Probability and Statistics, From Classical to Bayesian

Week 8

18th June - 24th June 2021

1 Introduction

For this final week of the project, we plan to introduce you to sampling and then applications of Bayesian in the field of Machine Learning, through applying it for the problem of Linear Regression. This will also bring us to an end of this long project.

2 Generating Distributions on computers

2.1 Inverse transformation on Uniform variable

First we need to know how to generate Uniform random numbers. This is the most basic problem. In this respect we use group theory results and machine powers. Most programming languages have this implemented(Even C has it implemented).

The most popular method is the inverse transformation. The following result can be used:
If U is a random variable with the uniform distribution, then $F^{-1}(X)$ follows the distribution F(X). Therefore $X = F^1(U)$ is the required variable. Note: For this to work my function F should be invertible. (Btw you can check, this is simple change of variable).

All the discrete distributions can be generated using inverse transformation method. Suppose P(X = a_i) = p_i , for i = 1, 2, ... Without loss of generality we can assume $a_1 < a_2 <$ .... Draw uniform P random number say u, if $\Sigma_{i=1}^{k-1} p_i < u < \Sigma_{i=1}^{k} p_i$, then X takes the value a_k .

Many continuous random variables can be generated using inverse transformation method, for example exponential, Weibull, generalized exponential distributions etc. On the other hand several well known distribution cannot be obtained using inverse transformation method. For example: normal, gamma etc.

2.2 Acceptance Rejection method

If a continuous distribution cannot be generated using inverse transformation method, one of the most useful method is the *Acceptance Rejection method.* The idea is as follows. If we want to generate from $f(x)$, try to find $g(x)$, from which generation is simple so that it satisfies the following $f(x) \leq cg(x)$.

Acceptance Rejection method : Algorithm

1. Generate Y from $g(x)$.
2. Generate a uniform random variable U.
3. If $U \leq f(Y)/cg(Y)$, set X = Y(i.e. positively sampled) , otherwise return to 1.

Example 1: Suppose we want to generate from

$$f(x) = 20x(1-x)^3; 0 < x < 1.$$

Take $g(x) = 1, 0 < x < 1, c = 135/64$.
Example 2: Suppose we want to generate from $f(x) = \frac{2}{\sqrt(\pi)} x^{1/2} e^x \quad 0 < x < \infty$ Take $g(x) = \frac{2}{3} e^{2x/3} \quad 0 < x < \infty$. and $c = \frac{3^{3/2}}{(2e)^{1/2}}$.
This allows us to generate many standard distributions, one of the most notable being the gamma distribution ($Gamma(\alpha, 1)$ for $0 < \alpha < 1$). Now if X $Gamma(\alpha, 1)$, then X/λ $Gamma(\alpha, \lambda)$. Also note that if $X_1, X_2 ... X_k$ $exp(1)$, then $_i$ $Gamma(k, 1)$, let X $Gamma(\alpha, 1)$, then X+$_i$ $Gamma(\alpha + k, 1)$. These are some standard change of variable results, they were written in the change of variable chart I had sent, as well. Combining these results we can generate Gamma distributions for any parameters.

2.3 Log Concave functions

If the log of the distribution is a concave function then it can be sampled using acceptance-rejection principle. Even better, there always exists a gamma distribution s.t. sampling form the gamma distribution is the same as sampling from the actual distribution. We simply equate the first two moments of the required distribution to that of a gamma distribution to get the value of the parameters for gamma.
Condition: $\frac{d^2}{dx^2} ln(F(x)) < 0$. Where F is the distribution of interest.

2.4 Importance Sampling

In Bayesian analysis often we need to compute the posterior mean as follows:

$$\theta = E(h(X)) = \int h(x) f(x) dx.$$

Here $f(x)$ is the PDF of X, and x can be a very high dimensional. In Bayesian analysis, $f(x)$ is the posterior density function. Calculating such integrals may not be computationally feasible. Alternatively, one can use Monte Carlo Simulations to to approximate the value of as follows:

$$\hat{\theta} = \frac{1}{N} \sum_{i=1}^{N} h(X_i)$$

here $X_1, \ldots, X_N$ is a random sample of size N from f(X).
But, often it is observed that it is not very easy to generate samples from $f(x)$, so what we do is, choose some part of f(x), from which we can actually generate samples easily (mostly known distributions, Gamma etc.) and generate samples.

$$\theta = \int h(x)f(x)dx = \int \frac{h(x)f(x)}{g(x)} g(x)dx$$

and,

$$\hat{\theta} = \frac{1}{N} \sum_{i=1}^{N} \frac{h(x_i)f(x_i)}{g(x_i)}$$

2.5 Gibbs Sampling

This is a rather different technique and certainly pretty interesting, I would suggest to follow this **video** to get the basic idea of how it is implemented. Mostly the first few samples that we generate are thrown away as our initial guess might be wrong. This technique can be used for any number of parameter greater than 2. For higher dimensions it is common to sample from each distribution at least once before heading on to use the newly sampled value in the expressions.

2.6 Bayesian in Linear Regression

Check out **this** rather descriptive article on applying Bayesian Approach on Linear Regression. The article mentions the usage of MCMC algorithms to optimize our problem. Gibbs sampling is one such method which is commonly used to do so. The conditional probabilities of each parameter is a simple normal distribution. Sample from each of them, throw away the initial samples and then simply apply WLLN on this sample to get an estimate for the weights. (Mean will converge to the expected value, which will be our estimate). This method is used when the number of parameters are large and calculating inverse of the matrix might be computationally a huge load.