# Problem 3.2

Chen Bo Calvin Zhang

19/10/2020

We want to generate a random sample from a multivariate normal distribution, given the number of samples wanted, the mean and the covariance matrix, without relying on mnormt or similar packages.

We will first use the univariate normal random number generator rnorm() availabe in R to create standard multivariate normal data. Then we will employ a coloring transformation to transform this data to multivariate normal data with the desired mean and covariance Sigma.

$$E(\boldsymbol{x}) = \mathbf{0} \text{ and } Var(\boldsymbol{x}) = \boldsymbol{I} \implies E(\boldsymbol{y}) = \boldsymbol{\mu} \text{ and } Var(\boldsymbol{y}) = \boldsymbol{\Sigma}$$

And inverse Mahalanobis coloring transformation

$$\boldsymbol{y} = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{X}$$

```r
myrmnorm = function(n, mu, Sigma) {

  d = length(mu)

  # generate data using a standard univariate normal random number generator
  univ_data = rnorm(n * d)
  # place into a matrix so that it is a a sample from a standard multivariate sample
  X = matrix(univ_data, nrow = n)

  # eigendecomposition of Sigma
  eigen = eigen(Sigma)
  lambda = eigen$values
  U = eigen$vectors

  #inverse square root of Sigma
  sqrt_Sigma = U %*% diag(lambda^(1/2)) %*% t(U)

  # make matrix from mu
  MU = t(replicate(n, mu))
  # use Mahalanobis coloring transformation to set mean and covariance
  Y = MU + X %*% sqrt_Sigma

  return(Y)
}
```

Let us test the method with some data.

```r
mu = c(1, 2, 3, 4, 5, 6)
Sigma = matrix(0.8, nrow = 6, ncol = 6)
diag(Sigma) = c(6, 5, 4, 3, 2, 1)

gen_data = myrmnorm(100, mu, Sigma)
```

```r
print(colMeans(gen_data))
```

```
## [1] 1.005828 1.969754 3.091823 3.987081 4.941007 5.976169
```

```r
print(cov(gen_data))
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] 6.7548137 1.1309384 1.6392375 0.7916287 0.7380113 0.9118163
## [2,] 1.1309384 4.3088806 0.8845779 0.5423363 0.3630767 0.4860354
## [3,] 1.6392375 0.8845779 4.7946780 0.8637140 0.8053003 1.1287523
## [4,] 0.7916287 0.5423363 0.8637140 2.6061686 0.4232571 0.4768465
## [5,] 0.7380113 0.3630767 0.8053003 0.4232571 1.5180255 0.5030684
## [6,] 0.9118163 0.4860354 1.1287523 0.4768465 0.5030684 0.7986551
```