

# 1 L1

## 1.1 Monte Carlo

**Definition** (Monte Carlo Method). Broad class of Algos that repeatedly sample random inputs to obtain numerical results. Focus on three classes: optimization numerical integration and generating draws from a probability distribution.

**Definition** (Monte Carlo Integration). Know how to sample from a distribution and evaluate a function at those samples, we can estimate the integral of the function w.r.t the distribution. **Approach:**

- Draw n samples  $x_1, x_2, \dots, x_n$  from distribution p(x)
- Evaluate the function at each sample:  $f(x_1), f(x_2), \dots, f(x_n)$
- Average the function values to obtain the estimate:

$$\hat{I} = \frac{1}{n} \sum_{i=1}^n f(x_i)$$

**Definition** (Numerical Integration). Integral with bounds, eg:  $\int_a^b f(x)dx$  we partition the interval [a,b] into n subintervals of equal width  $\Delta x = \frac{b-a}{n}$

Then compute the Riemann sum:

$$\hat{I} = \sum_{i=1}^n f(x_i^*) \Delta x$$

where  $x_i^*$  is a sample point in the i-th subinterval.

**Remark** (Uniform Distribution). We can simulate a Uniform distribution  $X = \frac{r_i \text{mod } M}{M} \sim U(0, 1)$  where  $r_i$  is a random integer and M is the maximum possible value of  $r_i$ . We can assume we always have access to uniform distribution  $U(0, 1)$ .

**Example** (Exponential distribution (Inverse CDF)). To simulate exponential dist. Take the density function:  $f(x) = \lambda e^{-\lambda x}$  for  $x \geq 0$

The CDF is given by:

$$F(x) = 1 - e^{-\lambda x}$$

To generate samples from this distribution using inverse transform sampling, we set  $F(x) = u$  where u is a uniform random variable in (0,1):

$$u = 1 - e^{-\lambda x}$$

Solving for x gives:

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

Since 1-u is also uniformly distributed in (0,1), we can simplify this to:

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

Thus, to generate a sample from an exponential distribution with rate parameter  $\lambda$ , we can generate a uniform random variable u in (0,1) and compute:

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

**Definition** (Sim from known Dist. (inverse)).  $X \sim F(x) \implies F(X) \sim U(0, 1)$

$U \sim (0, 1) \implies X = F^{-1}(U) \sim F(x)$  We can simulate from any distribution if we know its CDF and can compute its inverse CDF.

*Proof.*

$$\begin{aligned} P(X \leq t) &= P(F^{-1}(U) \leq t) \\ &= P(U \leq F(t)) \\ &= F(t) \end{aligned}$$

□

**Example** (Simulation of Normal Distribution (CLT)). Simulate  $u_1^{12}$  where  $u_i \sim U(0, 1)$

Then  $z = \sum_{i=1}^{12} u_i - 6 \sim N(0, 1)$  approximately by CLT.

This is due to the central limit theorem. This converges very fast.

**Example** (Simulation of Normal Distribution (Box-Muller)). Let  $u_1, u_2 \sim U(0, 1)$  independent. Then:

$$\begin{aligned}\theta &= 2\pi u_1 \sim U(0, 2\pi) \\ e &= -2 \ln(u_2) \sim \chi^2_2 \\ r &= \sqrt{e} \\ x &= r \cos(\theta) \\ y &= r \sin(\theta)\end{aligned}$$

Both  $x$  and  $y$  are independent  $N(0, 1)$  random variables.

This is due to the Box-Muller transform.

*Proof.*

$$\begin{aligned}f(x, y) &= \frac{1}{2\pi} e^{-\frac{x^2+y^2}{2}} \\ \text{Box-Muller transform} &= \begin{cases} r = \sqrt{x^2 + y^2} \\ \cos(\theta) = \frac{x}{r} \end{cases} \\ f(r, \theta) &= \frac{1}{2\pi} e^{-\frac{r^2}{2}} r\end{aligned}$$

□

**Example** (Bernoulli distribution). Simulate  $u \sim U(0, 1)$

set  $x = 1_{u < p}$  then  $x \sim Bern(p)$  This is due to the fact  $X = 0$  or  $1$  with probabilities  $1 - p$  and  $p$  respectively.

**Example** (Binomial and Multinomial). Simulate  $n$  independent Bernoulli trials with success probability  $p$

Let The sum of these trials follows a Binomial distribution  $Bin(n, p)$

For Multinomial distribution with parameters  $n$  and probabilities  $p_1, p_2, \dots, p_k$

**Case 1:N=1**

Simulate  $u \sim U(0, 1)$

$$\text{Set: } (x_1, x_2, \dots, x_k) = \begin{cases} (1, 0, \dots, 0) & \text{if } u < p_1 \\ (0, 1, 0, \dots, 0) & \text{if } p_1 \leq u < p_1 + p_2 \\ \dots \\ (0, 0, \dots, 1) & \text{if } \sum_{i=1}^{k-1} p_i \leq u < 1 \end{cases}$$

**Case 2:N>1**

Simulate  $u_1, u_2, \dots, u_n \sim U(0, 1)$  independent

Set:  $(x_{1j}, x_{2j}, \dots, x_{kj})$  for each  $u_j$  as in case 1. Then sum over j:

$$(x_1, x_2, \dots, x_k) = \left( \sum_{j=1}^n x_{1j}, \sum_{j=1}^n x_{2j}, \dots, \sum_{j=1}^n x_{kj} \right)$$

This follows the Multinomial distribution with parameters n and  $p_1, p_2, \dots, p_k$ .

