14

Mathematical background

The main mathematical tools of machine learning are optimization and statistics. At their core are concepts from multivariate calculus and probability. Here, we briefly review some of the concepts from calculus and probability that we will frequently make use of in the book.

Common notation

- Lowercase letters u, v, w, x, y, z, typically denote vectors. We use both $\langle u, v \rangle$ and $u^T v$ to denote the inner product between vectors u and v.
- Capital letters X, Y, Z typically denote random variables.
- The conditional probability $\mathbb{P}[A \mid B]$ of an event A conditional on an event B
- The gradient $\nabla f(x)$ of a function $f \colon \mathbb{R}^d \to \mathbb{R}$ at a point $x \in \mathbb{R}^d$ refers to the vector of partial derivatives of f evaluated at x.
- Identity matrix I
- The first k positive integers $[k] = \{1, 2, \ldots, k\}$.

Multivariable calculus and linear algebra

Positive definite matrices

Positive definite matrices are central to both optimization algorithms and statistics. In this section, we quickly review some of the core properties that we will use throughout the book.

A matrix M is *positive definite* (pd) if it is symmetric $M = M^T$ and $z^T M z > 0$ for all nonzero $z \in \mathbb{R}^d$. We denote this as $M \succ 0$ A matrix M is *positive semidefinite* (psd) if it is symmetric and $z^T M z \geq 0$ for all nonzero z. We denote this as $M \succeq 0$. All pd matrices are psd, but not vice versa.

Some of the main properties of positive semidefinite matrices include.

1. If $M_1 \succeq 0$, and $M_2 \succeq 0$, then $M_1 + M_2 \succeq 0$.
2. $a \in \mathbb{R}$, $a \geq 0$ implies $aM \succeq 0$.

3. For any matrix F, FF^T and F^TF are both psd. Conversely, if M is psd there exists an F such that $M = FF^T$.

Note that (1) and (2) still hold if "psd" is replaced with "pd." That is, the sum of two pd matrices is pd. And multiplying a pd matrix by a positive scalar preserves positive definiteness.

Recall that λ is an eigenvalue of a square matrix M if there exists a nonzero $x \in \mathbb{R}^d$ such that $Mx = \lambda x$. Eigenvalues of psd matrices are all non-negative. Eigenvalues of pd matrices are all positive. This follows by multiplying the equation $Ax = \lambda x$ on the left by x^T.

Gradients, Taylor's Theorem and infinitesimal approximation

Let $\Phi : \mathbb{R}^d \to \mathbb{R}$. Recall from multivariable calculus that the *gradient* of Φ at a point w is the vector of partial derivatives

$$\nabla\Phi(w) = \begin{bmatrix} \frac{\partial\Phi(w)}{\partial x_1} \\ \frac{\partial\Phi(w)}{\partial x_2} \\ \vdots \\ \frac{\partial\Phi(w)}{\partial x_d} \end{bmatrix}.$$

Sometimes we write $\nabla_x\Phi(w)$ to make clear which functional argument we are referring to.

One of the most important theorems in calculus is *Taylor's Theorem*, which allows us to approximate smooth functions by simple polynomials. The following simplified version of Taylor's Theorem is used throughout optimization. This form of Taylor's theorem is sometimes called the multivariable mean-value theorem. We will use this at multiple points to analyze algorithms and understand the local properties of functions.

Theorem 1. ***Taylor's Theorem.***

- *If Φ is continuously differentiable, then, for some $t \in [0,1]$,*

$$\Phi(w) = \Phi(w_0) + \nabla\Phi(tw + (1-t)w_0)^T(w - w_0)\,.$$

- *If Φ is twice continuously differentiable, then*

$$\nabla\Phi(w) = \nabla\Phi(w_0) + \int_0^1 \nabla^2\Phi(tw + (1-t)w_0)(w - w_0)\mathrm{d}t$$

and, for some $t \in [0,1]$

$$\Phi(w) = \Phi(w_0) + \nabla\Phi(w_0)^T(w - w_0) + \frac{1}{2}(w - w_0)^T\nabla^2\Phi(tw + (1-t)w_0)^T(w - w_0)\,.$$

