Multivariate Normal distribution: introduction

Suppose that we want to generate data from a Multivariate Normal distribution with a specific covariance structure (nonzero correlations).

The PDF of a Multivariate Normal distribution is:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} \mid \mathbf{\Sigma} \mid^{1/2}} \exp\left(- \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right)$$

with $\boldsymbol{\mu} = (\mu_1, ..., \mu_d)^T$ and $\mathbf{\Sigma}$ is a d x d symmetric positive definite covariance matrix whose general element $cov(X_i, X_j) = \rho_{ij} s_i s_j$

Gibbs sampling algorithm

The Gibbs sampling procedure to generate random variates from a $N_2(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the following:

At each iteration, do the following

1. Set $(x_1, x_2) = X(t-1)$
2. Generate $X_1^*(t)$ from $f(X_1 \mid x_2) \sim N\left(\mu_1 + \frac{\rho s1}{s2}(x_2 - \mu_2), (1-\rho^2)s_1^2\right)$
3. Update $x_1 = X_1^*(t)$.
4. Generate $X_2^*(t)$ from $f(X_2 \mid x_1) \sim N\left(\mu_2 + \frac{\rho s2}{s1}(x_1 - \mu_1), (1-\rho^2)s_2^2\right)$
5. Set $X(t) = (X_1^*(t), X_2^*(t))$

Example of a centered bivariate sample

Example Suppose that we want to generate a random sample from a $N_2(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with the following mean vector and covariance matrix:

$$\boldsymbol{\mu} = (0,0)^T \qquad \text{and} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

That is, that we defined the following standard deviations and correlation coefficients:

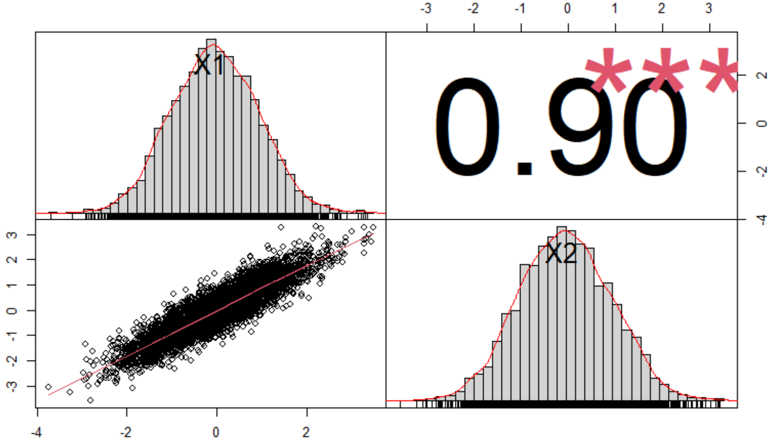
$$\rho_{12} = 0.9 \qquad s_1 = 1 \qquad s_2 = 1$$

Then we want to fit a simple linear regression model $X_2 = \beta_0 + \beta_1 X_1$ to the sample.

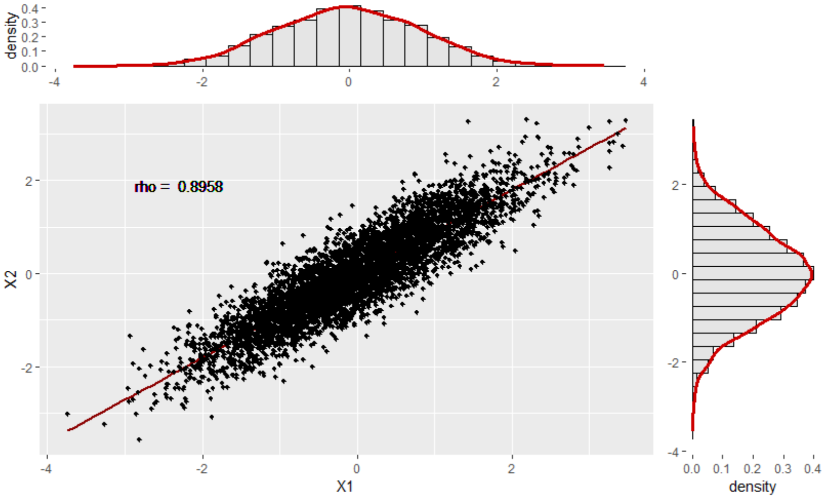
Implementation in the R language

```r
# 2. Gibbs sampling function to sample from Bivariate Normal distribution
mv.Gibbs_sampler <- function(n, burn, mu1, mu2, s1, s2, rho) {

  X <- matrix(0, n, 2) # the chain
  X[1,] <- c(mu1, mu2) # initial observation

  for(i in 2:n) {
    x2 <- X[i-1, 2]
    m1 <- mu1 + (x2 - mu2) * rho * (s1/s2)
    X[i, 1] <- rnorm(1, mu1 + (x2 - mu2) * rho * (s1/s2),
                     sqrt(1-rho^2) * s1)

    x1 <- X[i, 1]
    X[i, 2] <- rnorm(1, mu2 + (x1 - mu1) * rho * (s2/s1),
                     sqrt(1-rho^2) * s2)
  }

  b <- burn + 1
  x <- X[b:n, ]

  return(data.frame(x))
}
```

Visualizing the joint distribution



Another visualization



Cholesky factorization algorithm

The Cholesky algorithm to simulate random variates from a $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the following:

1. Generate an n x d matrix $\mathbf{Z}$ containing nd random variates $N(0, 1)$

2. Decompose the covariance matrix using Cholesky factorization $\boldsymbol{\Sigma} = \mathbf{Q^T Q}$, where $\mathbf{Q}$ is an upper triangular matrix.

