1 Algorithms

1.1 LU Decomposition

Algorithm 1 Doolittle Decomposition

Require: A, an $n \times n$ matrix
  for $i \leftarrow 1$ to n **do**
    if $a_{ii} = 0$ **then**
      if $a_{ji} \neq 0$ for any $i < j \leq n$ **then**
        report error and return
      end if
      continue
    end if
    for $j \leftarrow i + 1$ to n **do**
      $l_{ji} \leftarrow \frac{a_{ji}}{a_{ii}}$
      for $k \leftarrow 1$ to n **do**
        $a_{jk} \leftarrow a_{jk} - a_{ik} * l_{ji}$
      end for
    end for
  end for
  return L, A

Algorithm 2 Crout Decomposition

Require: A, an $n \times n$ matrix
  for $i \leftarrow 1$ to n **do**
    $j \leftarrow 1$
    while $j \leq i$ **do**
      $l_{ij} \leftarrow a_{ij} - \sum_{k=1}^{j-1} l_{ik} * u_{kj}$
      $j \leftarrow j + 1$
    end while
    if $l_{ii} = 0$ **and** $i < n$ **then**
      report error and return
    end if
    while $j \leq n$ **do**
      $u_{ij} \leftarrow \frac{1}{l_{ii}} * \left(a_{ij} - \sum_{k=1}^{i-1} l_{ik} * u_{kj} \right)$
      $j \leftarrow j + 1$
    end while
  end for
  return L, U

Algorithm 3 Cholesky Decomposition

Require: A, an $n \times n$ **symmetric** matrix
  $j \leftarrow 1$
  while $j < i$ **do**
    if $l_{jj} = 0$ **then**
      report error and return
    end if
    $l_{ij} \leftarrow \frac{1}{l_{jj}} * \left(\sum_{k=1}^{j-1} l_{ik} * l_{jk} \right)$
    $j \leftarrow j + 1$
  end while
  $l_{ij} \leftarrow \sqrt{a_{ij} - \sum_{k=1}^{i-1} l_{ik}^2}$
  return L