Taylor's theorem can be used to understand the local properties of functions. For example,

$$\Phi(w + \epsilon v) = \Phi(w) + \epsilon \nabla \Phi(w)^T v + \frac{\epsilon^2}{2} v^T \nabla^2 \Phi(w + \delta v)^T v$$

for some $0 \leq \delta \leq \epsilon$. This expression states that

$$\Phi(w + \epsilon v) = \Phi(w) + \epsilon \nabla \Phi(w)^T v + \Theta(\epsilon^2)\,,$$

So to first order, we can approximate Φ by a linear function.

Jacobians and the multivariate chain rule

The matrix of first order partial derivatives of a multivariate mapping $\Phi\colon \mathbb{R}^n \to \mathbb{R}^m$ is called *Jacobian matrix*. We define the Jacobian of Φ with respect to a variable x evaluated at a value w as the $m \times n$ matrix

$$\mathsf{D}_x \Phi(w) = \left[\frac{\partial \Phi_i(w)}{\partial x_j} \right]_{i=1\ldots m, j=1\ldots n}.$$

The i-th row of the Jacobian therefore corresponds to the transpose of the familiar gradient $\nabla_x^T \Phi_i(w)$ of the i-th coordinate of Φ. In particular, when $m = 1$ the Jacobian corresponds to the transpose of the gradient.

The first-order approximation given by Taylor's theorem directly extends to multivariate functions via the Jacobian matrix. So does the *chain rule* from calculus for computing the derivatives of function compositions.

Let $\Phi\colon \mathbb{R}^n \to \mathbb{R}^m$ and $\Psi\colon \mathbb{R}^m \to \mathbb{R}^k$. Then, we have

$$\mathsf{D}_x \Psi \circ \Phi(w) = \mathsf{D}_{\Phi(w)} \Psi(\Phi(w)) \mathsf{D}_x \Phi(w)\,.$$

As we did with the gradient notation, when the variable x is clear from context we may drop it from our notation and write $\mathsf{D}\Phi(w)$

Probability

Contemporary machine learning uses probability as its primary means of quantifying uncertainty. Here we review some of the basics we will make use of in this course. This will also allow us to fix notation.

We note that often times, mathematical rigor gets in the way of explaining concepts. So we will attempt to only introduce mathematical machinery when absolutely necessary.

Probability is a function on sets. Let $\mathcal{X}$ denote the sample set. For every $A \subset \mathcal{X}$, we have

$$0 \leq \mathbb{P}[A] \leq 1\,, \qquad \mathbb{P}[\mathcal{X}] = 1\,, \qquad \mathbb{P}[\emptyset] = 0\,,$$

and

$$\mathbb{P}[A \cup B] + \mathbb{P}[A \cap B] = \mathbb{P}[A] + \mathbb{P}[B]\,.$$

This implies that

$$\mathbb{P}[A \cup B] = \mathbb{P}[A] + \mathbb{P}[B]\,.$$

if and only if $\mathbb{P}[A \cap B] = 0$. We always have the inequality

$$\mathbb{P}[A \cup B] \leq \mathbb{P}[A] + \mathbb{P}[B]\,.$$

By induction, we get the union bound

$$\mathbb{P}\left[\bigcup_i A_i\right] \leq \sum_i \mathbb{P}[A_i]\,.$$

Random variables and vectors

Random variables are a particular way of characterizing outcomes of random processes. We will use capital letters like X, Y, and Z to denote such random variables. The sample space of a random variable will be the set where a variable can take values. Events are simply subsets of possible values. Common examples we will encounter in this book are

- **Probability that a random variable has a particular value**. This will be denoted as $\mathbb{P}[X = x]$. Note here that we use a lower case letter to denote the value that the random variable might take.
- **Probability that a random variable satisfies some inequality**. For example, the probability that X is less than a scalar t will be denoted as $\mathbb{P}[X \leq t]$.

A *random vector* is a random variable whose sample space consists of R^d. We will not use notation to distinguish between vectors and scalars in this text.

