear mapping all transformation matrices have the same determinant. Therefore the determinant is invariant to the choice of basis of a linear mapping.
- Characteristic polynomial for square matrix A: $p_A(\lambda) := \det(A - \lambda I) == c_0 + c_1\lambda + c_2\lambda^2 + \cdots + c_{n-1}\lambda^{n-1} + (-1)^n\lambda^n$. Note $c_0 = \det(\boldsymbol{A}),\ c_{n-1} = (-1)^{n-1}\operatorname{tr}(\boldsymbol{A})$

Properties

1. $det\ I = 1$
2. Row exchanges: for each row exchange, reverse the sign of the determinant
    - Det of permutations either +- 1 depending on if we do even or odd number of exchanges
3. Scalar factoring and linear function

a. Scalar factor can be pulled out of a row: $\begin{bmatrix} ta & tb \\ c & d \end{bmatrix} = t \begin{bmatrix} a & b \\ c & d \end{bmatrix}$

b. Determinant is a linear function of rows (within row, not globally $det(A+B) \neq det(A) + det(B)$): $\begin{bmatrix} a+a' & b+b' \\ c & d \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} a' & b' \\ c & d \end{bmatrix}$

4. 2 equal rows $\implies det = 0$
    - Proof: exchange the equal rows, the determinant should be the same since matrix is unchanged but violates property 2. Therefore must be 0
5. Subtraction scaled row l from row k $\implies$ determinant does not change, ie. elimination does not change determinant
    - $\begin{bmatrix} a & b \\ c-la & d-lb \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} a & b \\ -la & -lb \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} + -l \begin{bmatrix} a & b \\ a & b \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$
    - Proved using properties 3b, 3a, and 4
6. Row of 0's $\implies det = 0$
    - ie. elimination gives zero row - singular and non-invertible
    - Say t = 0, then det = 0 by 3b
7. Product of diagonals for triagular matrix: $det\ U = d_1 \times d_2 \times \ldots \times d_n$
    - Product of pivots after elimination (with sign determined by row exchanges too)
    - Using properties 5, 3a, 1 could make diagonal and factor out d's: $d_1 \ldots d_n(I)$
8. $det\ A = 0 \iff$ A is singular
    - Another way of seeing A is invertible only when have pivots of full rank
9. $det\ AB = (det\ A)(det\ B)$
10. $det\ A^T = det\ A$

Cofactors

- $det\ A = \sum_{n!\ terms} \pm a_{1\alpha} a_{2\beta} a_{3\gamma} \ldots a_{n\omega}$
- For 3x3: $\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} + (-1)a_{11}a_{23}a_{32} + a_{12}a_{21}a_{33} + (-1)a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} + (-1)a_{13}a_{22}a_{31}$
- Basic approach: determinant of smaller sub matrix: $\begin{vmatrix} a_{11} & & \\ & a_{22} & a_{23} \\ & a_{32} & a_{33} \end{vmatrix}$
- Cofactor of $a_{ij} = C_{ij}$, given + if i + j is even, - if i+j odd: $\begin{vmatrix} + & - & + \\ - & + & - \\ + & - & + \end{vmatrix}$
- Cofactor formula: $det\ A = a_{11}C_{11} + \ldots + a_{1n}C_{1n}$ along a single row (here row 1)

Eigenvalues and Eigenvectors

- $Ax = \lambda x$ - for a square matrix A
    - eigenvector - a vector that fits the property Ax || x
    - eigenvalue - some scalar value that allows eigenvector property to hold
- All vectors that are collinear to x are also eigenvectors of A.

- Eigenspace - the set of all eigenvectors of A associated with an eigenvalue λ spans a subspace, the eigenspace of A wrt λ, denoted E_λ. The set of all eigenvalues of A is called the eigenspectrum (or spectrum) of A. The eigenspace for λ solves the system $(A - \lambda I)x = 0$, ie. its the null space of the $(A - \lambda I)$
- If A is singular, 0 will be an eigenvalue
- Projection matrix - Px = x with $\lambda = 1$ for vectors in the plane, $\lambda = 0$ for x fitting Px = 0
- Permutation matrix - $\lambda = 1, -1$
- Trace = sum down diagonal of A = $a_{11} + a_{22} + \ldots + a_{nn}$
- The identity matrix has a single eigenvalue 1 repeated n times and clearly the eigenspace spans the full n dimensions.
- Geometrically, the eigenvector corresponding to a nonzero eigenvalue points in a direction that is stretched by the linear mapping. The eigenvalue is the factor by which it is stretched. If the eigenvalue is negative, the direction of the stretching is flipped.
- It is often a convention that eigenvalues are sorted in descending order, so that the largest eigenvalue and associated eigenvector are called the first eigenvalue and its associated eigenvector, and the second largest called the second eigenvalue and its associated eigenvector, and so on.

Tricks

- With symmetric matrices, eigenvalues will always be real
- Eigenvalues are always complex conjugates
- Triangular matrix - eigenvalues are just the diagonal values
- A matrix and its transpose possess the same eigenvalues (not necessarily same eigenvectors)
- Similar matrices have the same eigenvalues - eigenvalues are independent of the basis for a transformation matrix.

