Cholesky Decomposition

This algorithm is implemented in the class *RBasicCholeskyDecomposition* and takes a symmetric positive definite matrix **A** and splits it into the product

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T \tag{1}$$

where **L** is a lower-triangular matrix of the same dimension as **A**. Once the decomposition is complete the algorithms for forward and backward substitution are applied to determine the solution the linear system of equations **Ax**=**b**. However, the Cholesky decomposition has other uses, such as in probability transformations to standardize a vector of random variables. The approach takes its name from Andre-Louis Cholesky (1857-1918) and takes a symmetric and positive definite matrix as input. Several algorithms exist for obtaining **L** and its inverse. One calculates the components of the matrix row-by-row from left to right:

$$L_{ij} = \begin{cases} \sqrt{A_{ij} - \sum_{k=0}^{i-1} L_{ik}^2} & \text{for } i = j \\ \dfrac{A_{ij} - \sum_{k=0}^{j-1} \left(L_{ik} L_{jk}\right)}{L_{jj}} & \text{for } i > j \\ 0 & \text{for } i < j \end{cases} \tag{2}$$

followed by the calculation of $\mathbf{L}^{-1}$ row-by-row from right to left:

$$invL_{ij} = \begin{cases} \dfrac{1}{L_{ii}} & \text{for } i = j \\ -\dfrac{\sum_{k=j}^{i-1} \left(L_{ik} \cdot invL_{kj}\right)}{L_{jj}} & \text{for } i > j \\ 0 & \text{for } i < j \end{cases} \tag{3}$$