MA233 Introduction to Big Data Science

Mathematical Preliminary

Zhen Zhang

Southern University of Science and Technology

Outlines

Linear Algebra

Probability and Information Theory

Statistics and Machine Learning

References

Inner Product and Euclidean Norm

For $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, their inner product is defined as

$$< \mathbf{x}, \mathbf{y} >= \sum_{i=1}^{n} x_i y_i = \mathbf{x}^T \mathbf{y}.$$

It satisfies

1. (Commutativity) $< \mathbf{x}, \mathbf{y} >=< \mathbf{y}, \mathbf{x} >$;
2. (Scalar Multiplication) $< \lambda \mathbf{x}, \mathbf{y} >= \lambda < \mathbf{x}, \mathbf{y} >=< \mathbf{x}, \lambda \mathbf{y} >$;
3. (Bilinearity) $< \mathbf{x} + \mathbf{y}, \mathbf{z} >=< \mathbf{x}, \mathbf{z} > + < \mathbf{y}, \mathbf{z} >$, $< \mathbf{x}, \mathbf{y} + \mathbf{z} >=< \mathbf{x}, \mathbf{y} > + < \mathbf{x}, \mathbf{z} >$;
4. (Positivity) $< \mathbf{x}, \mathbf{x} >\geqslant 0$, and $< \mathbf{x}, \mathbf{x} >= 0$ iff $\mathbf{x} = \mathbf{0}$.

The Euclidean norm (l_2-norm) is $\|\mathbf{x}\| = \sqrt{< \mathbf{x}, \mathbf{x} >}$.

Linear Independency and Orthogonality

- Linear Independency :
  A set of vectors $U = \{bx_1, \ldots, \mathbf{x}_k\}$ is linearly independent if for $\forall i$, $\mathbf{x}_i$ does not lie in the space spanned by $\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \mathbf{x}_k$. We say U spans a subspace V if V is the span of the vectors in U. U is a basis of V if it is both independent and spans V. The dimension of V is the size of a basis of V (i.e., the number of linearly independent vectors in U).
- Orthogonality :
  We say that U is an orthogonal set if for all $i \neq j$, $< \mathbf{x}_i, \mathbf{x}_j >= 0$. We say that U is an orthonormal set if it is orthogonal and if for every i, $\|\mathbf{x}_i\| = 1$.

Gram-Schmidt Orthogonalization

Given a set of linear independent vectors $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$, we can apply Gram-Schmidt orthogonalization to obtain an orthonormal set $\{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ which have the same span as $\mathrm{span}\, V$. The procedure is as follows :

1. Let $\mathbf{u}_1 = \mathbf{v}_1 / \|\mathbf{v}_1\|$;
2. For $j = 2$ to k, project $\mathbf{v}_j$ onto $\mathrm{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_{j-1}\}$ and find the perpendicular part $\tilde{\mathbf{u}}_j = \mathbf{v}_j - \sum_{i=1}^{j-1} < \mathbf{u}_i, \mathbf{v}_j > \mathbf{u}_i$, then normalize it to be $\mathbf{u}_j = \tilde{\mathbf{u}}_j / \|\tilde{\mathbf{u}}_j\|$;

This procedure is summarized in the matrix form : $Q = AP$, where $Q = (\mathbf{u}_1 \cdots \mathbf{u}_k) \in \mathbb{R}^{k \times k}$ is an orthogonal matrix whose columns are given by $\mathbf{u}_i$'s, $A = (\mathbf{v}_1 \cdots \mathbf{v}_k) \in \mathbb{R}^{k \times k}$ is a nonsingular matrix whose columns are given by $\mathbf{v}_i$'s, and $P \in \mathbb{R}^{k \times k}$ is an upper tridiagonal matrix whose upper tridiagonal (i, j)-entry is given by $< \mathbf{u}_i, \mathbf{v}_j >$. This is known as the QR factorization : $A = QR$ where $R = P^{-1}$.

Concepts in Matrix

- Kernel and Range :
  Given a matrix $A \in \mathbb{R}^{n \times d}$, the range of A (Range(A)) is the span of its columns and the kernel of A (Ker(A)) is the subspace of all vectors that satisfy $A\mathbf{x} = \mathbf{0}$. The rank of A is the dimension of its range and is denoted by $\mathrm{rank}(A)$ or $\mathrm{r}(A)$ for short.
- Symmetric and Definite Matrix :
  A is symmetric if $A = A^T$. A symmetric matrix $A \in \mathbb{R}^{d \times d}$ is positive definite if for all $\mathbf{x} \in \mathbb{R}^d$, $< \mathbf{x}, A\mathbf{x} >\geqslant 0$, and equality holds if and only if ("iff") $\mathbf{x} = \mathbf{0}$. This definition can be relaxed to give semidefiniteness : A symmetric matrix $A \in \mathbb{R}^{d \times d}$ is positive semidefinite if for all $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{x}^T A \mathbf{x} \geqslant 0$. In particular, all the eigenvalues of a positive definite (resp. semidefinite) matrix are positive (resp. nonnegative). And $A = BB^T$ for some matrix B. (See next slides for eigen-decomposition)

Eigenvalues and Eigenvectors

Let $A \in \mathbb{R}^{d\times d}$ be a squared matrix. A nonzero vector $\mathbf{x}\mathbb{R}^d$ is an eigenvector of A with a corresponding eigenvalue λ if $A\mathbf{x} = \lambda\mathbf{x}$.

Theorem

(Eigen-decomposition or Spectral Decomposition) If $A \in \mathbb{R}^{d\times d}$ is a symmetric matrix of rank k, then there exists an orthogonal basis of $\mathbb{R}^d$, $\mathbf{x}_1, \ldots, \mathbf{x}_d$, such that each $\mathbf{x}_i$ is an eigenvector of A. Furthermore, A can be written as $A = \sum_{i=1}^{d} \lambda_i \mathbf{x}_i \mathbf{x}_i^T$, where each λ_i is the eigenvalue corresponding to the eigenvector $\mathbf{x}_i$. In matrix form, this is $A = UDU^T$, where the columns of U are the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_d$, and $D = \mathrm{diag}\{\lambda_1, \ldots, \lambda_d\}$ is a diagonal matrix. Finally, $\mathrm{r}(A)$ is the number of nonzero λ_i's, and the corresponding eigenvectors span the range of A. The eigenvectors corresponding to the zero eigenvalues span the null space of A.

Singular Values Decomposition (SVD)

Let $A \in \mathbb{R}^{m\times n}$ be a matrix of rank r. Unit (nonzero) vector $\mathbf{v} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^m$ are called right and left singular vectors of A with corresponding singular values σ if $A\mathbf{v} = \sigma\mathbf{v}$ and $\mathbf{u}^T A = \sigma\mathbf{u}^T$.

Theorem

(SVD) Let $A \in \mathbb{R}^{m\times n}$ be a matrix of rank r. Then there exist orthonormal sets of right and left singular vectors of A, say $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ and $\{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$ respectively, and the corresponding singular values $\sigma_1, \ldots, \sigma_r$, such that $A = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T$. In matrix form, this is $A = UDV^T$, where the columns of U are the vectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$, the columns of V are the vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$, and $D = \mathrm{diag}\{\sigma_1, \ldots, \sigma_d\}$ is a diagonal matrix.

Corollary

The squared matrices $A^T A \in \mathbb{R}^{n\times n}$ and $AA^T \in \mathbb{R}^{m\times m}$ have (a subset of) the eigenvectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ and $\{\mathbf{u}_1, \ldots, \mathbf{u}_r\}$ respectively, corresponding the the same eigenvalues $\sigma_1^2, \ldots, \sigma_r^2$.