General Procedure

- Write $Ax = \lambda x$
- Rewrite as $(A - \lambda I)x = 0$, giving us a singular matrix $(A - \lambda I)$ with det = 0
- Get characteristic equation (polynomial) from $det(A - \lambda I) = 0$
- Solve for λ as root to characteristic equation
- An n x n matrix will have n eigenvalues, though could be repeated
- Once lambdas found, find x with elimination, finding the null space of the singular matrix with lambda values plugged in
- If repeated eigenvalue, will have fewer eigenvectors. Otherwise the n eigenvectors for n eigenvalues are linearly independent.
- **Eigenvalues sum to trace, multiply to determinant**. Put another way the determinant is the product of eigenvalues $\det(\boldsymbol{A}) = \prod_{i=1}^{n} \lambda_i$ and the trace is the sum of the eigenvalues $\operatorname{tr}(\boldsymbol{A}) = \sum_{i=1}^{n} \lambda_i$

Diagonalization / Eigendecomposition

- Suppose n linearly independent eigenvectors of A - form columns of matrix S
- $AS = A\left[\, x_1 x_2 \ldots x_n \,\right] = \left[\, \lambda_1 x_1 \ldots \lambda_n x_n \,\right] = \left[\, x_1 x_2 \ldots x_n \,\right] \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ 0 & 0 & \ldots & \lambda_n \end{bmatrix} = S\Lambda$
- Key: $A = S\Lambda S^{-1}$ - A is similar to a diagonal matrix of eigenvalues

- Powers of A: $A^K = S\Lambda^K S^{-1}$
- Theorem: $A^K \to 0$ as $k \to \infty$ if all $|\lambda_i| < 1$
    - dependent on assumption of n independent eigenvectors, otherwise cannot diagonalize
- A is sure to be diagonalizable if all eigenvalues are different. If eigenvalues repeated, may or may not have n independent eigenvectors
- Can use to solve recursive formulae: $u_{k+1} = Au_k$ then can solve using $u_{k+1} = A^{k+1}u_0$
- Procedure
    - Compute eigenvalues and eigenvectors
    - Check that A can be diagonalized - do the eigenvectors form a basis in $\mathbb{R}^n$
    - Construct the transformation matrix P - collect eigenvectors of A in $S := [x_1\ x_2\ \dots]$

Symmetric Matrices

- Eigenvalues are always real and eigenvectors can be chosen to be orthogonal
- While usually have $A = S\Lambda S^{-1}$, now have $A = Q\Lambda Q^{-1} = Q\Lambda Q^T$ (can make the eigenvectors orthonormal to fit Q defn)
- Notation for complex conjugates: x is conjugate of $\bar{x}$
- Defn symmetric $A = A^T$ for real, otherwise $A = \bar{A}^T$.
  $A = Q\Lambda Q^T = [\,q_1 \dots q_n\,] \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & 0 & \dots & \lambda_n \end{bmatrix} \begin{bmatrix} q_1 \\ \dots \\ q_n \end{bmatrix} = \lambda_1 q_1 q_1^T + \dots \lambda_n q_n q_n^T$
- Each qq^T is a projection matrix - every symmetric matrix is a combination of perpendicular projection matrices
- For symmetric matrices, signs of the pivots are the same as the signs of the eigenvalues: # pos pivots = # pos eigenvalues

Positive Definite Matrices

- Symmetric matrices with only positive eigenvalues
- All pivots are positive, all subdeterminants are positive
- Tests
    1. Eigenvalues: $\lambda > 0\ \forall \lambda$
    2. Determinant: $a > 0,\ ac - b^2 > 0$
    3. Pivots: $a > 0,\ \frac{ac-b^2}{a} > 0$
    4. Key Test: $x^T Ax > 0$
- If only $x^T Ax \geq 0$ holds then the matrix is positive semi-definite
- $x^T Ax$ produces a quadratic form:
  $A = \begin{bmatrix} 2 & 6 \\ 6 & 20 \end{bmatrix} \implies x^T Ax = [\,x_1 \quad x_2\,] \begin{bmatrix} 2x_1 + 6x_2 \\ 6x_1 + 20x_2 \end{bmatrix} = 2x_1^2 + 12x_1x_2 + 20x_2^2$
- det = 4, trace = 22 - then both eigenvalues must be positive. $x^T Ax > 0$ except at x = 0
- Intuition is we need the squares to overwhelm the combined term. Notice $a_{11},\ a_{22}$ are the cofficients for the squares
- Essentially minimizing $f(x, y) = 2x^2 + 12xy + 20y^2 = 2(x + 3y)^2 + 2y^2$ - factoring using complete the square shows the sum of two squares -> always positive

- If A,B pos def, then A + B also pos def. Notice for least squares, A^TA is square and symmetric, so $x^TA^TAx = (Ax)^T(Ax) = ||Ax||^2 \geq 0$. Only 0 when the vector is 0, so for any matrix of rank n can say least squares will be positive definite
- For pos, def A $\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \hat{\boldsymbol{x}}^\top \boldsymbol{A}\hat{\boldsymbol{y}}$ defines an inner product with respect to basis B where $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}$ are the coordinate representations of x,y in the basis B
- The null space of A consists only of zero vector because $x^\top Ax > 0$ for all $x \neq 0$
- We can always produce a positive, semidefinite matrix by $\boldsymbol{S} := \boldsymbol{A}^\top \boldsymbol{A}$. If A has full column rank then positive definite.