3. Apply the transformation $\mathbf{X} = \mathbf{ZQ} + \mathbf{1_n}\boldsymbol{\mu}^T$, where $\mathbf{1_n}$ is a column vector of ones of size n.

4. Return the n x d matrix $\mathbf{X}$ for which each row is a random vector from the $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Example of a centered bivariate sample

Example Suppose that we want to generate a random sample from a $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with the following mean vector and covariance matrix:

$$\boldsymbol{\mu} = (0, 0, 2)^T \qquad \text{and} \qquad \boldsymbol{\Sigma} = \begin{pmatrix} 4 & -1.5 & 0.8 \\ -1.5 & 1 & -0.1 \\ 0.8 & -0.1 & 1 \end{pmatrix}$$

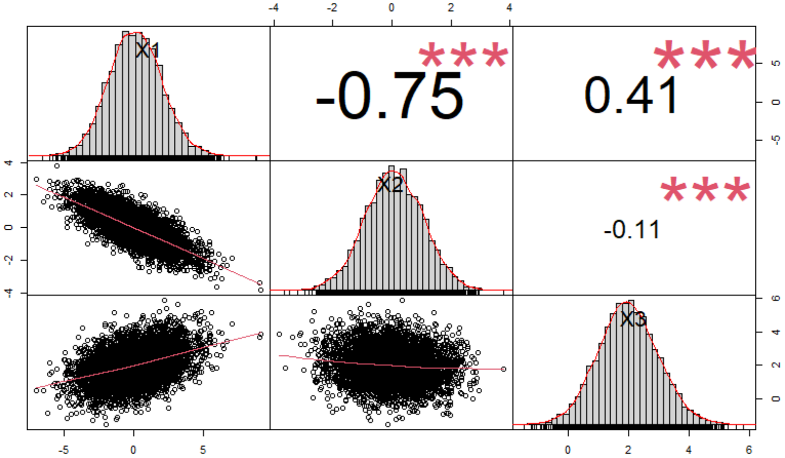
That is, that we defined the following standard deviations and correlation coefficients:

$$\begin{aligned} \rho_{12} &= -0.75 & \qquad s_1 &= 2 \\ \rho_{13} &= 0.4 & \qquad s_2 &= 1 \\ \rho_{23} &= -0.1 & \qquad s_3 &= 1 \end{aligned}$$

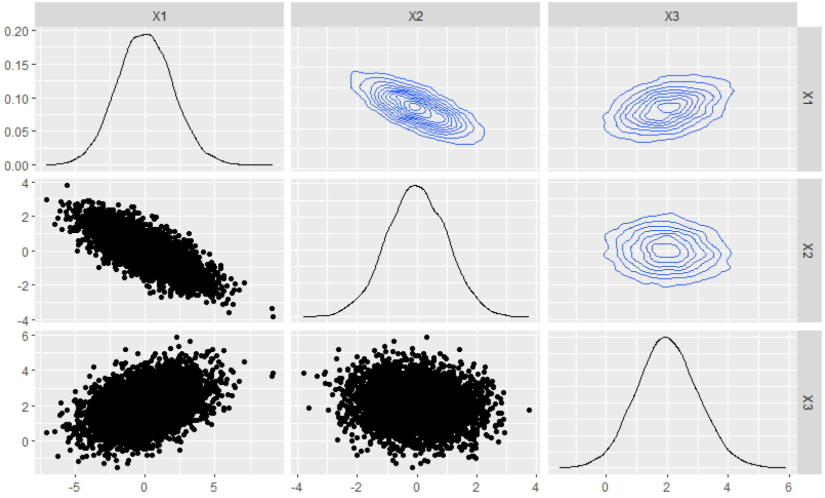
Implementation in the R language

```r
# 2. Mean and covariance matrix (cov_12 = s1*s2*rho_12)
mu <- c(mu1, mu2, mu3)

Sigma <- matrix(c(s1^2, s1*s2*rho_12, s1*s3*rho_13,
                  s2*s1*rho_12, s2^2, s2*s3*rho_23,
                  s1*s3*rho_13, s2*s3*rho_23, s3^2), nrow = 3)


# 3. Cholesky function to sample from multivariate Normal distribution
mv_cholesky <- function(n, mu, Sigma) {

  d <- length(mu)
  Q <- chol(Sigma)
  Z <- matrix(rnorm(n*d), nrow = n, ncol = d)
  X <- Z %*% Q +  rep(1,n) %*% t(mu)
  X <- data.frame(X)
  return(X)
}
```

Visualizing the joint distribution



Another visualization



References

Rizzo, M.L. (2019). Statistical Computing with R, Second Edition (2nd ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9780429192760

The R Project for Statistical Computing: https://www.r-project.org/