Reyleigh Quotient

Theorem

Let $A \in \mathbb{R}^{m\times n}$ be a matrix of rank r. Define $\mathbf{v}_1 = \underset{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1}{\arg\max} \|A\mathbf{v}\|$, $\mathbf{v}_2 = \underset{\substack{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1 \\ <\mathbf{v},\mathbf{v}_1>=0}}{\arg\max} \|A\mathbf{v}\|, \ldots, \mathbf{v}_r = \underset{\substack{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1 \\ \forall i<r, <\mathbf{v},\mathbf{v}_i>=0}}{\arg\max} \|A\mathbf{v}\|$. *Then $\mathbf{v}_1, \ldots, \mathbf{v}_r$ is an orthonormal set of right singular vectors of A.*

Remark :(Reyleigh Quotient) If $A \in \mathbb{R}^{n\times n}$ is a squared matrix, then its eigenvalues can be found as the solution to the following optimization problems :

$$\lambda_1 = \max_{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1} \mathbf{v}^T A\mathbf{v}, \quad \lambda_2 = \max_{\substack{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1 \\ <\mathbf{v},\mathbf{v}_1>=0}} \mathbf{v}^T A\mathbf{v},$$

$$\ldots, \quad \lambda_n = \max_{\substack{\mathbf{v}\in\mathbb{R}^n:\|\mathbf{v}\|=1 \\ \forall i<n, <\mathbf{v},\mathbf{v}_i>=0}} \mathbf{v}^T A\mathbf{v}.$$

Power Method - Dominant Eigenvalue

Assume the eigenvalues of A can be sorted according to their magnitudes : $|\lambda_1| > |\lambda_2| \geqslant |\lambda_3| \geqslant \cdots \geqslant |\lambda_n| \geqslant 0$. If The corresponding eigenvectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ form a basis of $\mathbb{R}^n$, then any vector can be expressed as $\mathbf{x} = \sum_{i=1}^{n} \beta_i \mathbf{v}_i$. Multiplying $\mathbf{x}$ by A on the left for n times, we have an idea

$$A^k\mathbf{x} = \sum_{i=1}^{n} \beta_i \lambda_i^k \mathbf{v}_i = \lambda_1^k \sum_{i=1}^{n} \beta_i \left(\frac{\lambda_i}{\lambda_1}\right)^k \mathbf{v}_i \sim \lambda_1^k \beta_1 \mathbf{v}_1, \quad k \to \infty$$

1. For any nonzero vector $\mathbf{x}$, let $\mathbf{y}^{(0)} = \mathbf{x}$;
2. For $k = 0, 1, \ldots$: compute the smallest integer p_k such that satisfying $y_{p_k}^{(k)} = \|\mathbf{y}^{(k)}\|_\infty$, then compute $\mathbf{x}^{(k)} = \mathbf{y}^{(k)}/y_{p_k}^{(k)}$, $\mathbf{y}^{(k+1)} = A\mathbf{x}^{(k)}$, $\mu^{(k+1)} = y_{p_k}^{(k+1)}$.

It can be shown that $\lim_{k\to\infty} \mu^{(k)} = \lambda_1$ and $\lim_{k\to\infty} \mathbf{x}^{(k)} = \mathbf{v}_1/\|\mathbf{v}_1\|_\infty$.
Other methods : QR factorization, Householder transformations

Linear Systems

A system of m linear algebraic equations in n unknown variables can be written in the matrix form : $A\mathbf{x} = \mathbf{b}$, where $A \in \mathbb{R}^{m\times n}$, $\mathbf{x} \in \mathbb{R}^n$, and $\mathbf{b} \in \mathbb{R}^m$. This system has solutions iff $\mathrm{r}([A, \mathbf{b}]) = \mathrm{r}(A)$.

Theorem (Solvability Condition)

1. *If $\mathrm{Ker}(A) = \{0\}$, then $A\mathbf{x} = \mathbf{b}$ either has a unique solution or has no solution. It has a solution iff $\mathbf{b} \perp \mathrm{Ker}(A^T)$.*
2. *If $\mathrm{Ker}(A) \neq \{0\}$, then $A\mathbf{x} = \mathbf{b}$ either has infinitely many solutions or has no solution. It has a solution iff $\mathbf{b} \perp \mathrm{Ker}(A^T)$.*

If $A \in \mathbb{R}^{n\times n}$ is a square matrix, we have a simple rule : the system has a unique solution iff $\det A \neq 0$. If the solution exists, we can solve it by $\mathbf{x} = A^{-1}\mathbf{b}$, where A^{-1} is the inverse of A satisfying $A^{-1}A = AA^{-1} = I$.
Moreover, we can find it by Cramer's rule : $x_i = \frac{\det A_i}{\det A}$, where A_i is the matrix obtained from A by replacing its i-th column with $\mathbf{b}$. The direct application of this formula requires $O(n!)$ arithmetic operations to find $\det A$, which is unacceptable for large n.

Gaussian Elimination

Gaussian Elimination is an algorithm that can reduce the computational complexity of solving linear systems to $O(n^3)$. It is equivalent to perform an elementary row transformation for A to obtain an upper or lower triangular matrix.

Another way to view Gaussian elimination is the LU decomposition : The k-th row transformation can be represented by a left multiplication by $M^{(k)}$, where $M^{(k)}$ is a lower triangular matrix with its diagonal entries being all 1's; after n operations, A is transformed to an upper triangular matrix U, i.e., $M^{(n)} \ldots M^{(2)} M^{(1)} A = U$; since the inverse of a lower triangular matrix is also a lower triangular matrix, we have $A = LU$, where $L = (M^{(1)})^{-1}(M^{(2)})^{-1} \cdots (M^{(n)})^{-1}$ with its diagonal entries being all 1's.

LU Decomposition

Theorem

An $n \times n$ nonsingular matrix A can be decomposed uniquely in the form $A = LU$, where

$$L = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ l_{n,1} & \cdots & l_{n,n-1} & 1 \end{pmatrix}, U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1,n} \\ 0 & u_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & u_{n-1,n} \\ 0 & \cdots & 0 & u_{n,n} \end{pmatrix}.$$

The computational complexity for LU decomposition is $O(n^3)$. If A is symmetric, we have Cholesky decomposition $A = LL^T$, where L is a lower diagonal matrix.

Iterative Solver

We introduce two commonly used iterative methods : Jacobi iteration and Gauss-Seidel iteration. First we write $A = D - L - U$, where $D = \mathrm{diag}\{a_{11}, \ldots, a_{nn}\}$, $L = \{l_{ij}\}$ and $U = \{u_{ij}\}$ are the lower and upper diagonal parts of $-A$ respectively. That means $l_{ij} = -a_{ij}$ for $i > j$ and 0 for $i \leqslant j$, $u_{ij} = -a_{ij}$ for $i < j$ and 0 for $i \geqslant j$.

- Jacobi iteration : Rewrite the linear system as $D\mathbf{x} = (L + U)\mathbf{x} + \mathbf{b}$, if D^{-1} exists ($a_{ii} \neq 0$), then we can build the iteration $\mathbf{x}^{(k)} = D^{-1}(L + U)\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b}$, $k = 1, 2, \ldots$
- Gauss-Seidel iteration : Rewrite the linear system as $(D - L)\mathbf{x} = U\mathbf{x} + \mathbf{b}$, if $(D - L)^{-1}$ exists ($a_{ii} \neq 0$), then we can build the iteration $\mathbf{x}^{(k)} = (D - L)^{-1}U\mathbf{x}^{(k-1)} + (D - L)^{-1}\mathbf{b}$, $k = 1, 2, \ldots$

Both are easy to implement in component form and can be written as $\mathbf{x}^{(k)} = T\mathbf{x}^{(k-1)} + \mathbf{c}$, with $T = D^{-1}(L + U)$ for Jacobi iteration and $(D - L)^{-1}U$ for Gauss-Seidel iteration. (Fixed point iteration !)