Complex Matrices

- For complex $z = \begin{bmatrix} z_1 \\ \dots \\ z_n \end{bmatrix} \in \mathbb{C}^n$, cannot use z^Tz for length squared since it is negative!
- Instead use $\bar{z}^Tz = ||z||^2$, where z-bar is the complex conjugate (flipped sign on complex part). Eg. $z = \begin{bmatrix} 1 \\ i \end{bmatrix}$, $\bar{z} = \begin{bmatrix} 1 \\ -i \end{bmatrix}$
- Hermetian: $z^Hz = \bar{z}^Tz$ - use this for the inner product when dealing with complex vectors
- Hermetian matrices: $A^H = A$ - real eigenvalues, orthogonal eigenvectors. Just like real symmetric matrices

Similar Matrices

- For 2 square matrices, A and B are similar if they have the same eigenvalues (easily checked with trace / determinant)
- For some matrix, can factor B: $B = M^{-1}AM$
- Have already seen a similar matrix in $\Lambda = S^{-1}AS \implies$ A is similar to Λ
- All matrices with same eigenvalues of A are similar to A and can be transformed via some M - form families
- If $\lambda_1 = \lambda_2$, then might not be diagonalizable depending if there is one eigenvector or two - this will split into different families.

Jordan Form

- Take the most diagonalizable family of similar matrices. Not always easy to do in practice since we need exactly the same eigenvalues
- Create jordon blocks that contain a single eigvector: $J_i = \begin{bmatrix} \lambda_i & 0 & \dots & 0 \\ 0 & \lambda_i & \dots & 0 \\ 0 & 0 & \dots & \lambda_i \end{bmatrix}$
- Every square matrix A is similar to a jordan matrix made of these jordon blocks.
- # blocks = # of eigenvectors

Decompositions

Cholesky Decomposition

- A square root equivalence on symmetric, positive definite matrices
- $A = LL^T$ where L is a lower triangular matrix with positive diagonal elements:

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ l_{n1} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & \cdots & l_{n1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & l_{nn} \end{bmatrix}$$

- Eg $\boldsymbol{A} = \begin{bmatrix} l_{11}^2 & l_{21}l_{11} & l_{31}l_{11} \\ l_{21}l_{11} & l_{21}^2 + l_{22}^2 & l_{31}l_{21} + l_{32}l_{22} \\ l_{31}l_{11} & l_{31}l_{21} + l_{32}l_{22} & l_{31}^2 + l_{32}^2 + l_{33}^2 \end{bmatrix}$, L is the Cholesky factor of A and L is unique.
- We can backward calculate what the components l_{ij} should be given the values for A and previously computed elements of L.
- The covariance matrix of a multivariate Gaussian variable (see Section 6.5) is symmetric, positive definite. The Cholesky factorization of this covariance matrix allows us to generate samples from a Gaussian distribution. It also allows us to perform a linear transformation of random variables, which is heavily exploited when computing gradients in deep stochastic models.
- Given the Cholesky decomposition $A = LL^T$, we know that $det(A) = det(L)det(L^T) = det(L)^2$. Since L is a triangular matrix, the determinant is simply the product of its diagonal entries so that $det(A) = \prod_i l_{ii}^2$.

Singular Value Decomposition

- $A = U\Sigma V^T$ for Σ diagonal, U,V orthogonal. Exists for any matrix A.
- Special case of positive definite: $A = Q\Lambda Q^T$
- Basic Idea: for V a basis in the row space, U a basis in the col space, and for sigma scaling factors:
  - $AV = U\Sigma \implies A\,[\,v_1 \ldots v_r\,] = [\,u_1 \ldots u_r\,] \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \ldots \end{bmatrix}$
  - $A = U\Sigma V^{-1} = U\Sigma V^T$
- The matrix S consists of the singular values. Ordered largest in the top left to smallest in the bottom right. S is unique, m x n rectangular - it is the same size as A. If m > n, square matrix of sigmas on top of matrix of 0. For n > m, square matrix of sigmas to the left of a matrix of 0's.
- The SVD expresses a change of basis in both the domain and codomain. This is in contrast with the eigendecomposition that operates within the same vector space, where the same basis change is applied and then undone. What makes the SVD special is that these two different bases are simultaneously linked by the singular value matrix S.
- The left singular vectors of A are eigenvectors of AA^T. The right singular vectors of A are the eigenvectors of A^TA. The non zero singular values of A are the square roots of the nonzero eigenvalues of AA^T and are equal to the nonzero eigenvalues of A^TA.
- Substituting a matrix with its SVD has often the advantage of making calculation more robust to numerical rounding errors.