**Example** (Poisson Distribution (Inter-arrival times)). Set  $k = 0, s = 0$

While  $s < 1$  do: Simulate  $u \sim U(0, 1)$

$$t = \frac{-1}{\lambda} \log(u)$$

$$s = s + t$$

$$k = k + 1$$

Return  $k - 1 \sim Poisson(\lambda)$  This is due to the fact that the inter-arrival times in a Poisson process are exponentially distributed.

**Example** (Poisson Distribution (CDF inversion)). set  $f = e^{-\lambda}, F = f, x = 0$

Simulate  $u \sim U(0, 1)$

While  $F \leq u$  do:  $x = x + 1$

$$f = \frac{\lambda^x}{x!} e^{-\lambda}$$

$$F = F + f$$

Return  $x \sim Poisson(\lambda)$  This is due to the fact that we are inverting the CDF of the Poisson distribution. This is better as you simulate only one uniform random variable.

**Example** (Simulate from noncentral  $\chi^2$  distribution). simulate  $k$  independent  $N(0, 1)$  random variables  $Z_1, Z_2, \dots, Z_k$

$$\text{set } x = (z_1 + \lambda)^2 + \sum_{i=2}^k z_i^2$$

Return  $x \sim \chi^2_k(\lambda)$  This is due to the definition of noncentral chi-squared distribution as the sum of squares of normal random variables with non-zero means.

*Proof.*

$$\begin{aligned}
X_i &\sim N(\mu_i, 1) \\
\implies X_i &= Z_i + \mu_i \text{ where } Z_i \sim N(0, 1) \\
\sum_{i=1}^k X_i^2 &= \sum_{i=1}^k (Z_i + \mu_i)^2 \\
&= \sum_{i=1}^k Z_i^2 + 2 \sum_{i=1}^k Z_i \mu_i + \sum_{i=1}^k \mu_i^2
\end{aligned}$$

The first term follows a central chi-squared distribution with  $k$  degrees of freedom. The second term is a linear combination of normal random variables, which is also normally distributed. The third term is a constant. Therefore, the sum follows a noncentral chi-squared distribution with  $k$  degrees of freedom and noncentrality parameter  $\lambda = \sum_{i=1}^k \mu_i^2$ .  $\square$

**Example** (Simulate Double Exponential Distribution). Simulate  $u_1, u_2 \sim U(0, 1)$  independent

Set  $z = -\theta \log(u_1)$

If  $u_2 \leq 0.5$  then  $x = -z$  else  $x = z$

Return  $x \sim DE(\theta)$

## 2 L2

**Definition** (Rejection Sampling). We need to simulate  $X \sim f(x)$  where  $f$  is known but difficult to sample from directly. We choose a proposal distribution  $g(x)$  that is easy to sample from and satisfies  $f(x) \leq M g(x)$  for some constant  $M \geq 1$ .

**Algorithm:** Simulate  $u \sim U(0, 1)$  and  $z \sim g(z)$  independent

If  $u \leq \frac{f(z)}{M g(z)}$  then accept  $z$  as a sample from  $f(x)$  else reject  $z$  and repeat the process.

The intuition behind this method is that we are using the proposal distribution  $g(x)$  to generate candidate samples and then accepting or rejecting them based on how well they match the target distribution  $f(x)$ . The constant  $M$  ensures that the acceptance probability is always less than or equal to 1.

For an efficient rejection sampling, we want to choose a proposal distribution  $g(x)$  that closely resembles the target distribution  $f(x)$  and a constant  $M$  that is as small as possible. This will increase the acceptance rate and reduce the number of rejected samples.

**Definition** (Importance Sampling). To sample  $X \sim f(x)$  which is known but difficult. IE we know  $f(x) = \frac{h(x)}{\int h(x) dx}$  where  $h$  is known but the integral is difficult. We choose a proposal distribution  $g(x)$  that is easy to sample from and satisfies  $g(x) > 0$  whenever  $h(x) > 0$ .

**Algorithm** Simulate  $x_1, x_2, \dots, x_n \sim g(x)$  independent

Sample a value of  $X$  from the set  $x_1, x_2, \dots, x_n$  with probability proportional to  $\frac{f(x_i)}{g(x_i)} = \frac{h(x_i)}{\int h(x) dx}$

The intuition behind this method is that we are using the proposal distribution  $g(x)$  to generate samples and then weighting them based on how well they match the target distribution  $f(x)$ . This allows us to obtain samples from the target distribution without having to compute the normalization constant.

We can apply this for an expectation:

$$\begin{aligned}
E_X[a(X)] &= \int a(x) f(x) dx \\
&= \int a(x) \frac{f(x)}{\int h(x) dx} g(x) dx \\
&= E_Z \left[ a(Z) \frac{f(Z)}{g(Z)} \right] \text{ where } Z \sim g(x) \\
&\approx \frac{1}{n} \sum_{i=1}^n a(z_i) \frac{f(z_i)}{g(z_i)} \text{ where } z_i \sim g(x) \text{ independent}
\end{aligned}$$

In summary, we can approximate the expectation of a function with respect to a difficult-to-sample distribution by using importance sampling with a proposal distribution that is easy to sample from.