Vector Norms

Vector Norm is a non-negative real-valued function on $\mathbb{R}^n$, usually denoted by $\|\cdot\| : \mathbb{R}^n \to \mathbb{R}$, with the following properties :

1. (Positivity) $\|\mathbf{x}\| \geqslant 0$ for all $\mathbf{x} \in \mathbb{R}^n$; $\|\mathbf{x}\| = 0$ iff $\mathbf{x} = \mathbf{0}$;
2. (Homogeneity) $\|\alpha\mathbf{x}\| = |\alpha|\|\mathbf{x}\|$ for $\forall \alpha \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{R}^n$;
3. (Triangle Inequality) $\|\mathbf{x} + \mathbf{y}\| \leqslant \|\mathbf{x}\| + \|\mathbf{y}\|$ for $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

Examples :

- l_2-norm : $\|\mathbf{x}\|_2 = (\sum_{i=1}^{n} x_i^2)^{\frac{1}{2}}$;
- l_1-norm : $\|\mathbf{x}\|_1 = \sum_{i=1}^{n} |x_i|$;
- l_∞-norm : $\|\mathbf{x}\|_\infty = \max_{1 \leqslant i \leqslant n} |x_i|$;

Theorem

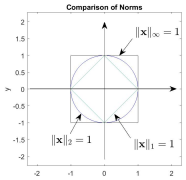
Define l_p-norm as $\|\mathbf{x}\|_p = (\sum_{i=1}^{n} x_i^p)^{\frac{1}{p}}$, it is really a norm for $p \leqslant 1$.

Vector Norms (Cont')

Remark : i) l_p-norm is not a norm for $0 < p \leqslant 1$, since the triangular inequality is not satisfied. It is called semi-norm.
ii) Useful to define l_0-norm : $\|\mathbf{x}\|_0 = \#\{1 \leqslant i \leqslant n : x_i \neq 0\}$.
Induced Distances : $\mathrm{dist}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$, e.g., l_2-distance is $\|\mathbf{x} - \mathbf{y}\|_2 = (\sum_{i=1}^{n} (x_i - y_i)^2)^{\frac{1}{2}}$.

Theorem

$\forall \mathbf{x} \in \mathbb{R}^n$, $\|\mathbf{x}\|_\infty \leqslant \|\mathbf{x}\|_2 \leqslant \|\mathbf{x}\|_1$.



Matrix Norms

Matrix Norm is a non-negative real-valued function on $\mathbb{R}^{n \times m}$, usually denoted by $\|\cdot\| : \mathbb{R}^{n \times m} \to \mathbb{R}$, with the following properties :

1. (Positivity) $\|A\| \geqslant 0$ for all $A \in \mathbb{R}^{n \times m}$; $\|A\| = 0$ iff $A = 0$;
2. (Homogeneity) $\|\alpha A\| = |\alpha|\|A\|$ for $\forall \alpha \in \mathbb{R}$ and $A \in \mathbb{R}^{n \times m}$;
3. (Triangle Inequality) $\|A + B\| \leqslant \|A\| + \|B\|$ for $\forall A, B \in \mathbb{R}^{np \times m}$;
4. $\|AB\| \leqslant \|A\|\|B\|$ for $\forall A, B \in \mathbb{R}^{n \times m}$.

Theorem

If $\|\cdot\|$ is a vector norm on $\mathbb{R}^n$, then $\|A\| = \max_{\|\mathbf{x}\|=1} \|A\mathbf{x}\| = \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$ is a matrix norm (called natural norm).

Corollary

$\|A\mathbf{x}\| \leqslant \|A\|\|\mathbf{x}\|$ *for* $\forall A \in \mathbb{R}^{n \times m}$ *and* $\mathbf{x} \in \mathbb{R}^n$.

Matrix Norm (Cont')

Examples :

- l_1-norm : $\|A\|_1 = \max_{1 \leqslant j \leqslant n} \sum_{i=1}^{n} |a_{ij}|$;
- l_∞-norm : $\|A\|_\infty = \max_{1 \leqslant i \leqslant n} \sum_{j=1}^{n} |a_{ij}|$;

l_2-norm is not trivial. For a symmetric matrix A, define its spectral radius as $\rho(A) = \max_{1 \leqslant i \leqslant n} \lambda_i$, where $\lambda_i (i = 1, \ldots, n)$ are the eigenvalues of A. Then

Theorem

1. $\|A\|_2 = \sqrt{\rho(A^T A)}$;
2. $\rho(A) \leqslant \|A\|$ *for any natural norm* $\|\cdot\|$.

Theorem (Convergence of Jacobi and Gauss-Seidel Iterations)

The Jacobi and Gauss-Seidel iterations converge to the unique solution of $\mathbf{x} = T\mathbf{x} + \mathbf{c}$ iff $\rho(T) < 1$. Moreover, we have the error estimate $\|\mathbf{x} - \mathbf{x}^{(k)}\| \leqslant \frac{\|T\|^k}{1 - \|T\|}\|\mathbf{x}^{(1)} - \mathbf{x}^{(0)}\|$.

Matrix Calculus

By convention, the lowercase letter a denotes a scalar, the bold letter $\mathbf{x} = (x_1, \ldots, x_n)^T$ denotes a column vector, and the uppercase letter $A = (a_{ij})$ denotes an $m \times n$ matrix. Assume $\mathbf{x}$ (or x) is independent variables, $\mathbf{a}$, $\mathbf{b}$, etc. are constant vectors, A, B, etc. are constant matrices, $f(x)$, $g(x)$, $\mathbf{u}(\mathbf{x})$, and $\mathbf{v}(\mathbf{x})$ are (scalar or vector valued) functions of $\mathbf{x}$ (or x)

- Vector-by-vector formula : (resulting in matrix $\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = (\frac{\partial y_i}{\partial x_j})$)
  1. Linear vector-valued functions : $\frac{\partial \mathbf{a}}{\partial \mathbf{x}} = 0$, $\frac{\partial (A\mathbf{x})}{\partial \mathbf{x}} = A$, $\frac{\partial (\mathbf{x}^T A)}{\partial \mathbf{x}} = A^T$,
  2. Nonlinear vector-valued functions : $\frac{\partial \mathbf{u}}{\partial \mathbf{x}} = (\frac{\partial u_i}{\partial x_j})$ is Jacobian, $\frac{\partial (a\mathbf{u}(\mathbf{x}) + b\mathbf{v}(\mathbf{x}))}{\partial \mathbf{x}} = a\frac{\partial \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} + b\frac{\partial \mathbf{v}(\mathbf{x})}{\partial \mathbf{x}}$, $\frac{\partial (f(\mathbf{x})\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}} = f(\mathbf{x})\frac{\partial \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} + \mathbf{u}\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$, $\frac{\partial (A\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}} = A\frac{\partial \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}}$
  3. Chain rule : $\frac{\partial \mathbf{g}(\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}} = \frac{\partial \mathbf{g}(\mathbf{u})}{\partial \mathbf{u}} \frac{\partial \mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}}$,