Interpretations

- Expresses every row of A as a linear combination of the rows of V^T, the right singular vectors. The rows of US are the coefficients to those combinations
- Expresses every column of A as linear combination of the columns of U, the left singular vectors, with coefficients given by SV^T. Therefore we interpret just the rows or columns of the decomposition, we can say something important about A.
- Say rows are customers, columns products, values ratings. The right singular values could be customer types, with each customer defined as a linear mixture of types. Left sigular values could be product types and SVD expresses A as a mixture of product types.

- When only a single direction is interesting, can use PCA instead of a full SVD.

Procedure

- Using $A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}$

1. $A^T A = \begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 25 & 7 \\ 7 & 25 \end{bmatrix}$
2. Find eigens of $A^T A$: $x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix},\ x_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ and $\lambda_1 = 32,\ \lambda_2 = 18$. Normalize eigenvectors: divide by length (here $\sqrt{2}$)
3. Set up: $A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = A = \begin{bmatrix} & \\ & \end{bmatrix} \begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} = U\Sigma V^T$
4. Find U's. AA^T is a positive definite symmetric matrix. $AA^T = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T$
   - Calc AA^T: $AA^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} \begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$
   - $\lambda's$ of AA^T are the same as for $A^T A$ - (32, 18). $x_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\ x_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
   - Then $U = x_1 = [\, x_1 \quad x_2 \,] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
5. $A = U\Sigma V^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$

Linear Transformations

- Follow two rules
  1. $T(v+w) = T(v) + T(w)$
  2. $T(cV) = cT(v)$
- If we want to use a matrix, we need to use coordinates that the transformation is relative to: $T(v) = Av$
- Coordinates come from a basis - $v = c_1 v_1 + \ldots + c_n v_n$. We typically assume the standard basis but not necessary
- Definitions for Transformations Φ of Two Vector Spaces V, W
  - Injective if $\Phi(x) = \Phi(y) \implies x = y$
  - Surjective if $\Phi(V) = W$
  - Bijective if injective and surjective. Every element in W can be "reached" via the mapping from V. A reversible mapping must also exist $\Psi = \Phi^{-1}$
- Special Transformations
  - Isomorphism: V to W linear and bijective
    - Finite-dimensional vector spaces V and W are isomorphic if and only if dim(V) = dim(W). Intuitively, this means that vector spaces of the same dimension are kind of the same thing, as they can be transformed into each other without incurring any loss.
  - Endomorphism: V to V linear
  - Automorphism: V to V linear and bijective

Matrix Form

- Consider a vector space V and an ordered basis B = (b1, . . . , bn) of V . For any x in V we obtain a unique representation (linear combination) $\boldsymbol{x} = \alpha_1 \boldsymbol{b}_1 + \ldots + \alpha_n \boldsymbol{b}_n$ of x with respect to B. Then a1, . . . , an are the coordinates of x with respect to B, and the vector $\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}$ is the coordinate vector/coordinate representation of x with respect to the ordered basis B.
- For vector spaces V, W with ordered bases B in Rn, C in Rm, take linear mapping $\Phi : V \to W$, then we can represent B uniquely in terms of C as $\Phi\left(\boldsymbol{b}_j\right) = \alpha_{1j}\boldsymbol{c}_1 + \cdots + \alpha_{mj}\boldsymbol{c}_m = \sum_{i=1}^{m} \alpha_{ij}\boldsymbol{c}_i$. The transformation matrix is then defined by m x n A: $A_\Phi(i, j) = \alpha_{ij}$.
- The transformation matrix can be used to map coordinates with respect to an ordered basis in V to coordinates with respect to an ordered basis in W.

Constructing a Transformation Matrix

- $T : \mathbb{R}^n \to \mathbb{R}^m$
- Choose basis $v_1, \ldots, v_n$ for inputs and $w_1, \ldots, w_m$ for outputs
- For projection onto a line, choose v1 = line itself, v2 = vector perpendicular to line. Then for $v = c_1 v_1 + c_2 v_2$, $T(v) = c_1 v_1$ taking $(c_1, c_2) \to (c_1, 0)$. Then $A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$
- Easiest choice for a transformation matrix is the eigenvector basis, since this leads to transformation Λ
- To find A given a basis, let the first column of A equal $T(v_1) = a_{11}w_1 + \ldots + a_{m1}w_m$, the second column equal $T(v_2) = a_{12}w_1 + \ldots + a_{m2}w_m$, etc...

Change of Basis

- W matrix of new basis vectors as columns
- To go to vector x in new basis from c in old basis, $x = Wc$
- Transforming between bases is equivalent to similar matrices. M is a change of basis matrix in $B = M^{-1}AM$
- Two matrices are equivalent if there exists matrices S and T st $\tilde{\boldsymbol{A}} = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{S}$. Similar matrices are always equivalent but not vice versa.