**Definition** (Gibbs Sampling). Simulate from a joint distribution  $f(x_1, x_2, \dots, x_n)$  where direct sampling is difficult. We iteratively sample from the conditional distributions of each variable given the current values of the other variables.

**Algorithm:** Set initial values  $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$

For  $t = 1, 2, \dots, T$  do:

- Sample  $x_1^{(t)} \sim f(x_1 | x_2^{(t-1)}, x_3^{(t-1)}, \dots, x_n^{(t-1)})$
- Sample  $x_2^{(t)} \sim f(x_2 | x_1^{(t)}, x_3^{(t-1)}, \dots, x_n^{(t-1)})$

- ...

- Sample  $x_n^{(t)} \sim f(x_n | x_1^{(t)}, x_2^{(t)}, \dots, x_{n-1}^{(t)})$

After a sufficient number of iterations, the samples  $(x_1^{(t)}, x_2^{(t)}, \dots, x_n^{(t)})$  will approximate the joint distribution  $f(x_1, x_2, \dots, x_n)$ . The intuition behind this method is that we are breaking down the complex joint distribution into simpler conditional distributions, which are easier to sample from. By iteratively updating each variable based on the current values of the others, we can explore the entire joint distribution over time.

**Example** (Gibbs Sampler).  $f(x, y) = \frac{2x+3y+2}{28}$  for  $0 \leq x \leq 2$  and  $0 \leq y \leq 2$

$$f(x) = \int_0^2 f(x, y) dy = \frac{4x+10}{28} \text{ for } 0 \leq x \leq 2$$

$$f(y) = \int_0^2 f(x, y) dx = \frac{6y+8}{28} \text{ for } 0 \leq y \leq 2$$

$$f(x|y) = \frac{f(x,y)}{f(y)} = \frac{2x+3y+2}{6y+8} \text{ for } 0 \leq x \leq 2$$

$$f(y|x) = \frac{f(x,y)}{f(x)} = \frac{2x+3y+2}{4x+10} \text{ for } 0 \leq y \leq 2$$

**Algorithm:** Set initial values  $x^{(0)} = -5, y^{(0)} = -5$

For  $t = 1, 2, \dots, T$  do:

- Sample  $x^{(t)} \sim f(x|y^{(t-1)})$
- Sample  $y^{(t)} \sim f(y|x^{(t)})$

After a sufficient number of iterations, the samples  $(x^{(t)}, y^{(t)})$  will approximate the joint distribution  $f(x, y)$ . **Note:** We can use inverse CDF to sample from the conditional distributions.

### 3 L3

**Definition** (Monte Carlo Markov Chains). Generate a markov chain  $X_1, X_2, \dots$  by Taking  $X_n = P(\cdot | X_{n-1})$ .  $P$  is the transition Kernel with denisty:  $p(t|x_k) = \frac{dP}{dt}$   
If  $f(x_{k+1}) = \int f(x_k)p(x_{k+1}|x_k)dx_k$  tghen we say the MC has an invariant probability measure

**Theorem 1.** Suppose a MC has a density function such that it is an invariant chain. The chain is aldo  $\pi$ -irreducible and aperiodic. Then as  $n \rightarrow \infty$ , the distribution of  $X_n$  converges to  $\pi$  regardless of the initial distribution of  $X_0$ .

**Definition** (Metropolis-Hastings Algorithm). Goal: Try to generate a RV  $X \sim f(x)$  where  $f$  is known up to a constant but difficult to sample from directly.

Idea: Propsal distribution  $g(\cdot|x)$  easy to sample from and contruct a MC with invariant distribution  $f$ .

**Algorithm:** Set initial value  $x_0$

At each new step choose new  $x_{k+1}$  as follows:

- sample  $z \sim g(\cdot|x_k)$
- Accept  $z$  as  $x_{k+1}$  with probability:

$$\alpha(x_k, z) = \min \left( 1, \frac{f(z)g(x_k|z)}{f(x_k)g(z|x_k)} \right)$$

- else reject z and set  $x_{k+1} = x_k$

The intuition behind this method is that we are using the proposal distribution  $g(\cdot|x)$  to generate candidate samples and then accepting or rejecting them based on how well they match the target distribution  $f(x)$ . The acceptance probability  $\alpha(x_k, z)$  ensures that the Markov chain has the desired invariant distribution.

