CSE6243: Advanced Machine Learning **Fall 2023**

Lecture 5: Basic Sampling Methods

Lecturer: Bo Dai *Scribes: Vedaant Shah, Feng Zhao*

Note: *LaTeX template courtesy of UC Berkeley EECS Department.*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

5.1 Recap

Up to now, we have been focused on optimization as a building block of machine learning. Specifically if we wish to estimate the parameters θ of our model f for a *supervised learning* setting, we can formulate this as an optimization problem of the form:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} l(y_i, f_\theta(x_i)) + \lambda \Omega(f_\theta)$$

In the above, (x_i, y_i) $(i = 1, 2, ...n)$ represents our training data, l is our loss function, Ω is our regularization function, and λ is a parameter weighting the regularization term. Similarly, we can also write an *unsupervised learning* problem in a similar optimization view:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} l(f_\theta(x_i)) + \lambda \Omega(f_\theta)$$

Notice that the only difference is the absence of y_i since in unsupervised learning, we are not given any outputs for our data. Today, we will examine Bayesian inference, another building block of machine learning, as well as sampling, a crucial component of Bayesian inference.

5.2 New Content

The optimization problems for estimating θ in the above section correspond to the frequentist view of statistics as we wish to find only one θ to use with our model.

Instead, the Bayesian view of statistics treats θ as a random variable with its own probability distribution that we wish to use with our model.

5.2.1 Bayesian Inference

In this Bayesian view, we start with a prior probability distribution $p(\theta)$ which represents our beliefs before seeing any data for which values of θ are more likely.

Afterwards, we are given 1 training sample (x, y), and our goal is now to find $p(\theta|x, y)$, thus telling us which values of θ are more likely after accounting for the given training point. This can be done using *Bayes' Rule* as follows:

$$p(\theta|x,y) = \frac{p(\theta)p(y|x,\theta)}{p(y|x)}$$

$p(y|x)$ is often called the normalization or evidence term and is calculated by integrating over all values of θ:

$$p(y|x) = \int p(y|x)p(\theta)d\theta$$

$p(y|x,\theta)$ is known as the likelihood term, and its expression depends on the choice of our model. For example, when performing linear regression it is common to use the following likelihood expression:

$$p(y|x,\theta) \propto \exp{(-\frac{||y-\theta^T x||_2^2}{2\sigma^2})}$$

If given multiple samples $\mathcal{D} = (x_i, y_i)$ for $i = 1, ...n$, we assume the examples are independent and identically distributed (iid). With this in mind, let $\{x\}$ denote the collection of all input samples x_i and $\{y\}$ denote the set of all output samples y_i. Then we have:

Given $\mathcal{D} = \{x_i, y_i\}_{i=1}^N$,

$$P(\theta|\mathcal{D}) = \frac{\prod_{i=1}^n P(y_i|x_i,\theta)P(\theta)}{P(\{y_i\}_{i=1}^n|\{x_i\}_{i=1}^n)} \tag{5.1}$$

We can now use the same formulations as the 1 sample case, but we replace $p(y|x,\theta)$ with $p(\{y\}|\{x\},\theta) = \prod_{i=1}^n p(y_i|x_i,\theta)$ and $p(y|x)$ with $p(\{y\}|\{x\}) = \int \prod_{i=1}^n p(y_i|x_i,\theta)p(\theta)d\theta$.

Thus, we now have a posterior distribution, denoted by $p(\theta|\mathcal{D}) = p(\theta|\{x\},\{y\})$, representing how likely each value of θ when accounting for our data.

5.2.2 Making Predictions

Approach I: With a posterior distribution of θ, the next step is to use the distribution to predict the output y_{test} for a new input point x_{test}. There are multiple ways to achieve this, such as plugging in the mean value of $p(\theta|\mathcal{D})$ into our model or randomly sampling 1 value from $p(\theta|\mathcal{D})$ and using the sampled value with our model (known as the Gibbs Predictor).

Approach II: However, each of these only use one θ value rather than the entire posterior distribution. Instead, another approach could be to take the expectation over all possible values of y and θ using our entire posterior distribution and likelihood, thus giving the below formula. For simplicity, we have used x in place of x_{test}:

$$y_{\text{test}} = E_{p(\theta|\mathcal{D})p(y|x,\theta)}[y|x] = E_{p(\theta|\mathcal{D})}[E_{p(y|x,\theta)}[y]] = \iint yp(y|x,\theta)p(\theta|\mathcal{D})dyd\theta$$

However, note that the double integral above could be extremely difficult to compute in many cases, making this form of prediction un-feasible to do directly. Instead, we could estimate this value by sampling, thus showing why sampling is such an important concept within Bayesian inference.