Matrix Calculus

Univariate Calc Key Results

- Difference quotient: $\frac{\delta y}{\delta x} := \frac{f(x+\delta x) - f(x)}{\delta x}$
- Derivative: $\frac{\mathrm{d}f}{\mathrm{d}x} := \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$
- Taylor Polynomial of degree n at x_0: $T_n(x) := \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$ where $f^{(k)}\left(x_0\right)$ is the kth derivative at x0
- Taylor Series at x_0: $T_\infty(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$
- Differentiation Rules:
$$\begin{aligned} &\text{Product rule:} && (f(x)g(x))' = f'(x)g(x) + f(x)g'(x) \\ &\text{Quotient rule:} && \left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2} \\ &\text{Sum rule:} && (f(x) + g(x))' = f'(x) + g'(x) \\ &\text{Chain rule:} && (g(f(x)))' = (g \circ f)'(x) = g'(f(x))f'(x) \end{aligned}$$
- You can think of $\frac{d}{dx}$ as an operator that maps a function of one parameter to another function - it's distributive and we can just pull out constants

Partial Differentiation and Gradients

- Partial Derivative: $\frac{\partial f}{\partial x_1} = \lim_{h \to 0} \frac{f(x_1+h, x_2, \ldots, x_n) - f(x)}{h}$ - collect them in a row vector
- Gradient is simply a vector of partials of f. Each entry is a partial derivative with respect to a different variable: $\nabla_{\boldsymbol{x}} f = \operatorname{grad} f = \frac{\mathrm{d}f}{\mathrm{d}\boldsymbol{x}} = \begin{bmatrix} \frac{\partial f(\boldsymbol{x})}{\partial x_1} & \frac{\partial f(\boldsymbol{x})}{\partial x_2} & \cdots & \frac{\partial f(\boldsymbol{x})}{\partial x_n} \end{bmatrix} \in \mathbb{R}^{1 \times n}$
- The reason why we define the gradient vector as a row vector is twofold: First, we can consistently generalize the gradient to vector-valued functions f : Rn → Rm (then the gradient becomes a matrix). Second, we can immediately apply the multi-variate chain rule without paying attention to the dimension of the gradient.
- Partial $\text{Chain rule:} \quad \frac{\partial}{\partial \boldsymbol{x}}(g \circ f)(\boldsymbol{x}) = \frac{\partial}{\partial \boldsymbol{x}}(g(f(\boldsymbol{x}))) = \frac{\partial g}{\partial f}\frac{\partial f}{\partial \boldsymbol{x}}$
  - $\frac{\mathrm{d}f}{\mathrm{d}t} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1(t)}{\partial t} \\ \frac{\partial x_2(t)}{\partial t} \end{bmatrix} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial t}$ for x a function of t
- For $f(x_1, x_2),\ x_1(s,t),\ x_2(s,t)$, the gradient given by $\frac{\mathrm{d}f}{\mathrm{d}(s,t)} = \frac{\partial f}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial (s,t)} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial s} & \frac{\partial x_2}{\partial t} \end{bmatrix}$

Gradients of Vector Valued Functions

- Function $\boldsymbol{f} : \mathbb{R}^n \to \mathbb{R}^m$ and vector X. Can take $\boldsymbol{f}(\boldsymbol{x}) = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_m(\boldsymbol{x}) \end{bmatrix} \in \mathbb{R}^m$
- Partial derivative of a vector valued function $\frac{\partial \boldsymbol{f}}{\partial x_i} = \begin{bmatrix} \frac{\partial f_1}{\partial x_i} \\ \vdots \\ \frac{\partial f_m}{\partial x_i} \end{bmatrix}$
- The Jacobian is the collection of all first order partial derivatives of vector valued function f:
  $\boldsymbol{J} = \nabla_x \boldsymbol{f} = \frac{\mathrm{d}\boldsymbol{f}(\boldsymbol{x})}{\mathrm{d}\boldsymbol{x}} = \begin{bmatrix} \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial x_n} \end{bmatrix} == \begin{bmatrix} \frac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_1(\boldsymbol{x})}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_m(\boldsymbol{x})}{\partial x_n} \end{bmatrix}$. J is an m x n matrix
- J can be seen as a basis change matrix, taking the determinant is the area of a parallelogram. The change in the absolute value of the determinant of J describes how the area changes under the basis change.
- The calculus approach gives us the familiar Jacobian $\boldsymbol{J} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} \end{bmatrix}$ for a mapping of y in terms of x. The absolute value of the Jacobian determinant |det(J)| is the factor by which areas or volumes are scaled when coordinates are transformed. These transformations are extremely relevant in ML learning in the context of training deep neural networks using the reparametrization trick, also called infinite perturbation analysis.
- Gradients in least squares: for $\boldsymbol{y} = \boldsymbol{\Phi\theta},\ L(e) := \|e\|^2,\ e(\boldsymbol{\theta}) := \boldsymbol{y} - \boldsymbol{\Phi\theta}$, we seek $\frac{\partial L}{\partial \boldsymbol{\theta}}$. We use the chain rule $\frac{\partial L}{\partial \boldsymbol{\theta}} = \frac{\partial L}{\partial e}\frac{\partial e}{\partial \theta}$. Then $\frac{\partial L}{\partial e} = 2e^\tau$ since $\|e\|^2 = e^T e$ and $\frac{\partial e}{\partial \theta} = -\Phi \in \mathbf{R}^{N \times D}$. In total:
  $\frac{\partial L}{\partial \theta} = -2e^\top \Phi \overset{(5,.77)}{=} \underbrace{-2\left(\boldsymbol{y}^\top - \boldsymbol{\theta}^\top \boldsymbol{\Phi}^\top\right)}_{1 \times N} \underbrace{\Phi}_{N \times D} \in \mathbf{R}^{1 \times D}$