The transition kernel of the M-H is given by:

$$p(x_{k+1}|x_k) = 1_{x_{k+1} \neq x_k} g(x_{k+1}|x_k) \alpha(x_k, x_{k+1}) + 1_{x_{k+1} = x_k} \int g(z|x_k)(1 - \alpha(x_k, z)) dz$$

**Corollary** (Special M-H). Original M-H where  $g(\cdot|x)$  is symmetric, ie:  $g(a|b) = g(b|a)$  for all a,b. Then the acceptance probability simplifies to

$$\alpha(x_k, z) = \min \left( 1, \frac{f(z)}{f(x_k)} \right)$$

Independence chain M-H where  $g(\cdot|x) = g(\cdot)$  independent of x. Then the acceptance probability simplifies to

$$\alpha(x_k, z) = \min \left( 1, \frac{f(z)g(x_k)}{f(x_k)g(z)} \right)$$

**Example** (M-H algorithm). simulate from  $f(x) = \begin{cases} \frac{1}{c} f_{t_5}(x)\{1 - \sin(20x)\} & |x| < 3 \\ 0 & \text{else} \end{cases}$  And  $c = \int_{-3}^3 f_{t_5}(x)\{1 - \sin(20x)\}dx$  where  $f_{t_5}(x)$  is the density of t-distribution with 5 degrees of freedom.

Use independence chain M-H with proposal distribution  $g(x) = f_{t_5}(x)$ . Then the acceptance probability is given by:

$$\alpha(x_k, z) = \min\left(1, \frac{1 - \sin(20z)}{1 - \sin(20x_k)}\right) \mathbb{1}_{|z| < 3}$$

Thus we can implement the M-H algorithm as follows: Set initial value  $x_0$

For  $k = 0, 1, 2, \dots, N-1$  do:

- sample  $z \sim f_{t_5}(x)$
- Compute  $\alpha(x_k, z) = \min\left(1, \frac{1 - \sin(20z)}{1 - \sin(20x_k)}\right) \mathbb{1}_{|z| < 3}$
- set  $x_{k+1} = \begin{cases} z & \text{with probability } \alpha(x_k, z) \\ x_k & \text{else} \end{cases}$

**Definition** (Pearson Correlation Coefficient). Population Version:  $\rho_{X,Y} = \frac{\text{Cov}(X,Y)}{\sqrt{\text{Var}(X)\text{Var}(Y)}} = \frac{E[XY] - E[X]E[Y]}{\sqrt{E[X^2] - (E[X])^2}\sqrt{E[Y^2] - (E[Y])^2}}$  If  $X, Y$  are independent then  $\rho_{X,Y} = 0$  but the converse is not true.

If  $X, Y$  are jointly normal then  $\rho_{X,Y} = 0$  implies independence.

Sample Version:  $\hat{\rho} = \frac{\frac{1}{n} \sum x_i y_i - \bar{x}\bar{y}}{s_1 s_2}$  where  $\bar{x} = \frac{1}{n} \sum x_i$ ,  $\bar{y} = \frac{1}{n} \sum y_i$ ,  $s_1^2 = \frac{1}{n} \sum (x_i - \bar{x})^2$ ,  $s_2^2 = \frac{1}{n} \sum (y_i - \bar{y})^2$

**Definition** (Spearman's  $\rho$ ). It is used to measure the strength and direction of the monotonic relationship between two variables. It is a non-parametric measure that assesses how well the relationship between two variables can be described using a monotonic function.

Suppose  $X, Y \sim F(x, y)$  with marginals  $F_1(x), F_2(y)$

Denote their densities by  $f(x, y), f_1(x), f_2(y)$  respectively.

$$p_s(X, Y) = \rho_{F_1(X), F_2(Y)} = \frac{\text{Cov}(F_1(X), F_2(Y))}{\sqrt{\text{Var}(F_1(X))\text{Var}(F_2(Y))}}$$

$$= 12 \left( \int \int F_1(x) F_2(y) f(x, y) dx dy \right) - 3$$

Sample Version:  $\hat{p}_s = \frac{\frac{1}{n} \sum r_i^{(x)} r_i^{(y)} - \bar{r}^{(x)} \bar{r}^{(y)}}{s_r(x) s_r(y)}$  Where  $r_i^{(x)}$  is the rank of  $x_i$  among  $x_1, x_2, \dots, x_n$

**Definition** (Kendall's Tau). Most effective for non-linear relationships/ related to ranks.

Population Version: Suppose  $X, Y \sim F(x, y)$  and  $(X', Y') \sim F(x, y)$  independent copies.

$$\tau_K = P((X - X')(Y - Y') > 0) - P((X - X')(Y - Y') < 0)$$

IE the concordance probability - discordance probability.

Sample Version: Suppose  $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$  are n independent observations.

We check all pairs  $(X_i, Y_i), (X_j, Y_j)$  for  $1 \leq i < j \leq n$ .

Let C be the number of concordant pairs and D be the number of discordant pairs.

Then  $\hat{\tau}_K = \frac{C - D}{\frac{n(n-1)}{2}}$  similarly we can write:  $\hat{\tau}_K = 4 \int \int F(x, y) f(x, y) dx dy - 1$

**Definition** (Copula). Copulas is a mapping from  $[0, 1]^n$  to  $[0, 1]$  that joins univariate marginals to form multivariate distribution.

The copular for a pair  $X \sim F_1(x), Y \sim F_2(y)$  is given by:

$$C(t, s) = F(F_1^{-1}(t), F_2^{-1}(s))$$

$$C(t, s) = F(F_1^{-1}(t), F_2^{-1}(s)) \iff F(x, y) = C(F_1(x), F_2(y))$$

where  $F_1^{-1}, F_2^{-1}$  are the inverse CDFs of  $X$  and  $Y$  respectively.