Densities

Random vectors are often characterized by *probability densities* rather than by probabilities. The density p of a random variable X is defined by its relation to probabilities of sets:

$$\mathbb{P}[X \in A] = \int_{x \in A} p(x)\mathrm{d}x\,.$$

Expectations

If f is a function on R^d and X is a random vector, then the expectation of f is given by

$$\mathbb{E}[f(X)] = \int f(x)p(x)\mathrm{d}x$$

If A is a set, the *indicator function of the set* is the function

$$I_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{otherwise} \end{cases}$$

Note that the expectation of an indicator function is a probability:

$$\mathbb{E}[I_A(X)] = \int_{x \in A} p(x)\mathrm{d}x = \mathbb{P}[X \in A]\,.$$

This expression links the three concepts of expectation, density, and probability together.

Note that the expectation operator is linear:

$$\mathbb{E}[af(X) + bg(X)] = a\,\mathbb{E}[f(X)] + b\,\mathbb{E}[g(x)]\,.$$

Two other important expectations are the mean and covariance. The *mean* of a random variable is the expected value of the identity function:

$$\mu_X := \mathbb{E}[X] = \int xp(x)\mathrm{d}x\,.$$

The *covariance* of a random variable is the matrix

$$\Sigma_X := \mathbb{E}[(X - \mu_X)(X - \mu_X)^T]\,.$$

Note that covariance matrices are positive semidefinite. To see this, take a nonzero vector z and compute

$$z^T\Sigma_X z := \mathbb{E}[z^T(X - \mu_X)(X - \mu_X)^T z] = \mathbb{E}[((X - \mu_X)^T z)^2]\,.$$

Since the term inside the expectation is nonnegative, the expectation is nonnegative as well.

Important examples of probability distributions

- **Bernoulli random variables.** A Bernoulli random variable X can take two values, 0 and 1. In such a case $\mathbb{P}[X = 1] = 1 - \mathbb{P}[X = 0]$

- **Gaussian random vectors.** Gaussian random vectors are the most ubiquitous real valued random vectors. Their densities are parameterized only by their mean and covariance:

$$p(x) = \frac{1}{\det(2\pi\Sigma)^{1/2}} \exp\left(-\tfrac{1}{2}(x-\mu_X)^T\Sigma^{-1}(x-\mu_X)\right) .$$

Gaussian random variables are often called "normal" random variables. We denote the distribution of a normal random variable with mean μ and covariance Σ as

$$\mathcal{N}(\mu, \Sigma) .$$

The reason Gaussian random variables are ubiquitous is because of the central limit theorem: averages of many independent random variables tend to look like Gaussian random variables.

Conditional probability and Bayes' Rule

Conditional probability is applied quite cavalierly in machine learning. It's actually very delicate and should only be applied when we really know what we're doing.

$$\mathbb{P}[A|B] = \frac{\mathbb{P}[A \cap B]}{\mathbb{P}[B]}$$

A and B are said to be *independent* if $\mathbb{P}[A|B] = \mathbb{P}[A]$. Note that from the definition of conditional probability A and B are independent if and only if

$$\mathbb{P}[A \cap B] = \mathbb{P}[A]\,\mathbb{P}[B] .$$

Bayes' Rule is an immediate corollary of the definition of conditional probability. In some sense, it's just a restatement of the definition.

$$\mathbb{P}[A|B] = \frac{\mathbb{P}[B|A]\,\mathbb{P}[A]}{\mathbb{P}[B]}$$

This is commonly applied when A is one of a set of several alternatives. Suppose A_i are a collection of disjoint sets such that $\cup_i A_i = \mathcal{X}$ then for each i, Bayes' Rule states

$$\mathbb{P}[A_i|B] = \frac{\mathbb{P}[B|A_i]\,\mathbb{P}[A_i]}{\sum_j \mathbb{P}[B|A_j]\,\mathbb{P}[A_j]} .$$

This shows that if we have models of the likelihood of B under each alternative A_i and if we have beliefs about the probability of each A_i, we can compute the probability of observing A_i under the condition that B has occurred.