Gradients of Matrices

- For details, see explained.ai
- When we move from derivatives of one function to derivatives of many functions, we move from the world of

vector calculus to matrix calculus.

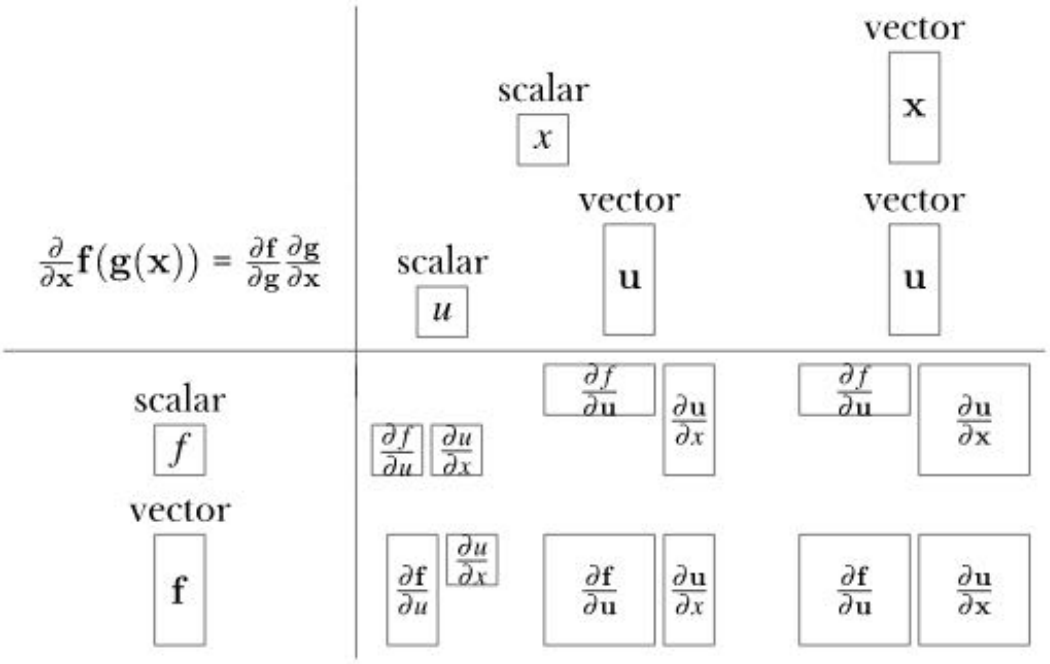
- Gradient vectors organize all of the partial derivatives for a specific scalar function. Say we have functions $f(x,y) = 3x^2y,\ g(x,y) = 2x + y^8$ If we have two functions, we can also organize their gradients into a matrix by stacking the gradients - this gives us a Jacobian matrix.
  $J = \begin{bmatrix} \nabla f(x,y) \\ \nabla g(x,y) \end{bmatrix} = \begin{bmatrix} \frac{\partial f(x,y)}{\partial x} & \frac{\partial f(x,y)}{\partial y} \\ \frac{\partial g(x,y)}{\partial x} & \frac{\partial g(x,y)}{\partial y} \end{bmatrix} = \begin{bmatrix} 6yx & 3x^2 \\ 2 & 8y^7 \end{bmatrix}$. Note this layout is the **numerator layout**. Some in ML use the denominator layout, which is the transpose: $\begin{bmatrix} 6yx & 2 \\ 3x^2 & 8y^7 \end{bmatrix}$
- With multiple scalar-valued functions, we can combine them all into a vector just like we did with the parameters. Let y = f(x) be a vector of m scalar-values functions that each take a vector x of length n. From our prior examples: $y_1 = f_1(x) = 3x_1^2x_2,\ y_2 = f_2(\mathbf{x}) = 2x_1 + x_2^8$
- Generally, the Jacobian is the collection of all m x n possible partial derivatives, ie. a stack of m gradients wrt x: $\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \begin{bmatrix} \nabla f_1(\mathbf{x}) \\ \nabla f_2(\mathbf{x}) \\ \cdots \\ \nabla f_m(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} f_1(\mathbf{x}) \\ \frac{\partial}{\partial \mathbf{x}} f_2(\mathbf{x}) \\ \cdots \\ \frac{\partial}{\partial \mathbf{x}} f_m(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x_1} f_1(\mathbf{x}) & \frac{\partial}{\partial x_2} f_1(\mathbf{x}) & \cdots & \frac{\partial}{\partial x_n} f_1(\mathbf{x}) \\ \frac{\partial}{\partial x_1} f_2(\mathbf{x}) & \frac{\partial}{\partial x_2} f_2(\mathbf{x}) & \cdots & \frac{\partial}{\partial x_n} f_2(\mathbf{x}) \\ \cdots & & \cdots & \\ \frac{\partial}{\partial x_1} f_m(\mathbf{x}) & \frac{\partial}{\partial x_2} f_m(\mathbf{x}) & \cdots & \frac{\partial}{\partial x_n} f_m(\mathbf{x}) \end{bmatrix}$
- We often have functions combined by element-wise binary operators (apply an operator the first item of ecch vector to get the first item of the output, then the second, etc). We are left with an ugly Jacobian applied to $\mathbf{y} = \mathbf{f}(\mathbf{w}) \bigcirc \mathbf{g}(\mathbf{x})$ wrt the x vector. However, we are left with a diagonal matrix, since when i not equal j, $\frac{\partial}{\partial w_j}(f_i(\mathbf{w}) \bigcirc g_i(\mathbf{x})) = 0$ since taking derivatives of constants.
- Scalars: changing vectors by a scalar is really an element-wise operation. For scalar, wrt to the variable z, we get a vector $\frac{\partial}{\partial z}(f_i\left(x_i\right) + g_i(z)) = \frac{\partial (x_i + z)}{\partial z} = \frac{\partial x_i}{\partial z} + \frac{\partial z}{\partial z} = 0 + 1 = 1$. Alternatively, wrt x, we get a diagonal Jacobian with elements $\frac{\partial}{\partial x_i}(f_i\left(x_i\right) \otimes g_i(z)) = x_i \frac{\partial z}{\partial x_i} + z\frac{\partial x_i}{\partial x_i} = 0 + z = z$
- Sums: we often need to sum over results, and we can move the sum outside of the derivative. For $y = \text{sum}(\mathbf{f}(\mathbf{x})) = \sum_{i=1}^{n} f_i(\mathbf{x})$, we can get gradient $\nabla y = \left[\sum_i \frac{\partial f(x)}{\partial x_1}, \sum_i \frac{\partial f(x)}{\partial x_2}, \ldots, \sum_i \frac{\partial f(x)}{\partial x_n}\right] = \left[\sum_i \frac{\partial x_i}{\partial x_i}, \sum_i \frac{\partial x_i}{\partial x_i}, \ldots, \Sigma_i \frac{\partial x}{\partial x_i}\right] = \left[\frac{\partial x_1}{\partial x_1}, \frac{\partial x_2}{\partial x_2}, \ldots, \frac{\partial x_n}{\partial x_n}\right] = [1, 1, \ldots, 1] = \vec{1}^T$ since $\frac{\partial}{\partial x_j} x_i = 0 \text{ for } j \neq i$.