Matrix Calculus

- Scalar-by-vector : (resulting in row vector $\frac{\partial y}{\partial \mathbf{x}} = (\nabla_{\mathbf{x}} y)^T$) Some of the formula can be obtained from the previous page by letting the numerator be of dimension one, the others are :
  1. Inner product : $\frac{\partial (\mathbf{a}^T \mathbf{x})}{\partial \mathbf{x}} = \frac{\partial (\mathbf{x}^T \mathbf{a})}{\partial \mathbf{x}} = \mathbf{a}^T$, $\frac{\partial \mathbf{a}^T \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{a}^T \frac{\partial (\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}}$, $\frac{\partial (\mathbf{u}(\mathbf{x})^T A \mathbf{v}(\mathbf{x}))}{\partial \mathbf{x}} = \mathbf{u}^T A \frac{\partial (\mathbf{v}(\mathbf{x}))}{\partial \mathbf{x}} + \mathbf{v}^T A^T \frac{\partial (\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}}$
  2. Quadratic forms : $\frac{\partial (\mathbf{x}^T A \mathbf{x})}{\partial \mathbf{x}} = \mathbf{x}^T (A + A^T)$, $\frac{\partial (\mathbf{x}^T A \mathbf{x})}{\partial \mathbf{x}} = 2\mathbf{x}^T A$ if A is symmetric, $\frac{\partial (\mathbf{a}^T \mathbf{x} \mathbf{x}^T \mathbf{b})}{\partial \mathbf{x}} = \mathbf{x}^T (\mathbf{a}\mathbf{b}^T + \mathbf{b}\mathbf{a}^T)$ $\frac{\partial (A\mathbf{x}+\mathbf{b})^T C (D\mathbf{x}+\mathbf{e})}{\partial \mathbf{x}} = (D\mathbf{x} + \mathbf{e})^T C^T A + (A\mathbf{x} + \mathbf{b})^T CD$
  3. l_2 norm : $\frac{\partial \|\mathbf{x}-\mathbf{a}\|}{\partial \mathbf{x}} = \frac{(\mathbf{x}-\mathbf{a})^T}{\|\mathbf{x}-\mathbf{a}\|}$
  4. 2nd order derivative (resulting in a matrix) : $\frac{\partial^2 (\mathbf{x}^T A \mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} = (A + A^T)$, $\frac{\partial^2 (\mathbf{x}^T A \mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} = 2A$ if A is symmetric, $\frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x} \partial \mathbf{x}^T} = H = (\frac{\partial f}{\partial x_i \partial x_j})$ is the Hessian matrix.

Trace and Frobenius inner product

Trace is defined as the sum of the diagonal entries in a matrix : $\mathrm{tr}(A) = \sum_{i=1}^n a_{ii}$

- $\mathrm{tr}(A) = \mathrm{tr}(A^T)$
- $\mathrm{tr}(AB) = \mathrm{tr}(BA)$, $\mathrm{tr}(ABC) = \mathrm{tr}(CAB) = \mathrm{tr}(BCA)$
- $\frac{\partial \mathrm{tr}(AB)}{\partial A} = B^T$, $\frac{\partial \mathrm{tr}(ABA^T C)}{\partial A} = CAB + C^T AB^T$
- $a = \mathrm{tr}(a)$ for scalar a, as a result, $< \mathbf{x}, \mathbf{y} >= \mathrm{tr}(\mathbf{x}^T \mathbf{y}) = \mathrm{tr}(\mathbf{y}\mathbf{x}^T)$ (useful formula)

The Frobenius inner product is defined for matrices : $< A, B >_F = \mathrm{tr}(AB^T) = \sum_{i,j=1}^n a_{ij} b_{ij}$. The induced norm is called Frobenius norm : $\|A\|_F = \sqrt{\mathrm{tr}(AA^T)} = \sqrt{\sum_{i,j=1}^n a_{ij}^2}$.

A last useful formula : $\frac{\mathrm{d}}{\mathrm{d}t} \log \det(A(t)) = \mathrm{tr}(A(t)^{-1} A'(t))$

Outlines

Linear Algebra

Probability and Information Theory

Statistics and Machine Learning

References

Basics

Three Sources of Uncertainty

1. Inherent randomness in the system being modeled. (e.g. quantum mechanics, particles are probabilitic.)
2. Incomplete observability (e.g. Monty Hall problem. 3 dogs, one of which has a bonus behind it.)
3. Incomplete modeling (e.g. linear regression.)

Random Variables (r.V.)

X : r.V., its values x

$X = x$ happens with a certain probability $p(X = x)$. Range of X may be discrete or continuous.

Discrete variables & Probability mass functions (PMF)

$P(X = x) = p(x)$ or $X \sim p(x)$, $0 \le p(x) \le 1$.

Joint probability distribution : $p(X = x, Y = y) = p(x, y)$

Properties :

(1) $Dom(P) = Range(X) =$ set of all possible states of X.

(2) $\forall x \in Range(X)$, $0 \le p(x) \le 1$.

(3) $\sum_{x \in Range(X)} p(x) = 1$ (Normalized)

e.g. uniform distribution : $P(X = x_i) = \frac{1}{k}, \quad i = 1, \ldots, k$.

Continuous variables & Probability density function (p.d.f)

Properties :

(1) $Dom(P) = Range(X) =$ set of all possible states of X.

(2) $\forall x \in Range(X), p(x) \geq 0$. (No need for $p(x) \leq 1$)

(3) $\int p(x)dx = 1$.

$p(x)$ does not give the probability of a specific state x, but the prob of X landing inside the interval $[x, x + \delta x)$ with an infinitesimal δx. ($prob = p(x)\delta x$)
e.g. uniform distribution function on an interval $[a, b]$:

$$u(x; a, b) = \begin{cases} 0 & x \notin [a, b] \\ \frac{1}{b-a} & x \in [a, b] \end{cases} \qquad x \sim U(a, b)$$

Marginal Probability & Conditional Probability

Marginal Probability :
prob over a subset of all variables.
discrete : $p(x) = P(X = x) = \sum_y P(X = x, Y = y) = \sum_y p(x, y)$
continuous : $p(x) = \int p(x, y)dy$

Conditional Probability :
prob of some event given some other events have happened.
$P(Y = y|X = x) = \frac{P(X=x, Y=y)}{P(X=x)}$
(well-defined if $P(X = x) > 0$)

Chain Rules :

$$P(X^{(1)}, \cdots, X^{(n)}) = P(X^{(1)})P(X^{(2)}|X^{(1)})P(X^{(3)}|X^{(1)}, X^{(2)}) \cdots P(X^{(n)}|X^{(1)}, \cdots, X^{(n-1)})$$

Independence & Conditional Independence

X and Y are independent, if
$\forall x, y,\ P(X = x, Y = y) = P(X = x)P(Y = y)$

X, Y are conditionally independent given Z if $\forall x, y, z$,
$P(X = x, Y = y|Z = z) = P(X = x|Z = z)P(Y = y|Z = z)$.

short-hand notation : $X \perp Y$, $X \perp Y|Z$.

Expectation Variance and Covariance

$E_{x\sim p}[f(x)] = \begin{cases} \sum_x p(x)f(x) \\ \int p(x)f(x)dx \end{cases}$,
$E[\alpha f(x) + \beta g(x)] = \alpha Ef(x) + \beta Eg(x)$
$Var[f(x)] = E[(f(x) - Ef(x))^2]$, $std[f(x)] = \sqrt{Var[f(x)]}$
$Cov[f(X), g(Y)] = E[(f(x) - Ef(x))(g(Y) - Eg(Y))]$
measures how much two variable are linearly related to each other as well as their scales.
$Cov[f(X), g(Y)] = \frac{Cov[f(X),g(Y)]}{std[f(X),g(Y)]} \in [-1, 1]$ normalized
$X \perp Y \Rightarrow Cov[X, Y] = 0$ or $Corv[X, Y] = 0$
$''\Leftarrow''$ is only true when $X, Y \sim$ normal distribution
Covariance Matrix : $Cov(\vec{X}) = (Cov(X_i, Y_j))_{1\leq i,j\leq n}$
Diagonal $Cov(X_i, X_i) = Var(X_i)$

Common Prob Distributions

1) Bernoulli : binary r.v. one parameter $\phi \in [0, 1]$
$p(X = 1) = \phi$, $p(X = 0) = 1 - \phi$ or
$p(X = x) = \phi^x(1 - \phi)^x \quad x \in \{0, 1\}$,
$E_X[X] = \phi \quad Var_X[X] = \phi(1 - \phi)$.