Conditional densities

Suppose X and Z are random variables whose joint distribution is continuous. If we try to write down the conditional distribution for X given $Z = z$, we find

$$\mathbb{P}[X \in A | Z = z] = \frac{\mathbb{P}[X \in A \cap Z = z]}{\mathbb{P}[Z = z]}$$

Both the numerator and denominator are equal to zero. In order to have a useful formula, we can appeal to densities.

$$\begin{aligned}
\mathbb{P}[x \in A | z \leq Z \leq z + \epsilon] &= \frac{\int_z^{z+\epsilon} \int_{x \in A} p(x, z') \mathrm{d}x \mathrm{d}z'}{\int_z^{z+\epsilon} p(z') \mathrm{d}z'} \\
&\approx \frac{\epsilon \int_{x \in A} p(x, z)}{\epsilon p(z) \mathrm{d}z} \\
&= \int_{x \in A} \frac{p(x, z)}{p(z)} \mathrm{d}x
\end{aligned}$$

Letting ϵ go to zero, this calculation shows that we can use the *conditional density* to compute the conditional probabilities of X when $Z = z$:

$$p(x|z) := \frac{p(x, z)}{p(z)} .$$

Conditional expectation and the law of iterated expectation

Conditional expectation is short hand for computing expected values with respect to conditional probabilities:

$$\mathbb{E}[f(x, z) | Z = z] = \int f(x, z) p(x|z) \mathrm{d}x$$

An important formula is the law of iterated expectation:

$$\mathbb{E}[f(x, z)] = \mathbb{E}[\mathbb{E}[f(x, z) | Z = z]]$$

This formula follows because

$$\begin{aligned}
\mathbb{E}[f(x, z)] &= \int \int f(x, z) p(x, z) \mathrm{d}x \mathrm{d}z \\
&= \int \int f(x, z) p(x|z) p(z) \mathrm{d}x \mathrm{d}z \\
&= \int \left(\int f(x, z) p(x|z) \mathrm{d}x \right) p(z) \mathrm{d}z .
\end{aligned}$$

Estimation

This book devotes much of its attention to probabilistic decision making. A different but related statistical problem is *parameter estimation*. Assuming that data X is generated by a statistical model, we'd like to infer some *nonrandom* property about its distribution. The most canonical examples here would be estimating the mean or variance of the distribution. Note that estimating these parameters has a different flavor than decision theory. In particular, our framework of risk minimization no longer applies.

If we aim to minimize a functional

$$\text{minimize}_f \ \mathbb{E}[loss(\vartheta, f(x))]$$

then the optimal choice is to set $f(x) = \vartheta$. But we don't know this parameter in the first place. So we end up with an algorithm that's not implementable.

Instead, what we do in estimation theory is pose a variety of plausible estimators that might work for a particular parameter and consider the efficacy of these parameters in different settings. In particular, we'd like estimators that take a set of observations $S = (x_1, \ldots, x_n)$ and return a guess for the parameter whose value improves as n increases:

$$\lim_{n\to\infty} \mathop{\mathbb{E}}_{S}[loss(\vartheta, \widehat{\vartheta}(S))] = 0$$

Even though estimators are constructed from data, their design and implementation require a good deal of knowledge about the underlying probability distribution. Because of this, estimation is typically considered to be part of classical statistics and not machine learning. Estimation theory has a variety of powerful tools that are aimed at producing high quality estimators, and is certainly worth learning more about. We need rudimentary elements of estimation to understand popular baselines and algorithms in causal inference and reinforcement learning.

Plug-in Estimators

We will restrict our attention to *plug-in estimators*. Plug-in estimators are functions of the moments of probability distributions. They are plug-in because we replace the true distribution with the empirical distribution. To be precise, suppose there exist vector valued functions g and ψ such that $\vartheta = g(\mathbb{E}[\psi(x)])$. Then, given a dataset, $S = (x_1, \ldots, x_n)$, the associated plug-in estimator of ϑ is

$$\widehat{\vartheta}(S) = g\left(\frac{1}{n}\sum_{i=1}^{n} \psi(x_i)\right)$$

that is, we replace the expectation with the sample average. There are canonical examples of plugin estimators.