This is just another way to define a joint distribution using the marginals and a copula function.

**Definition** (Kendall's Tau of Copula). We can define the Kendall's Tau of a copula as:

$$\tau_K = 4 \int_0^1 \int_0^1 C(u, v) dC(u, v) - 1$$

where  $C(u, v)$  is the copula function. This measures the dependence structure captured by the copula.

**Definition** (Gaussian Copula). Recall bivariate normal distribution with correlation  $\rho$ :

$$(X, Y) \sim N \left( \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix}, \begin{pmatrix} \sigma_X^2 & \rho \sigma_X \sigma_Y \\ \rho \sigma_X \sigma_Y & \sigma_Y^2 \end{pmatrix} \right)$$

When we do a direct calculation we get:

$$C(s, t) = \int_{-\infty}^{\Phi^{-1}(t)} \int_{-\infty}^{\Phi^{-1}(s)} \frac{1}{2\pi\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}(x^2 - 2\rho xy + y^2)\right) dx dy$$

When  $\rho = 0$  we get  $C(s, t) = st$  which is the independence copula. When  $\rho = 1$  we get  $C(s, t) = \min(s, t)$  which is the comonotonicity copula.

**Corollary** (General Case). Consider  $n$  covariates  $X_1, X_2, \dots, X_n$  with marginals  $F_1(x), F_2(x), \dots, F_n(x)$  respectively. The copula is given by:

$$C(t_1, t_2, \dots, t_n) = F(F_1^{-1}(t_1), F_2^{-1}(t_2), \dots, F_n^{-1}(t_n))$$

**Definition** (Cholesky Factorization). Any symmetric positive definite matrix  $\Sigma$  can be written as:

$$\Sigma = LL^T$$

where L is a lower triangular matrix with positive diagonal entries. This is known as the Cholesky factorization of  $\Sigma$ . It is useful for simulating multivariate normal distributions and for solving systems of linear equations. For a 2x2 matrix:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

The Cholesky factorization is given by:

$$L = \begin{pmatrix} \sigma_1 & 0 \\ \rho\sigma_2 & \sigma_2\sqrt{1-\rho^2} \end{pmatrix}$$

For an  $n \times n$  matrix, we can compute the entries of L as follows:

$$a_{ij} = \begin{cases} \sqrt{\sigma_{ii} - \sum_{k=1}^{j-1} a_{ik}^2} & \text{if } i = j \\ \frac{1}{a_{jj}} (\sigma_{ij} - \sum_{k=1}^{j-1} a_{ik}a_{jk}) & \text{if } i > j \\ 0 & \text{if } i < j \end{cases}$$

The algorithm to calculate the Cholesky factorization is as follows: **Algorithm**

For  $j = 1$  to  $n$  do:

- For  $i = j$  to  $n$  do:
  - $v_i = \sigma_{ij}$
  - For  $k = 1$  to  $j - 1$  do:
    - \*  $v_i = v_i - a_{ik}a_{jk}$
    - $a_{ij} = \frac{v_i}{\sqrt{v_j}}$

**Definition** (Simulate Multivar Normal RV). We have the assumptions:  $X \sim N(\mu, \Sigma)$  and  $\Sigma = LL^T$  where L is the Cholesky factorization of  $\Sigma$ .

**Algorithm:** Get the Cholesky factorization L of  $\Sigma$

Simulate  $Z = (Z_1, Z_2, \dots, Z_n)$  where  $Z_i \sim N(0, 1)$  independent

Set  $X = \mu + LZ$

Then  $X \sim N(\mu, \Sigma)$

This is due to the fact that if Z is a standard normal vector, then any linear transformation of Z will also be normally distributed.

**Definition** (Gaussian Copula Simulation). Goal: Generate Dependent Random numbers such the marginals  $x_i \sim F_i(x)$  for  $i = 1, 2, \dots, n$  and the dependence structure is given by a Gaussian copula with correlation matrix R.

Hidden assumption, co-dependence is assumed to be determined by a covariance matrix  $\Sigma$  through Gaussian copula.

**Algorithm:** Get the Cholesky factorization L of R

Simulate  $Z = (Z_1, Z_2, \dots, Z_n)$  where  $Z_i \sim N(0, 1)$  independent

Set  $Z^* = LZ \sim N(0, \Sigma)$

for  $i = 1, 2, \dots, n$  do:

- $u_i = \Phi(Z_i^*/\sigma_i) \sim U(0, 1)$
- $x_i = F_i^{-1}(u_i) \sim F_i(x)$

$x_1, x_2, \dots, x_n$  have the desired marginals and dependence structure.

