线性方程组的直接解法

上机题6

按照上课所讲，实现了Choelsky分解法，将 H 矩阵分解为 LL^T 的形式，其中 L 为下三角矩阵，则原方程化为 $LL^Tx = b$，先解 $Ly = b$，再解 $L^Tx = y$。即可得到 x。

运行代码 `python P6.py`，得到结果如下：

```
b = [2.92896825 2.01987734 1.60321068 1.34680042 1.16822899 1.03489566
 0.93072899 0.84669538 0.77725094 0.7187714 ]
norm(r) = 4.440892098500626e-16
norm(e) = 6.943372649304003e-05

b = [2.92896855 2.01987764 1.60321097 1.34680071 1.16822929 1.03489595
 0.93072929 0.84669567 0.77725123 0.7187717 ]
norm(r) = 4.440892098500626e-16
norm(e) = 2.0506893658671603

n = 8
norm(r) = 4.440892098500626e-16
norm(e) = 6.025062604386733e-08

n = 10
norm(r) = 4.440892098500626e-16
norm(e) = 6.943372649304003e-05

n = 12
norm(r) = 2.220446049250313e-16
norm(e) = 0.5521155258554964

*******/P6.py:15: RuntimeWarning: invalid value encountered in sqrt
  L[j, j] = np.sqrt(H[j, j] - np.sum(L[j, :j] ** 2))
n = 14
norm(r) = nan
norm(e) = nan
```

其中第一部分的结果计算了 $n = 10$ 时，使用Choelsky分解法求解线性方程组的残差和误差。可以看到此时的残差和误差都很小，

第二部分是加上了 10^{-7} 的扰动（相对变化量），可以发现此事的残差依旧很小，但是误差很大。

后面是 $n \in \{8, 10, 12, 14\}$ 时的结果，可以看到 $n = 8, 10$ 时的残差和误差都较小。但是当 $n = 12$ 时，残差依旧很小，但是误差已经变得有些大了。当 $n = 14$ 时会发生错误，观察程序的中间结果发现，这是因为误差的累积，在计算 $L[j, j]$ 时，由于 $H[j, j] - \sum_{k=1}^{j-1} L[j, k]^2$ 的值小于0，导致了无法计算 $L[j, j]$，从而导致了错误。

通过这个实验，这个算法对于 b 的扰动是很敏感的。同时当 n 较大时，由于每一行使用的数据都是前面行运算后的数据，所以误差会逐渐累积，导致最后的结果不准确。