Chain Rules

- Forward differentiation from x to y: $\frac{dy}{dx} = \frac{du}{dx}\frac{dy}{du}$. Backward differentiation from y to x: $\frac{dy}{dx} = \frac{dy}{du}\frac{du}{dx}$
- When x affects y through a single data flow path in nested functions, we simply introduce intermediate variables, compute derivatives wrt each, then combine using the chain rule, eg $\frac{dy}{dx} = \frac{du_4}{dx} = \frac{du_4}{du_3}\frac{du_3}{du_2}\frac{du_2}{du_1}\frac{du_1}{dx}$. With an expression like $f(x) = x + x^2$, we need a different technique since x affects y through 2 different pathways.
- We use the law of total derivatives - to compute derivative we need to sum up all possible contributions from changes in x to the change in y. **Single variable total derivative chain rule** that assumes all variables may be codependent: $\frac{\partial f(x, u_1, \ldots, u_n)}{\partial x} = \frac{\partial f}{\partial x} + \frac{\partial f}{\partial u_1}\frac{\partial u_1}{\partial x} + \frac{\partial f}{\partial u_2}\frac{\partial u_2}{\partial x} + \ldots + \frac{\partial f}{\partial u_n}\frac{\partial u_n}{\partial x} = \frac{\partial f}{\partial x} + \sum_{i=1}^{n}\frac{\partial f}{\partial u_i}\frac{\partial u_i}{\partial x}$
- The total derivative is adding terms because it represents a weighted sum of all x contributions to the change in y.
- Vector chain rule: We can take the single variable chain rule $\frac{d}{dx}f(g(x)) = \frac{df}{dg}\frac{dg}{dx}$ and convert to a vector rule $\frac{\partial}{\partial x}\mathbf{f}(\mathbf{g}(x)) = \frac{\partial \mathbf{f}}{\partial \mathbf{g}}\frac{\partial \mathbf{g}}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial g_1} & \frac{\partial f_1}{\partial g_2} \\ \frac{\partial f_2}{\partial g_1} & \frac{\partial f_2}{\partial g_2} \end{bmatrix} \begin{bmatrix} \frac{\partial g_1}{\partial x} \\ \frac{\partial g_2}{\partial x} \end{bmatrix}$. To broaden to multiple parameters, vector x, we now multiply