**Remark.** From observed data to Gaussian copula:  $x_1, x_2, \dots, x_n$  observed data

For each company  $i$  use its observed data to get an emirical cdf  $\hat{F}_i(x)$

Where  $\hat{F}_i(t) = \frac{\sum_{k=1}^m 1(x_{i,k} < t)}{m}$  for  $i = 1, 2, \dots, n$

define  $y_{i,k} = \Phi^{-1}(\hat{F}_i(x_{i,k}))$  for  $i = 1, 2, \dots, n$  and  $k = 1, 2, \dots, m$

Then estimate the correlation matrix R by the sample correlation of  $y_{i,k}$  across companies by  $\hat{\Sigma} = \frac{1}{m-1} \sum_{k=1}^m (y_{i,k} - \bar{y}_i)(y_{j,k} - \bar{y}_j)$

**Definition** (Student T copula). Similar to Gaussian copula but using multivariate t-distribution instead.

Goal: Generate Dependent Random numver such the marginals  $x_i \sim F_i(x)$  for  $i = 1, 2, \dots, n$  and the dependence structure is given by a Student t copula with correlation matrix R and degrees of freedom  $\gamma$ .

**Algorithm:** Get the Cholesky factorization L of R

Simulate  $Z = (Z_1, Z_2, \dots, Z_n)$  where  $Z_i \sim N(0, 1)$  independent

Set  $Z^* = LZ \sim N(0, \Sigma)$

Simulate  $s \sim \chi_\gamma^2$  independent

for  $i = 1, 2, \dots, n$  do:

- $u_i = F_{t_\gamma}(\frac{z_i^*/\sigma_i}{\sqrt{s/\gamma}}) \sim U(0, 1)$

- $x_i = F_i^{-1}(u_i) \sim F_i(x)$

$x_1, x_2, \dots, x_n$  have the desired marginals and dependence structure

The math is complicated but similar to Gaussian copula. and the algorithm is similar.

**Definition** (Archimedian Copula). For a strickly devreasing convex function  $h(t) : (0, 1] \rightarrow [0, \infty)$  we get a copula:

$$C(t, s) = h^{-1}(h(t) + h(s))$$

where  $h^{-1}$  is the inverse function of h.

Given the marginals  $F_1(x), F_2(y)$  we get the joint distribution:

$$F(x, y) = C(F_1(x), F_2(y)) = h^{-1}(h(F_1(x)) + h(F_2(y)))$$

**Algorithm:** Simulate  $u_1, u_2 \sim U(0, 1)$  independent

Solve for  $r$  in the equation:  $u_2 = r - \frac{h(r)}{h'(r)}$

Set:  $s = h^{-1}(u_1 h(r))$  and  $t = h^{-1}((1 - u_1)h(r))$

Then  $x = F_1^{-1}(s)$  and  $y = F_2^{-1}(t)$  have the desired marginals and dependence structure.

## 4 L5

**Definition** (Bootstrap). Bootstrap method: Method to estimate the distribution of a statistic by resampling with replacement from the observed data.

Bootstrap samples: Given observed data  $X_1, X_2, \dots, X_n$  we generate bootstrap samples  $X_1^*, X_2^*, \dots, X_n^*$  by sampling with replacement from the original data. Each bootstrap sample is of size n.

Bootstrap distribution: For a statistic  $\hat{\theta} = s(X_1, X_2, \dots, X_n)$  we compute the statistic for each bootstrap sample to get  $\hat{\theta}^* = s(X_1^*, X_2^*, \dots, X_n^*)$ . The distribution of  $\hat{\theta}^*$  across all bootstrap samples is the bootstrap distribution of the statistic.

The motivation is statistical inference, we want to estimate the distribution of a statistic (like mean, variance, etc.) without making strong parametric assumptions about the underlying population. By resampling from the observed data, we can approximate the sampling distribution of the statistic and use it for inference (like confidence intervals, hypothesis testing, etc.)

**Algorithm:** From the observed data resample with replacement to get a bootstrap sample.

Compute the statistic of interest for the bootstrap sample.

Repeat the above two steps B times to get B bootstrap statistics.

Then order the bootstrap statistics in ascending order.

To get a  $(1 - \alpha)100\%$  confidence interval for the statistic, (in a symetric case) for  $[\hat{\theta}_L, \hat{\theta}_U]$  we set:

$$L = \frac{\alpha N}{2}, U = 1 - \frac{\alpha}{2}N$$

IE the lower and upper bounds are the  $\frac{\alpha}{2}$  and  $1 - \frac{\alpha}{2}$  percentiles of the bootstrap statistics.

For the asymmetric case we set:

$$[2\hat{\theta} - \hat{\theta}_U^*, 2\hat{\theta} - \hat{\theta}_L^*]$$

where  $\hat{\theta}$  is the statistic computed from the original data, and  $\hat{\theta}_L^*$  and  $\hat{\theta}_U^*$  are the lower and upper bounds from the bootstrap distribution. And we set  $L = \frac{\alpha N}{2}, U = 1 - \frac{\alpha}{2}N$  as before. Cheat Sheet

**Example** (Bootstrap algo). Step 1: at each iteration 1 to 1000 generate bootstrap sample of 20 by sampling with replacement from the original data.

These new samples are  $X_1^*, X_2^*, \dots, X_{20}^*$ .

Step 2: compute the sample correlation coefficient  $\hat{\rho}^*$  for the bootstrap sample.

Step 3: repeat steps 1 and 2 for B = 1000 times to get  $\hat{\rho}_1^*, \hat{\rho}_2^*, \dots, \hat{\rho}_{1000}^*$ .

Step 4: order the bootstrap statistics in ascending order.

