Lapack 的网站地址为　　http://www.netlib.org/lapack/

Lapack 又叫线性代数软件包，里面包含了各种矩阵的运算，例如本征值问题的求解，它通常是需要矩阵的基本运算库 blas 库，两个库一起使用。

用户手册地址为　http://www.netlib.org/lapack/lug/

- Contents of LAPACK
  - What's new in version 3.0?
  - Structure of LAPACK
    - Levels of Routines
    - Data Types and Precision
    - Naming Scheme
  - Driver Routines
    - Linear Equations
    - Linear Least Squares (LLS) Problems
    - Generalized Linear Least Squares (LSE and GLM) Problems
    - Standard Eigenvalue and Singular Value Problems
    - Generalized Eigenvalue and Singular Value Problems
  - Computational Routines
    - Linear Equations
    - Orthogonal Factorizations and Linear Least Squares Problems
    - Generalized Orthogonal Factorizations and Linear Least Squares Problems
    - Symmetric Eigenproblems
    - Nonsymmetric Eigenproblems
    - Singular Value Decomposition
    - Generalized Symmetric Definite Eigenproblems
    - Generalized Nonsymmetric Eigenproblems
    - Generalized (or Quotient) Singular Value Decomposition

1. Generalized Symmetric Definite Eigenproblems

This section is concerned with the solution of the generalized eigenvalue problems （广义本征值问题）

$Az = \lambda Bz$, $ABz = \lambda z$, and $BAz = \lambda z$,

where A and B are real symmetric or complex Hermitian and B is positive definite.

Each of these problems can be reduced to a standard symmetric eigenvalue problem, using a Cholesky（乔列斯基）factorization of B

Hence the eigenvalues of

$Az = \lambda Bz$

are those of $Cy = \lambda y$

Table 2.13: Reduction of generalized symmetric definite eigenproblems to standard problems

	Type of	Factorization	Reduction	Recovery of
	problem	of B		eigenvectors
1.	$Az = \lambda Bz$	$B = LL^T$	$C = L^{-1} A L^{-T}$	$z = L^{-T} y$
		$B = U^T U$	$C = U^{-T} A U^{-1}$	$z = U^{-1} y$
2.	$ABz = \lambda z$	$B = LL^T$	$C = L^T A L$	$z = L^{-T} y$
		$B = U^T U$	$C = U A U^T$	$z = U^{-1} y$
3.	$BAz = \lambda z$	$B = LL^T$	$C = L^T A L$	$z = L y$
		$B = U^T U$	$C = U A U^T$	$z = U^T y$

The routines **xyy**GST overwrite A with the matrix C of the corresponding standard problem

Table 2.14: Computational routines for the generalized symmetric definite eigenproblem

Type of matrix	Operation	Single precision		Double precision	
and storage scheme		real	complex	real	complex
symmetric/Hermitian	reduction	SSYGST	CHEGST	DSYGST	ZHEGST
symmetric/Hermitian	reduction	SSPGST	CHPGST	DSPGST	ZHPGST
(packed storage)					
symmetric/Hermitian	split Cholesky	SPBSTF	CPBSTF	DPBSTF	ZPBSTF
banded	factorization				
	reduction	SSBGST	DSBGST	CHBGST	ZHBGST

After $\boldsymbol{B}$ has been factorized in this way by the routine xPBSTF the reduction of the banded generalized problem

$$Az = \lambda Bz$$

to a banded standard problem

$$Cy = \lambda y$$

is performed by the routine xHBGST (or xHBGST for complex matrices). This routine implements a vectorizable form of the algorithm, suggested by Kaufman

根据以上的分析，这样我们计算的是对称厄米双精度复数双矩阵，因此使用 LAPACK 程序包的是 ZHEGST 这个程序完成这个将广义本征值问题化简为标准本征值问题。

2. Symmetric Eigenproblems

Let ***A*** be a real symmetric or complex Hermitian ***n***-by-***n*** matrix. A scalar λ is called an **eigenvalue** and a nonzero column vector ***z*** the corresponding **eigenvector**

if $Az = \lambda z$. λ is always real when ***A*** is real symmetric or complex Hermitian.

The basic task of the symmetric eigenproblem routines is to compute values of λ and, optionally, corresponding vectors ***z*** for a given matrix ***A***.

This computation proceeds in the following stages:

1. The real symmetric or complex Hermitian matrix ***A*** is reduced to **real tridiagonal form *T***. If ***A*** is real symmetric this decomposition is $\boldsymbol{A=QTQ^T}$ with ***Q*** orthogonal and ***T*** symmetric tridiagonal. If ***A*** is complex Hermitian, the decomposition is $\boldsymbol{A=QTQ^H}$ with ***Q*** unitary and ***T***, as before, *real* symmetric tridiagonal.