5.2.3 Sampling

Instead of working directly with the expectation above, let us start by estimating the more general problem below using sampling, where f some arbitrary function and p is some arbitrary probability distribution:

$$E_{p(x)}[f(x)]$$

If we had the ability to sample from $p(x)$, then one way to approximate this expectation is by taking the mean of f across all of the sampled points. Mathematically, this approximation is

$$\frac{1}{n}\sum_{i=1}^{n} f(x_i), \quad \{x_i\}_{i=1}^{n} \sim p(x)$$

where the notation $\{x_i\}_{i=1}^{n} \sim p$ denotes each x_i for $i = 1, ...n$ is sampled from $p(x)$.

This approximation is known as the Monte-Carlo Approximation, and there are 3 properties we can conclude:

Property 1. The estimation is unbiased. This is mathematically represented as:

$$E_{\{x_i\}_{i=1}^{n} \sim P(x)}\left[\frac{1}{n}\sum_{i=1}^{n} f(x_i)\right] = E_{p(x)}[f(x)]$$

In other words, for any n, if we have k trials where in each trial we produce our *Monte Carlo Approximation*, then it follows that the mean of the approximations from all trials is expected to approach the $E_{P(x)}[f(x)]$ as k approaches ∞. Below we have an example of a few trials to illustrate this concept:

- Trial 1: $\{X_i^1\}_{i=1}^{n}$, $\left[\frac{1}{n}\sum_{i=1}^{n} f(x_i^1)\right]$

- Trial 2: $\{X_i^2\}_{i=1}^{n}$, $\left[\frac{1}{n}\sum_{i=1}^{n} f(x_i^2)\right]$
- ...

- Trial k: $\{X_i^k\}_{i=1}^{n}$, $\left[\frac{1}{n}\sum_{i=1}^{n} f(x_i^k)\right]$

Proof: using linearity of expectation, the proof of the above property is given by:

$$E_{\{x_i\}_{i=1}^{n} \sim p(x)}\left[\frac{1}{n}\sum_{i=1}^{n} f(x_i)\right] = \frac{1}{n}\sum_{i=1}^{n} E_{x_i \sim p(x)}[f(x_i)] = \frac{1}{n} * n * E_{x \sim p(x)}[f(x)] = E_{x \sim p(x)}[f(x)]$$

This represents the unbiasedness of the estimator.

Property 2. Furthermore, as n approaches infinity, the sample mean converges almost surely to the expectation under the distribution $P(x)$. This can be represented as:

$$\lim_{n \to \infty}\left(\frac{1}{n}\sum_{i=1}^{n} f(x_i)\right) \overset{\text{a.s.}}{=} E_{P(x)}[f(x)]$$

The "a.s" denotes almost surely, and it formally means:

$$P\left(\lim_{n \to \infty}\left(\frac{1}{n}\sum_{i=1}^{n} f(x_i)\right) = E_{P(x)}[f(x)]\right) = 1$$

We omit the proof of this property, but it follows from the Law of Large Numbers.

Property 3.

$$\text{Var}[\frac{1}{n}\sum_{i=1}^{n} f(x_i))] = \frac{1}{n^2}\sum_{i=1}^{n} \text{Var}[f(x_i)]$$

Proof: this follows directly from the properties of variance. Note that as $n \to \infty$, the right side approaches 0.

With these properties in mind, the next step is to discuss various methods to actually sample from $p(x)$. We outline two such sampling methods in the following sections to achieve this. For both methods, we are assuming that one can already sample from the uniform distribution between 0 and 1, $U(0, 1)$.

5.2.3.1 Inverse Transformation Sampling

Inverse Transformation Sampling is a technique used to generate random samples from a given probability distribution. The method always assumes a standard uniform distribution as a starting point.

Let us denote the target probability distribution by $p(x)$ and its cumulative distribution function (CDF) by $F(z)$. The CDF is obtained by integrating the probability density function $p(x)$. Note that the CDF is a non-decreasing function whose range will always be $[0, 1]$:

$$F(z) = \int_{-\infty}^{z} p(x)\, dx = P(X \leq z)$$

This is the graph of it:

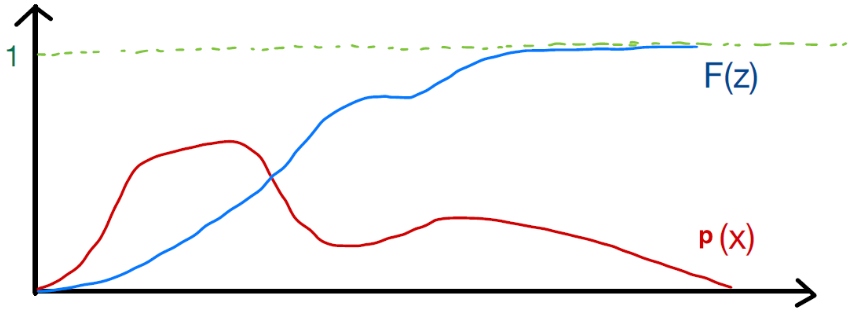

Figure 5.1: Graph illustrating the CDF and PDF

The steps in Inverse Transformation Sampling can be summarized as follows:

1. Generate a random variable u from a uniform distribution in the range [0,1]: $u \sim U[0, 1]$

2. Find the inverse of the cumulative distribution function, denoted as $F^{-1}(u)$, and set it equal to our sample output x:

$$x = F^{-1}(u)$$

To see why this works, note that for any u sampled from $U(0, 1)$, we have:

$$P(F^{-1}(u) \leq z) = P(u \leq F(z)) = F(z)$$

meaning that the CDF of our procedure is exactly F, so we are appropriately sampling from p as desired.