Step 5: to get a 95% confidence interval for the correlation coefficient, we set:

$$L = 0.025 \times 1000 = 25, U = 1 - 0.025 \times 1000 = 975$$

Thus the 95% confidence interval is  $[2\hat{\rho} - \hat{\rho}_{975}^*, 2\hat{\rho} - \hat{\rho}_{25}^*]$  where  $\hat{\rho}$  is the sample correlation coefficient computed from the original data and since the distribution is asymmetric we use the asymmetric formula.

**Definition** (Bootstrap CLT). Under some regularity conditions, as the number of bootstrap samples B approaches infinity,

$$(\hat{\theta}^* - \hat{\theta})|\hat{\theta} \xrightarrow{d} (\hat{\theta} - \theta_0)|\theta_0$$

where  $\theta_0$  is the true parameter value. This means that the distribution of the bootstrap statistic centered at the original statistic converges in distribution to the distribution of the original statistic centered at the true parameter value.

**Definition** (Proofs of CI using BSCLT). pg 24-29 Lecture 5 notes

**Definition** (Alternative applications of BS sample method). Approx standard error of statistic:

$$\hat{s}_{e_{BS}} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{\theta}_i^* - \hat{\theta}^*)^2}$$

where  $\hat{\theta}^* = \frac{1}{N} \sum_{i=1}^N \hat{\theta}_i^*$

Bias correction: Often  $\text{Bias}(\hat{\theta}) = \hat{\theta} - \theta_0$

$$\text{Bias}_{BS}(\hat{\theta}) = \frac{1}{N} \sum_{i=1}^N \hat{\theta}_i^* - \hat{\theta}$$

linear regression:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \quad \epsilon_i \sim N(0, \sigma^2)$$

Least squares estimate:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

**Definition** (Resampling). We can resample data pairs as we have been doing so far.

We can also resample residuals.

Step 1: Fit the linear regression model to get  $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$  and residuals  $\hat{\epsilon}_i = y_i - \hat{y}_i$  for  $i = 1, 2, \dots, n$ .

Step 2: Generate bootstrap residuals  $\hat{\epsilon}_1^*, \hat{\epsilon}_2^*, \dots, \hat{\epsilon}_n^*$  by sampling with replacement from the original residuals.

Step 3: Create bootstrap response variables  $y_i^* = \hat{y}_i + \hat{\epsilon}_i^*$  for  $i = 1, 2, \dots, n$ .

Step 4: Fit the linear regression model to the bootstrap sample  $(x_i, y_i^*)$  to get the bootstrap estimate  $\hat{\beta}_1^*$ .

Step 5: Repeat steps 2 to 4 for B times to get  $\hat{\beta}_{1,1}^*, \hat{\beta}_{1,2}^*, \dots, \hat{\beta}_{1,B}^*$ .

Step 6: Use the bootstrap estimates to compute the confidence interval for  $\beta_1$  using the same methods as before (percentile method, bias-corrected method, etc.)

## 5 L6

**Definition** (Bayesian Approach). We want to estimate an unknown parameter  $\theta$  given observed data  $X = (X_1, X_2, \dots, X_n)$ .

We start with a prior distribution  $p(\theta)$  that reflects our beliefs about the parameter before seeing the data.

We then use Bayes' theorem to update our beliefs about the parameter after seeing the data:

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)}$$

where  $p(X|\theta)$  is the likelihood function and  $p(X)$  is the marginal likelihood.

The parameter  $\theta$  has a prior distribution  $\pi(\theta)$  and Infrence on the paramter is based on its posterior distribution given the data:

$$\pi(\theta|y) = \frac{f(y|\theta)\pi(\theta)}{\int f(y|\theta)\pi(\theta)d\theta}$$

where  $f(y|\theta)$  is the likelihood function.

**Definition** (Priors). Elicited Prior: Based on expert knowledge or historical data.

Conjugate Prior: A prior that, when combined with the likelihood, results in a posterior distribution of the same family as the prior. This simplifies calculations.

Non-informative Prior: A prior that is intentionally vague or flat, indicating little prior knowledge about the parameter. Examples include uniform priors or Jeffreys priors.

**Definition** (Bayesian Infrence). Point estimation: Use the posterior mean, median, or mode as a point estimate for the parameter.

Interval estimation: Use credible intervals derived from the posterior distribution to quantify uncertainty about the parameter. A credible interval is an interval within which the parameter lies with a certain probability (e.g., 95% credible interval).

**Definition** (Bayes Factor). The Bayes factor is the ratio of the posterial odds of model  $M1$  to the prior odds of model  $M1$  vs those of model  $M2$ :

$$BF = \frac{P(M1|y)/P(M2|y)}{P(M1)/P(M2)} = \frac{P(y|M1)}{P(y|M2)}$$

The scale of evidence provided by the Bayes factor is as follows:

- $< \frac{1}{10}$ : Strong evidence for  $M2$
- $\frac{1}{10}$  to  $\frac{1}{3}$ : Moderate evidence for  $M2$
- $\frac{1}{3}$  to 1: Weak evidence for  $M2$
- 1 to 3: Weak evidence for  $M1$
- 3 to 10: Moderate evidence for  $M1$
- $> 10$ : Strong evidence for  $M1$

**Definition** (Bayesian Analysis). It requires 2 main components. Likelihood:  $f(y|\theta)$  and Prior:  $\pi(\theta)$ .