2. Eigenvalues and eigenvectors of the real symmetric tridiagonal matrix ***T*** are computed. If all eigenvalues and eigenvectors are computed, this is equivalent to factorizing ***T*** as $T = S\Lambda S^T$, where ***S*** is orthogonal and Λ is diagonal. The diagonal entries of Λ are the eigenvalues of ***T***, which are also the eigenvalues of ***A***, and the columns of ***S*** are the eigenvectors of ***T***; the eigenvectors of ***A*** are the columns of ***Z=QS***, so that $A = Z\Lambda Z^T$ ($Z\Lambda Z^H$ when ***A*** is complex Hermitian).

In the real case, the decomposition $\boldsymbol{A = Q\ T\ Q^T}$ is computed by one of the routines xSYTRD, **x**SPTRD, or **x**SBTRD,

The complex analogues of these routines are called **x**HETRD, **x**HPTRD, and **x**HBTRD

The routine xSYTRD (or xHETRD) represents the matrix ***Q*** （A real orthogonal or complex unitary matrix (usually denoted Q)）as a product of elementary reflectors.

The routine xORGTR (or in the complex case xUNMTR) is provided to form ***Q*** explicitly; this is needed in particular before calling xSTEQR to compute all the eigenvectors of ***A*** by the *QR* algorithm.

The routine **x**ORMTR (or in the complex case xUNMTR) is provided to multiply another matrix by $\boldsymbol{Q}$ without forming $\boldsymbol{Q}$ explicitly; this can be used to transform eigenvectors of $\boldsymbol{T}$ computed by **x**STEIN, back to eigenvectors of $\boldsymbol{A}$.

When packed storage is used, the corresponding routines for forming $\boldsymbol{Q}$ or multiplying another matrix by $\boldsymbol{Q}$ are xOPGTR and xOPMTR (in the complex case, xUPGTR and xUPMTR).

When $\boldsymbol{A}$ is banded and xSBTRD (or xHBTRD) is used to reduce it to tridiagonal form, $\boldsymbol{Q}$ is determined as a product of Givens rotations, not as a product of elementary reflectors; if $\boldsymbol{Q}$ is required, it must be formed explicitly by the reduction routine. xSBTRD is based on the vectorizable algorithm due to Kaufman.

There are several routines for computing eigenvalues and eigenvectors of $\boldsymbol{T}$, to cover the cases of computing some or all of the eigenvalues, and some or all of the eigenvectors. In addition, some routines run faster in some computing environments or for some matrices than for others. Also, some routines are more accurate than other routines.

Table 2.10: Computational routines for the symmetric eigenproblem

Type of matrix	Operation	Single precision		Double precision	
and storage scheme		real	complex	real	complex
dense symmetric	tridiagonal reduction	SSYTRD	CHETRD	DSYTRD	ZHETRD
(or Hermitian)					
packed symmetric	tridiagonal reduction	SSPTRD	CHPTRD	DSPTRD	ZHPTRD
(or Hermitian)					
band symmetric	tridiagonal reduction	SSBTRD	CHBTRD	DSBTRD	ZHBTRD
(or Hermitian)					
orthogonal/unitary	generate matrix after	SORGTR	CUNGTR	DORGTR	ZUNGTR
	reduction by xSYTRD				
	multiply matrix after	SORMTR	CUNMTR	DORMTR	ZUNMTR
	reduction by xSYTRD				
orthogonal/unitary	generate matrix after	SOPGTR	CUPGTR	DOPGTR	ZUPGTR
(packed storage)	reduction by xSPTRD				
	multiply matrix after	SOPMTR	CUPMTR	DOPMTR	ZUPMTR
	reduction by xSPTRD				

symmetric	eigenvalues/	SSTEQR	CSTEQR	DSTEQR	ZSTEQR
tridiagonal	eigenvectors via QR				
	eigenvalues only	SSTERF		DSTERF	
	via root-free QR				
	eigenvalues/	SSTEDC	CSTEDC	DSTEDC	ZSTEDC
	eigenvectors via				
	divide and conquer				
	eigenvalues/	SSTEGR	CSTEGR	DSTEGR	ZSTEGR
	eigenvectors via				
	RRR				
	eigenvalues only	SSTEBZ		DSTEBZ	
	via bisection				
	eigenvectors by	SSTEIN	CSTEIN	DSTEIN	ZSTEIN
	inverse iteration				
symmetric	eigenvalues/	SPTEQR	CPTEQR	DPTEQR	ZPTEQR
tridiagonal	eigenvectors				
positive definite					

根据以上分析，我们需要程序 ZHETRD，**ZSTEQR 这两个程序求解标准本征值问题，第一个程序是利用 Householder 该方法把矩阵化为三对角矩阵。第二个程序是对三对角矩阵求解本征值问题**