Example 1:

Given a probability density function defined as:

$$P(x) = \lambda e^{-\lambda x}$$

The cumulative distribution function $F(z)$ is then calculated as:

$$F(z) = \int_{0}^{z} e^{-\lambda x} d(-\lambda x) = -\left. e^{-\lambda x}\right|_{0}^{z} = 1 - e^{-\lambda z}$$

From this, we find that:

$$u = F(z) \Rightarrow u = 1 - e^{-\lambda z}$$

Solving for z, we get:

$$z = -\frac{1}{\lambda}\ln(1-u)$$

5.2.3.2 Acceptance-Rejection Sampling

Target/Setting:

We have some proposal distribution $q(x)$, and we wish to sample $x \sim p(x)$ The process can be described in the following steps:

1. Find a M such that $M \cdot q(x) \geq p(x)$ for all x (illustrated as the downward arrow at x).
2. Sample x from $q(x)$: $x \sim q(x)$
3. Generate a random variable u from a uniform distribution in the range [0,1]: $u \sim U[0,1]$. If $u \leq \frac{P(x)}{M \cdot q(x)}$, accept x; otherwise, reject x.

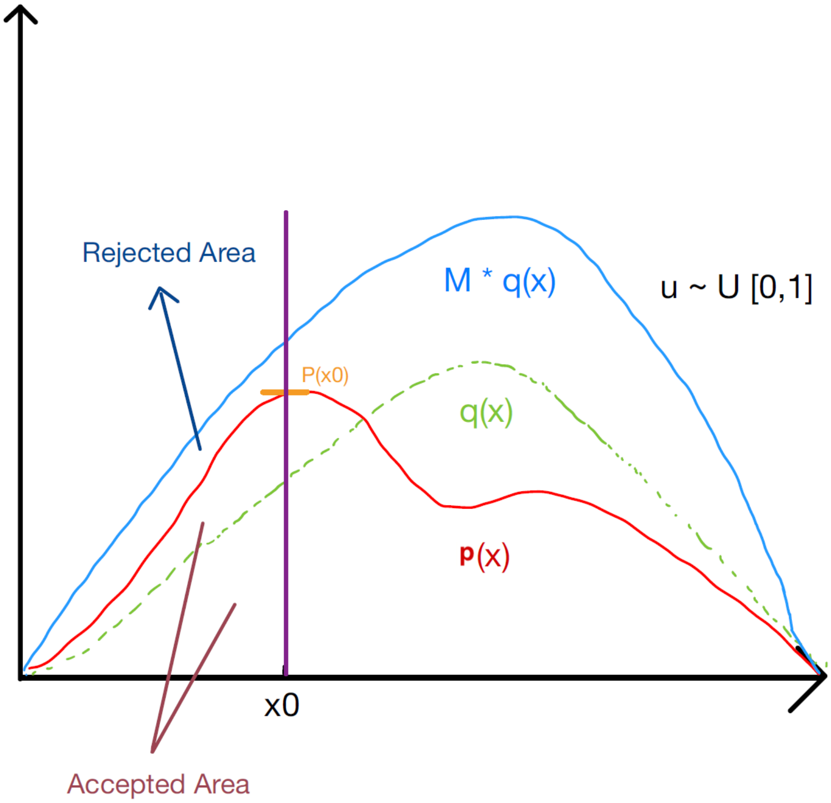

Figure 5.2: Graph of sampling

Let's say we sample x_0 from q. In the graph, there are two regions represented corresponding to acceptance and rejectance: from the x-axis to $P(x_0)$ and from $p(x_0)$ to $M \cdot q(x_0)$. Specifically:

- For the region between $p(x_0)$ and $M \cdot q(x_0)$, the sampled value is rejected.
- For the area below $p(x_0)$, the sampled value is accepted.

Note that any proposal distribution $q(x)$ will work, and we can pick M to be anything that satisfies the required condition. However, we clearly want $M \cdot q(x)$ to be as close to $p(x)$ as possible for all x, so that there is a greater chance we accept each sample, and our algorithm terminates quicker. For poor choices of M and q, our algorithm will work correctly, but not be practical due to it rejecting many samples until finding one to accept.