The posterior distribution is given by:

$$\pi(\theta|y) = \frac{f(y|\theta)\pi(\theta)}{\int f(y|\theta)\pi(\theta)d\theta}$$

where the denominator is the marginal likelihood.

**Example** (Bayesian Infrence example). 16 consumers compared flavor of patties, patties where in a high quality freezer vs a low quality freezer. 13 out of 16 preferred the high quality freezer.

**Likelihood:**  $Y|\theta \sim \text{Binomial}(n = 16, \theta)$  where  $\theta$  is the proportion of consumers who prefer high quality freezer.

We choose this since we assume that consumer are indepentent and the proabbility  $\theta$  is the same for all consumers. This is the same as  $f(y|\theta) = \binom{16}{y} \theta^y (1 - \theta)^{16-y}$  for  $y = 0, 1, \dots, 16$ .

**Prior:** We choose a Beta prior  $\theta \sim \text{Beta}(\alpha, \beta)$  ie  $\pi(\theta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \theta^{\alpha-1} (1 - \theta)^{\beta-1}$  for  $0 < \theta < 1$ .

If we choose  $\alpha = 1, \beta = 1$  we get a uniform prior which is non-informative,  $\alpha = .5, \beta = .5$  we get a Jeffreys prior which is also non-informative.  $\alpha = 2, \beta = 2$  we get a prior that favors values around 0.5 ie a skeptical prior.

Thanks to conjugacy, the posterior distribution is also a Beta distribution:

$$\begin{aligned} p(\theta|y) &\propto f(y|\theta)\pi(\theta) \\ &\propto \beta(y + \alpha, n - y + \beta) \end{aligned}$$

Where  $n = 16, y = 13$ .

Now we get multiple options for point estimation: We can select .6 as the crtical value that  $\theta$  must exceed inorder for the improvement to be considered significant.

Thus  $M1 : \theta > .6$  vs  $M2 : \theta \leq .6$

We can compute the Bayes factor:

$$\begin{aligned} BF &= \frac{P(y|M1)}{P(y|M2)} \\ &= \frac{\int_0^1 f(y|\theta)\pi(\theta)d\theta}{\int_0^6 f(y|\theta)\pi(\theta)d\theta} \end{aligned}$$

This ends up with 31.1 which is strong evidence for  $M1$  over  $M2$ .

**Definition** (Point estimation). Posterior mean or median can be used as point estimates.

$$\hat{\theta}_{Bayes} = E[\theta|y] = \int \theta \pi(\theta|y) d\theta$$

**Example** (Point Estimate with bootstrap).  $\hat{\theta} = \frac{1}{m} \sum_{i=1}^m \theta_i^*$  For CI we sort the bootstrap statistics and set:

$$L = \frac{\alpha N}{2}, U = 1 - \frac{\alpha}{2}N$$

Thus the 95% confidence interval is  $[\hat{\theta}_{(L)}^*, \hat{\theta}_{(U)}^*]$  where  $\hat{\theta}_{(L)}^*$  and  $\hat{\theta}_{(U)}^*$  are the L-th and U-th ordered bootstrap statistics.

**Example** (Linear Model). put it on cheat sheet Observation  $y|\beta, \sigma^2 \sim N(X\beta, \sigma^2 I)$  where  $X$  is the design matrix.

Prior:  $\beta|\sigma^2 \sim N(\beta_0, \sigma^2 B_0)$ ,  $\sigma^2 \sim \mathcal{G}(c, C_0)$  where  $\mathcal{G}$  is the inverse gamma distribution.

Posterior:

$$\begin{aligned} p(\beta, \sigma^2 | y) &\propto p(y|\beta, \sigma^2)p(\beta|\sigma^2)p(\sigma^2) \\ &\propto \frac{1}{\sigma}^n \exp\left(-\frac{1}{2\sigma^2}(y - X\beta)^T(y - X\beta)\right) \\ &\times \frac{1}{\sigma}^p \exp\left(-\frac{1}{2\sigma^2}(\beta - \beta_0)^T B_0^{-1}(\beta - \beta_0)\right) \\ &\times \frac{1}{\sigma^2}^{c_0+1} \exp\left(-\frac{C_0}{\sigma^2}\right) \end{aligned}$$

The posterior follows a normal-inverse-gamma distribution: Which can be shown that

$$\hat{\beta}_{Bayes} = (X^T X + B_0^{-1})^{-1} (B_0^{-1} b_0 + X^T y)$$

With  $A = (X^T X + B_0^{-1})^{-1} X^T X$

We can interpret the parameter as the weighted mean of prior expectations  $b_0$  and OLS estimate:

$$\hat{\beta}_{Bayes} = (I - A)b_0 + A\hat{\beta}_{OLS}$$

When  $B_0$  has large diagonal entries,  $A \approx I$  and  $\hat{\beta}_{Bayes} \approx \hat{\beta}_{OLS}$ .

When  $B_0$  has small diagonal entries,  $A \approx 0$  and  $\hat{\beta}_{Bayes} \approx b_0$ .

**Definition** (Gibbs Sampler for Bayes).