2) Muttinoulli (Categorical)
$Range(X) = \{0, 1, \cdots, k\}$
$p(X = i) = p_i \quad \sum_{i=1}^{k} p_i = 1$ only $k - 1$ parameters.

3) Gaussian (Normal)
$N(x; \mu, \sigma^2) = \sqrt{\frac{1}{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}(x - \mu)^2)$
$E(X) = \mu, \quad Var[X] = \sigma^2 > 0$
$\beta = \sigma^{-2}$=precision.

4) ..., continue after the importance of Gaussian.

Why are Gaussian important

(1) Central Limit Theorem (many versions)
$\{X_i\}_{i=1}^n$ i.i.d. (independent and identically distributed),
$S_n = \frac{1}{n}(X_1 + \cdots + X_n)$. (Law of Large numbers
$\Rightarrow S_n \to EX_1 = \mu$ as $n \to \infty$)
stronger result : $\sqrt{n}(S_n - \mu) \Rightarrow N(0, \sigma^2)$ as $n \to \infty$),in distribution $\sigma^2 = Var[X_1]$.
$\lim_{n\to\infty} P_r[\sqrt{n}(S_n - \mu) \leq z] = \Phi(\frac{z}{\sigma})$, where $\Phi(x)$ is c.d.f of $N(0, 1)$.

Why are Gaussian important

(2) Information theory : Gaussian encodes the maximum amout of uncertainty
$H[x] = E_{x\sim p}[-\log p(x)] = -\int p(x)\log p(x)dx$.
$\max_p E[X]$ s.t.
$\lambda_1 \quad \int p(x)dx = 1 \Rightarrow p^*(x) = \sqrt{\frac{1}{2\pi\sigma^2}}\exp(-\frac{1}{2\sigma^2}(x-\mu)^2)$
$\lambda_2 \quad \int xp(x)dx = \mu$
$\lambda_3 \quad \int (x-\mu)^2 p(x)dx = \sigma^2$,
$p(x) = \exp\{\lambda_1 - 1 + \lambda_2(x-\mu) + \lambda_3(x-\mu)^3\}$, $\lambda_2 = 0$,
λ_1 : *nomalization constant*, $\lambda_3 = -1/(2\sigma^2)$.

Multivariate normal :
$N(\vec{x};\vec{\mu},\vec{\Sigma}) = \sqrt{\frac{1}{(2\pi)^n \det(\Sigma)}}\exp(-\frac{1}{2}(\vec{x}-\vec{\mu})^T\Sigma^{-1}(\vec{x}-\vec{\mu}))$
$\Sigma^{-1} = \Omega$ precision matrix.
$Cov(\vec{x}) = \Sigma, \quad E\vec{x} = \vec{\mu}$

Common Prob Distributions (Continued)

4) Exponential and Laplace (Doubly exponential)
exponential : $p(x;\lambda) = \lambda\perp_{x\geq 0}\exp(-\lambda x)$
Laplace $(x;\mu,r) = \frac{1}{2r}\exp(-\frac{|x-\mu|}{r})$.
5) Dirac and Empirical
Dirac : $p(x) = \delta(x-\mu)$ which puts prob $\frac{1}{m}$ on each of $x^{(i)}$.
(samples), i.e. $P(X = x^{(i)}) = \frac{1}{m}$, $i = 1,\cdots,m$.

Common Prob Distributions (Continued)

6) Mixture :
$P(x) = \sum P(c=i)P(x|c=i)$, where $P(c)$ is multimoulli.
Empirical is mixture with Dirac.
Latent variable c, $P(X|c)$ relates the latent variable c to the visible variables X.
e.g. Gaussian Mixtures $P(x|c=i)$ is Gaussian for $\forall i$,
$\sim N(x;\mu^{(i)},\Sigma^{(i)})$.
$P(c=i)$ is prior prob. $P(c|x)$ is posterior prob.
Gaussian mixture model is universal approximator of density in the sense that any smooth density can be approximated with any specific non-zero amount of error by a Gaussian mixture model with enough components.

Bayes' Rule

$P(x|y) = \frac{P(x)P(y|x)}{P(y)}$ i.e $P(x|y)P(y) = P(x,y) = P(x)P(y|x)$
$P(y) = \sum_x P(x)P(y|x)$
$P(x)$: prior ; $P(x|y)$: posterior

Transformations

X, Y two r.v.s. $Y = g(X)$, g is invertible
p.d.f. of X is P_x, p.d.f. of Y is P_y, then $P_k(\vec{y}) = P_x(g^{-1}(\vec{y}))\left|\frac{\partial\vec{x}}{\partial\vec{y}}\right|$
or $P_k(\vec{x}) = P_y(g^{-1}(\vec{x}))\left|\frac{\partial\vec{y}}{\partial\vec{x}}\right| = P_y(g(\vec{x}))\,|\det J|$.
Let $J = (\frac{\partial y_i}{\partial x_j})_{i,j}$.

Information Theory

information measures the amount of uncertainty :
(1) Likely events have low information
(2) Less likely events have higher information
(3) Information events have additive information. (toss a coin twice)

Self-information at event $X = x$, $I(x) \triangleq -\log P(x)$, other logarithms (base-2) is called bits or shannons.
Shannon entropy : $H(X) = E_{x\sim p}[I(X)] = -E_{x\sim p}[\log P(x)]$.
It gives a lower bound on the number of bits need on average to encode symbols drawn from P.
If X is continuous, $H(P)$ is differential entropy.
e.g. Discrete uniform distribution maximite discrete entropy within the distributions having the same number of states
$\max_{\{p_i\}} \sum_{i=1}^k -p_i\log p_i = E\log p$, s.t. $\sum_{i=1}^k p_i = 1. \Rightarrow p_i = \frac{1}{k}$.

Examples of Entropy

If the sun rises in the East there is no information content, the sun always rises in the East.

If you toss an unbiased coin then there is information in whether it lands heads or tails up. If the coin is biased towards heads then there is more information if it lands tails.

Surprisal associated with an event is the negative of the logarithm of the probability of the event $-\log_2(p)$.

People use different bases for the logarithm, but it doesn't make much difference, it only makes a scaling difference. But if you use base 2 then the units are the familiar bits. If the event is certain,so that $p = 1$, the information associated with it is zero. The lower the probability of an event the higher the surprise, becoming infinity when the event is impossible.

Examples of Entropy

But why logarithms? The logarithm function occurs naturally in information theory. Consider for example the tossing of four coins. There are 16 possible states for the coins, HHHH, HHHT, ..., TTTT. But only four bits of information are needed to describe the state. HTHH could be represented by 0100.

$$4 = \log_2(16) = -\log_2(1/16).$$

Going back to the biased coin, suppose that the probability of tossing heads is $3/4$ and $1/4$ of tossing tails. If I toss heads then that was almost expected, there's not that much information. Technically it's $-\log_2(0.75) = 0.415$ bits. But if I toss tails then it is $-\log_2(0.25) = 2$ bits.