1. *The sample mean.* The sample mean is the plug-in estimator where g and ψ are both the identity functions.

2. *The sample covariance.* The sample covariance is

$$\widehat{\Sigma}_x = \sum_{i=1}^{n} x_i x_i^T - \left(\frac{1}{n}\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} x_i\right)^T .$$

   From this formula, we can take

$$\psi(x) = \begin{bmatrix}1\\ x\end{bmatrix}\begin{bmatrix}1\\ x\end{bmatrix}^T \quad \text{and} \quad g\left(\begin{bmatrix}A & B\\ B^T & C\end{bmatrix}\right) = C - BB^T .$$

3. *Least-squares estimator.* Suppose we have three random vectors, y, x, and v and we assume that v and x are zero-mean and uncorrelated and that $y = Ax + v$ for some matrix A. Let's suppose we'd like to estimate A from a set of pairs $S = ((x_1, y_1), \ldots, (x_n, y_n))$. One can check that

$$A = \Sigma_{yx}\Sigma_x^{-1} .$$

   And hence the plug-in estimator would use the sample covariances:

$$\widehat{A} = \left(\sum_{i=1}^{n} y_i x_i^T\right)\left(\sum_{i=1}^{n} x_i x_i^T\right)^{-1}$$

   In this case, we have the formulation

$$\psi(x) = \begin{bmatrix}x\\ y\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix}^T \quad \text{and} \quad g\left(\begin{bmatrix}A & B\\ B^T & C\end{bmatrix}\right) = BA^{-1} .$$

Convergence rates

In our study of generalization, we reasoned that the empirical risk should be close to the true risk because sample averages should be close to population values. A similar reasoning holds true for plug-in estimators: smooth functions of sample averages should be close to their population counterparts.

We covered the case of the sample mean in our discussion of generalization. To recall, suppose x is a Bernoulli random variable with mean p.

Let $x_1, \ldots, x_n$ be independent and identically distributed as x. Then Hoeffding's inequality states that

$$\mathbb{P}\left[\left|\frac{1}{n}\sum_{i=1}^{n} x_i - p\right| > \epsilon\right] \leq 2\exp(-2n\epsilon^2)\,.$$

Or, in other words, with probability $1-\delta$,

$$\left|\frac{1}{n}\sum_{i=1}^{n} x_i - p\right| \leq \sqrt{\frac{\log(2/\delta)}{2n}}\,.$$

Let's consider a simple least-squares estimator. Suppose we know that $y = w^T x + v$ where w and x are a vectors, w is deterministic, and x and v are uncorrelated. Consider the least-squares estimator $\widehat{w}_S$ from n data points.. The estimation error in w is the vector $e_S = \widehat{w}_S - w$. The expectation of e_S is zero and the expected norm of the error is given by

$$\mathbb{E}\left[\|e_S\|^2\right] = \text{Trace}\left(\left(\sum_{i=1}^{n} x_i x_i^T\right)^{-1}\right)\,.$$

This error is small if the sample covariance has large eigenvalues. Indeed, if λ_S denotes the minimum eigenvalue of the sample covariance of x, then

$$\mathbb{E}\left[\|e_S\|^2\right] \leq \frac{d}{n\lambda_S}\,.$$

This expression suggests that the distribution of x must have density that covers all directions somewhat equally in order for the least-squares estimator to have good performance. On top of this, we see that the squared error decreases roughly as d/n. Hence, we need far more measurements than dimensions to find a good estimate of w. This is in contrast to what we studied in classification. Most of the generalization bounds for classification we derived were *dimension free* and only depended on properties like the margin of the data. In contrast, in parameter estimation, we tend to get results that scale as number of parameters over number of data points. This rough rule of thumb that the error scales as the ratio of number of parameters to number of data points tends to be a good guiding principle when attempting to understand convergence rates of estimators.

Bibliography