two full matrix Jacobians: $\frac{\partial}{\partial \mathbf{x}} \mathbf{f}(\mathbf{g}(\mathbf{x})) = \begin{bmatrix} \frac{\partial f_1}{\partial g_1} & \frac{\partial f_1}{\partial g_2} & \cdots & \frac{\partial f_1}{\partial g_k} \\ \frac{\partial f_2}{\partial g_1} & \frac{\partial f_2}{\partial g_2} & \cdots & \frac{\partial f_2}{\partial g_k} \\ \frac{\partial f_m}{\partial g_1} & \frac{\partial f_m}{\partial g_2} & \cdots & \frac{\partial f_m}{\partial g_k} \end{bmatrix} \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \cdots & \frac{\partial g_2}{\partial x_n} \\ \frac{\partial g_k}{\partial x_1} & \frac{\partial g_k}{\partial x_2} & \cdots & \frac{\partial g_k}{\partial x_n} \end{bmatrix}$

- Most often, the Jacobian reduces to a diagonal matrix whose elements are the single variable chain rule values
- A summary to get to the Jacobian:



Applications

Low Rank Matrix Approximations

- The following are equivalent definitions for the rank of a matrix B to be k
  - The largest linearly independent subset of columns
  - The largest linearly independent subset of rows
  - B can written as, or "factored into," the product of long and skinny (n×k) matrix Y_k and a short and long (k×d) matrix Z_k^T. Think outer product
- Idea is to approximate our matrix with a matrix of rank k - useful for compression or denoising.
- For every n x d A with rank target k and given a rank-k n x d matrix B, $\|\mathbf{A} - \mathbf{A}_k\|_F \leq \|\mathbf{A} - \mathbf{B}\|_F$
- For $X = A - A_k$, $||X||_F$ measures the discrepancy between A and its approximation. Want to find the A_k that minimizes this distance.

Using SVD

- $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, for U nxn orthogonal matrix, V dxd orthogonal matrix, S nxd matrix of nonnegative entries with diagonal entries sorted from high to low. Columns of U are left singular values of A, V are right singular values of A. Entries of S are singular values of A.

- Choosing a rank-k matrix boils down to choosing a set of k basis vectors. What vectors to choose? The SVD gives us a representation of A as a linear combination of sets of vectors ordered by importance!
- Given n x d matrix A and target rank k, we do the following
    - Compute SVD $A = USV^T$. Keep only the top k right singular vectors: set V_k^T equal to the first k rows of V^T
    - Keep only the top k left singular vectors: first k columns of U
    - Keep only the top k singular values: first k rows / columns of S, the k largest singular values of A
- Low rank approximation is then $\mathbf{A}_k = \mathbf{U}_k\mathbf{S}_k\mathbf{V}_k^T$. This now takes $O(k(n+d))$ space to store instead of $O(nd)$
- This is akin to approximating A in terms of k "concepts" where the singular values express the signal strength of these concepts, rows of V^T and columns of U express the canonical row/column associated with each concept and rows of U and cols of V^T express each row / column of A as a linear combination of the canonical rows
- Alternate approach: From U and V can construct rank-1 matrices via $\boldsymbol{A}_i := \boldsymbol{u}_i \boldsymbol{v}_i^\top$. A matrix of rank r can be written as a sum of rank 1 matrices. Construct rank-1 matrices using each singular value, then consider a rank-2 (etc) matrix $\widehat{\boldsymbol{A}}(2) = \sigma_1 \boldsymbol{A}_1 + \sigma_2 \boldsymbol{A}_2$. Stop the combination when we have a close enough approximation.

Choosing K

- Ideally, guidance from eigenvalues of A^TA or singular values of A. If top few are big and rest are small, cut off is relatively obvious
- Often choose k st the sum of the top k eigenvalues is at least c times as big as the sum of othe eigenvalues.
- The effect of small eigenvalues on matrix products is small. Thus, it seems plausible that replacing these small eigenvalues by zero will not substantially alter the product

Application: Fill in missing values

- A is a matrix of Netflix customers and movie ratings. A reasonable assumption that makes the problemmore tractable is that the matrix to be recovered is well-approximated by a low-rank matrix.
- If there aren't too many missing entries, and if the matrix to be recoveredis approximately low rank, then the following application of the SVD can yield a good guessas to the missing entries
- Fill in missing entries with suitable default values to obtain a matrix $\hat{A}$ then compute the best rank-k approximation of $\hat{A}$

Markov Matrices

- Example: $A = \begin{bmatrix} .1 & .01 & .3 \\ .2 & .99 & .3 \\ .7 & 0 & .4 \end{bmatrix}$
- All entries ≥ 0 and all columns add to 1
- Property 2 guarantees 1 is an eigenvalue, all other eigenvals must be less than 1
- Taking $A - 1\lambda$ creates singular matrix, cols add to 0.

Fast Fourier Transform

Differential Equations

Matrix Exponentials