Examples of Entropy

This leads naturally to looking at the average information, this is our entropy :

$$-\sum p \log_2(p),$$

where the sum is taken over all possible outcomes.

(Note that when there are only two possible outcomes the formula for entropy must be the same when p is replaced by $1 - p$. And this is true here.)

Cross Entropy

Suppose you have a *model* for the probability of discrete events, call this p_k^{M} where the index k just means one of K possibilities. The sum of these probabilities must obviously be one.

And suppose that you have some empirical data for the probabilities of those events, p_k^{E}. With the sum again being one.

The cross entropy is defined as

$$-\sum_k p_k^{\mathrm{E}} \ln(p_k^{\mathrm{M}}).$$

It is a measure of how far apart the two distributions are.

Example of Cross Entropy

Suppose that you have a machine-learning algorithm that is meant to tell you whether a fruit is a passion fruit, orange or guava.

As a test you input the features for an orange. And the algorithm is going to output three numbers, perhaps thanks to the softmax function, which can be interpreted as the probabilities of the fruit in question (the orange) being one of P, O or G. Will it correctly identify it as an orange?

The model probabilities come out of the algorithm as

$$p_{\mathrm{P}}^{\mathrm{M}} = 0.13,\ p_{\mathrm{O}}^{\mathrm{M}} = 0.69,\ \text{and}\ p_{\mathrm{G}}^{\mathrm{M}} = 0.18.$$

Ok, it's done quite well. It thinks the fruit is most likely to be an orange. But it wasn't 100% sure.

Example of Cross Entropy

Empirically we know that

$$p_{\mathrm{P}}^{\mathrm{E}} = 0,\ p_{\mathrm{O}}^{\mathrm{E}} = 1,\ \text{and}\ p_{\mathrm{G}}^{\mathrm{E}} = 0,$$

because it definitely is an orange. The cross entropy is thus

$$-(0 \times \ln(0.13) + 1 \times \ln(0.69) + 0 \times \ln(0.18)) = 0.371.$$

The cross entropy is minimized when the model probabilities are the same as the empirical probabilities.

Example of Cross Entropy

To see this we can use Lagrange multipliers. Write

$$L = -\sum_k p_k^{\mathrm{E}} \ln(p_k^{\mathrm{M}}) - \lambda \left(\sum_k p_k^{\mathrm{M}} - 1 \right).$$

The second term on the right is needed because the sum of the model probabilities is constrained to be one.

Now differentiate with respect to each model probability, and set the results to zero :

$$\frac{\partial L}{\partial p_k^{\mathrm{M}}} = -\frac{p_k^{\mathrm{E}}}{p_k^{\mathrm{M}}} - \lambda = 0.$$

But since the sums of the two probabilities must be one we find that $\lambda = -1$ and $p_k^{\mathrm{M}} = p_k^{\mathrm{E}}$.

Example of Cross Entropy

Because the cross entropy is minimized when the model probabilities are the same as the empirical probabilities we can see that cross entropy is a candidate for a useful cost function when you have a classification problem.

If you take another look at the sections on MLE and on cost functions, and compare with the above on entropy you'll find a great deal of overlap and similarities in the mathematics. The same ideas keep coming back in different guises and with different justifications and uses.

Kullback-Leibler (KL) divergence

Two distributions $P(x), Q(x)$

$H(P, Q) = -E_{x\sim p} \log Q(x)$ cross-entropy

$D_{KL}(P||Q) = E_{x\sim p}[\log \frac{P(X)}{Q(X)}] = -H(P) + H(P, Q)$ Extra information "diatance".

Properties : $D_{KL}(P||Q) = 0$ iff $P = Q$ a.e. $D_{kL}(P||Q) \geq 0$
But $D_{kL}(P||Q) \neq D_{kL}(Q||P)$, $0 \log 0 = \lim_{x\to 0} x \log x = 0$.

Outlines

Linear Algebra

Probability and Information Theory

Statistics and Machine Learning

References

Tasks of Machine Learning

(1) Classification : find $f : \mathbb{R}^n \to \{1, \cdots, k\}$ s.t. $y \approx f(x)$.
(2) Regression : find $f : \mathbb{R}^n \to \mathbb{R}$, s.t. $y \approx f(x)$.
(3) Anomaly Detection : find $P(x)$ for normal samples, predict abnormal samples $\hat{x}$ with small prob $P(\tilde{x}) < \varepsilon$.
(4) Imputation of missing values : predict $P(x)$ for $x = (x_1, \cdots, x_n)$ then insert the value of x_i for a new sample $\vec{x}$ with x_i missing.
(5) Denoising : Given a corrupted example $\tilde{x} \in \mathbb{R}^n$, find a clean example $\tilde{x} \in \mathbb{R}^n$ s.t. $x \approx \tilde{x}$ with some noise, i.e. find $P(x|\tilde{x})$.

Tasks of Machine Learning

(6) Density Estimation or PMF estimation :
find $P_{model} : \mathbb{R}^n \to \mathbb{R}$, $P_{\mathrm{model}}(x)$ for given samples $\vec{x}$. All previous problems, and clustering dimensionality reduction etc, could fall into this category.
This is rather difficult, computationally intractable.
supertrained learning : $P(y|x) = \frac{P(x,y)}{\sum_{y'} P(x,y')}$.
Learning conditional statistics (e.g. expectation). given a measure of diviation (loss function). $L(y, f)$ want to find the best f^* that minimize it i.e. $\min_f E_{x,y\sim P_{data}} L(y, f(x))$.
If $L = \|y - f\|_2^2$, then $f^*(x) = E_{y\sim P_{data}(y|x)}[y]$.
If $L = \|y - f\|_1$, then $f^*(x) =$ conditional median.

Linear Regression

$P_{data}(x,y) = \frac{1}{m}\sum_{i=1}^{m}\delta(x-x^{(i)},y-y^{(i)})$
$f(x) = w^T x,$
$\min_{f=w^T x} E_{x,y\sim P_{data}(x,y)} \|y-f(x)\|_2^2 \approx \min_w \frac{1}{m}\sum_{i=1}^{m}\|y^{(i)}-w^T x^{(i)}\|_2^2$
$P_{data}(x,y) \approx P_{train}(x,y)$
$\nabla_w MSE_{train} = 0 \Leftrightarrow w = (X^{(train)T}X^{(train)})^{-1}X^{(train)T}y^{(train)}$

Goals :

1) Make MSE_{train} small (under fitting if this is not achieved)
2) Make $MSE_{train} - MSE_{test}$ small (over fitting if this is not achieved)

Linear Regression

e.g. $y = b + \sum_{i=1}^{m} w_i x^i$
$m = 9$, overfitting
$m = 1$, underfitting
$m = 3$, optimal

Linear Regression

Performance : $MSE_{test} = \frac{1}{m^{(test)}}\|X^{(test)}w - y^{(test)}\|_2^2$.

If $(X^{(train)}, y^{(train)})$ and $(X^{(test)}, y^{(test)}) \sim P_{data}(x,y)$, then $MSE_{train} = MSE_{test}$ for $\hat{w}$ computed from 1).

However, in general, $MSE_{train} \leq MSE_{test}$, since $\hat{w}$ is computed from 2).

Regularizaition

Insteat of minimizing MSE_{train}, we take into account the model complexity, in the case of linear regression, $\lambda\|w\|_2^2$,
$J(w) = MSE_{train} + \lambda\|w\|_e^2$, $\lambda \to 0$ over fitting.

$\lambda \to +\infty$, underfitting, optimal $\lambda \in (0, +\infty)$.

Statistical Estimators

Point Estimation : estimate Q in $f(x;Q)$ by $\hat{Q}$.

$$\hat{Q} = \arg\min_Q E_{x,y\sim P_{data}(x,y)} L(y, f(x,Q))$$
$$= \arg\min_Q \frac{1}{m}\sum_{i=1}^{m} L(y^{(i)}, f(x^{(i)};Q))$$
$$\Rightarrow \hat{Q}_m = g((x^{(i)},y^{(n)}) \sim (x^{(m)},y^{(m)});Q)$$

Function Estimation : nonparameterized $f(x)$ in some functional space $\mathfrak{H}$, then the least square rule gives $\hat{f}$ as the best approximation of f among $\mathfrak{H}$.–point estimation in $\mathfrak{H}$.
bais$(\hat{Q}_m) = E[\hat{Q}_m] - Q$
$\hat{Q}$ is unbiased if $bais(\hat{Q}_m) = 0$.
asymptotically unbiased if $\lim_{m\to\infty} bais(\hat{Q}_m) = 0$.

Statistical Estimators

e.g. Bernoulli, $\{x^{(1)},\cdots,x^{(m)}\}$, $P(x^{(i)};Q) = Q^{x^{(i)}}(1-Q)^{1-x^{(i)}}$.
Let $\hat{Q}_m = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$, $E[\hat{Q}_m] = \frac{1}{m}\sum_{i=1}^{m} E[x^{(i)}] = 0$, unbiased.

e.g. Gaussian, $\{x^{(1)},\cdots,x^{(m)}\}$,
$P(x^{(i)}) = N(x^{(i)};\mu,\sigma^2) = \frac{1}{\sqrt{2}\sigma}\exp(-\frac{(x^{(i)}-\mu)^2}{2\sigma^2})$
$\hat{\mu}_m = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$, $E[\hat{\mu}_m] = \frac{1}{m}\sum_{i=1}^{m} E[x^{(i)}] = \mu$. unbiased
$\hat{\sigma}_m^2 = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)}-\hat{\mu}_m)^2$ sample variances
$E[\hat{\sigma}_m^2] = \frac{m-1}{m}\sigma^2 \neq \sigma^2$ biased
but $\hat{\sigma}_m^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x^{(i)}-\hat{\mu}_m)^2 = \frac{m-1}{m}\hat{\sigma}_m^2$ is unbiased.

Variance and Standard Error

$\hat{Q}_m$:estimator. $\sqrt{Var(\hat{Q}_m)}$ =standard error $= SE(\hat{Q}_m)$
e.g. $SE(\hat{\mu}_m) = \sqrt{Var[\frac{1}{m}\sum_{i=1}^{m} x^{(i)}]} = \frac{\sigma}{\sqrt{m}}$
$CLT \Rightarrow \hat{\mu}_m \sim N(\mu, SE^2(\hat{\mu}_m))$
Confidence interval : $(\hat{\mu}_m - 1.96SE(\hat{\mu}_m), \hat{\mu}_m + 1.96SE(\hat{\mu}_m))$.
e.g. Bernoulli : $Var(\hat{Q}_m) = \frac{1}{m^2}\sum_{i=1}^{m} Var(x^{(i)}) = \frac{1}{m}Q(1-Q)$,
$SE(\hat{Q}_m) = \frac{\sqrt{Q(1-Q)}}{\sqrt{m}}$ decreasing function of m.
Bias-variance Tradeoff
$MSE = E[(\hat{Q}_m - Q)^2] = Bais(\hat{Q}_m)^2 + Var(\hat{Q}_m)$
$= E[(\hat{Q}_m - E\hat{Q}_m)^2 + 2(\hat{Q}_m - E\hat{Q}_m)(E\hat{Q}_m - Q) + (E\hat{Q}_m - Q)^2]$

Variance and Standard Error

Underfitting : High bias, low variance
Overfitting : Low bias, high variance
Consistency :
$\hat{Q}_m$ is a consistent estimator if $\hat{Q}_m \xrightarrow{P} Q$ $(m \to \infty)$.
i.e. $P(\hat{Q}_m - Q|_{>\varepsilon}) \to 0$ $(m \to \infty)$ for $\forall \varepsilon > 0$.
Consistency$\Rightarrow$ Asymptotic unbiasedness.
A counter-example for the reverse statement :
$x^{(i)} \sim N(x; \mu, \sigma^2)$, $\hat{\mu} = x^{(1)}$, then $E[\hat{\mu}] = E[x^{(1)}] = \mu$.
But $\hat{\mu}$ does not trends to μ as $m \to \infty$.

Maximum Likelihood Estimation (MLE)

Usually used in parametric density estimation.
$P_{data}(x) \approx P_{\text{model}}(x; Q)$, $\{x^{(i)}\}_{i=1}^{m}$ drawed i.i.d from $P_{data}(x)$.
Assume $\exists Q$, as if $x^{(i)} \sim P_{\text{model}}(x; Q)$.
MLE for Q is

$$Q_{ML} = \arg\max_Q P_{\text{model}}(X; Q) = \arg\max_Q \prod_{i=1}^{m} P_{\text{model}}(x^{(i)}; Q)$$

or

$$Q_{ML} = \arg\max_Q \prod_{i=1}^{m} \log P_{\text{model}}(x^{(i)}; Q)$$
$$= \arg\max_Q E_{X \sim \hat{P}_{data}} \log P_{\text{model}}(X; Q)$$

Property : Q_{ML} minimizes KL divergence (dissimilarity) between $\hat{P}_{data}$ and P_{model}
$D_{KL}(\hat{P}_{data} || P_{\text{model}}) = E_{X \sim \hat{P}_{data}}[\log \hat{P}_{data}(X) - \log P_{\text{model}}(X; Q)]$
$\min_Q D_{KL} \Leftrightarrow \min_Q E_{X \sim \hat{P}_{data}}[-\log P_{\text{model}}(X; Q)]$

Examples of MLE

Maximum Likelihood Estimation (MLE) is a common method for estimating parameters in a statistical/probabilistic model.

In words, you simply find the parameter (or parameters) that maximizes the likelihood of observing what actually happened.

Let's see this in a few classical examples.

Examples of MLE

Example : Taxi numbers

You arrive at the train station in a city you've never been to before. You go to the taxi rank so as to get to your final destination. There is one taxi, you take it. While discussing European politics with the taxi driver you notice that the cab's number is 1234. How many taxis are in that city ?

To answer this we need some assumptions. Taxi numbers are positive integers, starting at 1, no gaps and no repeats. We'll need to assume that we are equally likely to get into any cab. And then we introduce the parameter N as the number of taxis.

What is the MLE for N ?

Examples of MLE

Example : Taxi numbers

What is the MLE for N ?

Well, what is the probability of getting into taxi number 1234 when there are N taxis ?

It is $\frac{1}{N}$ for $N \geq 1234$ and zero otherwise.

What value of N maximizes this expression ? Obviously it is $N = 1234$. That is the MLE for the parameter N. It looks a bit disturbing because it seems a coincidence that you happened to get into the cab with the highest number. But then the probability of getting into any cab is equally likely. It is also disturbing that if there are N taxis then the average cab number is $(N+1)/2$, and we somehow feel that should play a role.

Examples of MLE

Example : Coin tossing

Suppose you toss a coin n times and get h heads. What is the probability, p, of tossing a head next time ?

The probability of getting h heads from n tosses is, assuming that the tosses are independent,

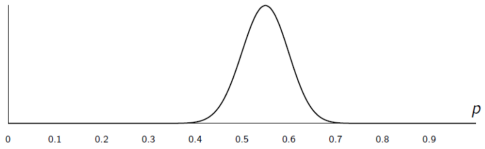
$$\frac{n!}{h!(n-h)!}p^h(1-p)^{n-h} = \binom{n}{h}p^h(1-p)^{n-h}.$$

Applying MLE is the same as maximizing this expression with respect to p.

Examples of MLE

Example : Coin tossing

This likelihood function (without the coefficient in the front that is independent of p) is shown below for $n = 100$ and $h = 55$. There is a very obvious maximum.



Examples of MLE

Example : Coin tossing

Often with MLE when multiplying probabilities, as here, you will take the logarithm of the likelihood and maximize that.

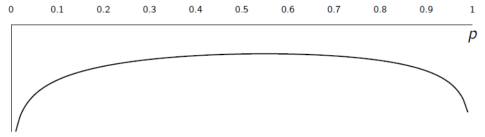
This doesn't change the maximizing value but it does stop you from having to multiply many small numbers, which is going to be problematic with finite precision.

(Look at the scale of the numbers on the vertical axis in the figure.)

Examples of MLE

Example : Coin tossing

Since the first part of this expression is independent of p we maximize $h \ln p + (n-h)\ln(1-p)$ with respect to p. See below.



This just means differentiating with respect to p and setting the derivative equal to zero. This results in $p = \frac{h}{n}$, which seems eminently reasonable.

Conditional Log-likelihood

MLE for Q in $P(Y|X;Q)$

$$Q_{ML} = \arg\max_Q P(Y|X;Q) = \arg\max_Q \prod_{i=1}^{m} P(y^{(i)}|x^{(i)};Q)$$

e.g. linear Regression : $y = w^T x + \varepsilon, \qquad \varepsilon \sim N(0,\sigma^2)$.
Then $P(y|x;w) \sim N(y; w^T x, \sigma^2)$, $\frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{(y-w^T x)^2}{2\sigma^2})$.

Conditional Log-likelihood

$$\begin{aligned}
MLE &\Leftrightarrow \max_w \sum_{i=1}^{m} \log P(y^{(i)}|x^{(i)};w) \\
&= \max_w \sum_{i=1}^{m} (-\log\sigma - \frac{1}{2}\log(2\pi) - \frac{(y^{(i)} - w^T x^{(i)})^2}{2\sigma^2}) \\
&= -m\log\sigma - \frac{m}{2}\log(2\pi) - \min_w \sum_{i=1}^{m} (y^{(i)} - w^T x^{(i)})^2/\sigma^2 \\
&\Leftrightarrow \min_w \frac{1}{m}\sum_{i=1}^{m} (y^{(i)} - w^T x^{(i)})^2 = \min_w MSE_{train}
\end{aligned}$$

Properties of MLE

Under certain conditions (given below). MLE is a consistent estimator of the truth.

(1) P_{data} lies in $\{P_{\text{model}}(;Q);Q\}$; otherwise, no estimator can recover P_{data}.

(2) $\exists Q$, s.t. $P_{data} = P_{\text{model}}(;Q)$; otherwise, MLE can recover P_{data}, but not be able to determine Q.

$E_{X\sim P_{data}}[\hat{Q}_{ML} - Q]^2 \searrow$ Cramer-Rao bowd as $m \to \infty$.

But MLE is not always unbiased, e.g. $\hat{\sigma}^2_{ML} = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \hat{\mu})^2$

Bayesian Statistics

Previously, Q is fixed but unknown. $\hat{Q}$ is a random variable as a function of data $\{x^{(i)}\}_{i=1}^m$ ($x^{(i)}$ is random).
Bayesian : $\{x^{(i)}\}_{i=1}^m$ is observed and non-random. Q is unknown and uncertain (random).
Prior prob distribution $P(Q)$, before observing the data. Given Q, $\{x^{(i)}\}_{i=1}^m$ is generated from $P(x^{(1)}, \cdots, x^{(m)}|Q)$.
Bayes' rule $\Rightarrow P(Q|x^{(1)}, \cdots, x^{(m)}) = \frac{P(x^{(1)}, \cdots, x^{(m)}|Q)P(Q)}{P(x^{(1)}, \cdots, x^{(m)})}$.
$P(Q)$ is usually given e.g. uniform or Gaussian with high entropy, observation of data causes the posterior to loose entropy and concentrate around a few highly likely values of parameters.
Bayesian estimates the distribution of Q instead of point estimate.

$$P \in X^{(m+1)}|x^{(1)}, \cdots, x^{(m)}) = \int P(x^{(m+1)}|Q)P(Q|x^{(1)}, \cdots, x^{(m)})dQ$$

As more observations are given, knowledge about Q becomes different (more).

Bayesian Statistics

	Point Estimate	Bayesian
uncertainty	variance of estimator through random sampling of data	distribution, integral over it
prior info	No	Yes with human knowledge
performance	good as sample size increases	generalize better for limited training data
computational cost	low	high

Bayesian Statistics

e.g. Bayesian Linear Regression $\hat{y} = w^T x$.
Given $(X^{(train)}, y^{(train)})$, $\hat{y}^{(train)} = X^{(train)}w$
$P(y^{(train)}|X^{(train)}, w) = N(y^{(train)}; X^{(train)}w, I)\exp(-\frac{1}{2}(y^{(train)} - X^{(train)}w)^T(y^{(train)} - X^{(train)}w))$
Prior of w : $P(w) = N(w;\mu_0,\Lambda_0)\exp(-\frac{1}{2}(w-\mu_0)^T\Lambda_0^{-1}(w-\mu_0))$
Posterior : $P(w|X^{(train)}, y^{(train)}))P(y^{(*)}|X^{(train)}, w)P(w)$
$\exp(-\frac{1}{2}(y-Xw)^T(y-Xw))\exp(-\frac{1}{2}(w-\mu_0)^T\Lambda_0^{-1}(w-\mu_0))$
$\exp(-\frac{1}{2}(-2y^TXw + w^TX^TXw + w^T\Lambda_0^{-1}w - 2\mu_0^T\Lambda_0^{-1}w))$
$\exp(-\frac{1}{2}(w-\mu_m)^T\Lambda_m^{-1}(w-\mu_m) + \frac{1}{2}\mu_m^T\Lambda_m^{-1}\mu_m)$
$\exp(-\frac{1}{2}(w-\mu_m)^T\Lambda_m^{-1}(w-\mu_m))$
terms without w are normalizing constant.
If $\mu_0 = 0$, $\Lambda_0 = \frac{1}{\alpha}I$, $\mu_m = (X^TX + \alpha I)^{-1}X^Ty$ is the ridge regression estimator of w.
also gives the variance $\Lambda_m = (X^TX + \alpha I)^{-1}$

Maximum A Posterior (MAP) similar to MLE

e.g. μ_m is MAP in the previous example.
MAP choose the point of maximum posterior prob.

$$Q_{MAP} = \arg\max_Q P(Q|X) = \arg\max_Q\{\log P(X|Q) + \log P(Q)\}$$

Bayesian linear regression, $\log -prior\ \|w\|_2^2$ ridge penalties.
Additional information in prior helps to reduce the variance in the MAP, but does not increase bias.
Different regularizations (penalties) corresponds to different log-prior. (but not all, smoe penalty may not be a logarithm of a prob distribution, some others depend on data)
Lsso Penalty $\to$ Laplace distribution Laplace$(x;0,\lambda^{-1})$.

Outlines

Linear Algebra

Probability and Information Theory

Statistics and Machine Learning

References

References

- Numerical Analysis, 9th Edition, by Richard L. Burden, J. Douglas Faires, Brooks/Cole, 2011.
- Machine learning : an applied mathematics introduction, by Wilmott, Paul. Panda Ohana Publishing, 2019.
- Deep learning, by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, MIT Press, 2016.