Review of Linear Algebra

Max Turgeon

STAT 4690–Applied Multivariate Analysis

Basic Matrix operations

Matrix algebra and `R`

- Matrix operations in `R` are *very* fast.
- This includes various class of operations:
    - Matrix addition, scalar multiplication, matrix multiplication, matrix-vector multiplication
    - Standard functions like determinant, rank, condition number, etc.
    - Matrix decompositions, e.g. eigenvalue, singular value, Cholesky, QR, etc.
    - Support for *sparse* matrices, i.e. matrices where a significant number of entries are exactly zero.

Matrix functions i

```r
A <- matrix(c(1, 2, 3, 4), nrow = 2, ncol = 2)
A
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    2    4
```

```r
# Determinant
det(A)
```

```
## [1] -2
```

Matrix functions ii

```r
# Rank
library(Matrix)
rankMatrix(A)
```

```
## [1] 2
## attr(,"method")
## [1] "tolNorm2"
## attr(,"useGrad")
## [1] FALSE
## attr(,"tol")
## [1] 4.440892e-16
```

Matrix functions iii

```r
# Condition number
kappa(A)
```

```
## [1] 18.77778
```

```r
# How to compute the trace?
sum(diag(A))
```

```
## [1] 5
```

Matrix functions iv

```r
# Transpose
t(A)
```

```
##      [,1] [,2]
## [1,]    1    2
## [2,]    3    4
```

```r
# Inverse
solve(A)
```

Matrix functions v

```
##      [,1] [,2]
## [1,]   -2  1.5
## [2,]    1 -0.5
```

```
A %*% solve(A) # CHECK
```

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

Matrix operations i

```r
A <- matrix(c(1, 2, 3, 4), nrow = 2, ncol = 2)
B <- matrix(c(4, 3, 2, 1), nrow = 2, ncol = 2)

# Addition
A + B
```

```
##      [,1] [,2]
## [1,]    5    5
## [2,]    5    5
```

Matrix operations ii

```r
# Scalar multiplication
3*A
```

```
##      [,1] [,2]
## [1,]    3    9
## [2,]    6   12
```

```r
# Matrix multiplication
A %*% B
```

Matrix operations iii

```
##      [,1] [,2]
## [1,]   13    5
## [2,]   20    8
```

```r
# Hadamard product aka entrywise multiplication
A * B
```

```
##      [,1] [,2]
## [1,]    4    6
## [2,]    6    4
```

Matrix operations iv

```r
# Matrix-vector product
vect <- c(1, 2)
A %*% vect
```

```
##       [,1]
## [1,]    7
## [2,]   10
```

```r
# BE CAREFUL: R recycles vectors
A * vect
```

Matrix operations v

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    4    8
```

Eigenvalues and Eigenvectors

Eigenvalues

- Let $\mathbf{A}$ be a square $n \times n$ matrix.
- The equation

$$\det(\mathbf{A} - \lambda I_n) = 0$$

  is called the *characteristic equation* of $\mathbf{A}$.
- This is a polynomial equation of degree n, and its roots are called the *eigenvalues* of $\mathbf{A}$.

Example

Let

$$\mathbf{A} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}.$$

Then we have

$$\begin{aligned} \det(\mathbf{A} - \lambda I_2) &= (1 - \lambda)^2 - 0.25 \\ &= (\lambda - 1.5)(\lambda - 0.5) \end{aligned}$$

Therefore, $\mathbf{A}$ has two (real) eigenvalues, namely

$$\lambda_1 = 1.5, \lambda_2 = 0.5.$$

A few properties

Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of $\mathbf{A}$ (with multiplicities).

1. $\mathrm{tr}(\mathbf{A}) = \sum_{i=1}^{n} \lambda_i$;
2. $\det(\mathbf{A}) = \prod_{i=1}^{n} \lambda_i$;
3. The eigenvalues of $\mathbf{A}^k$ are $\lambda_1^k, \ldots, \lambda_n^k$, for k a nonnegative integer;
4. If $\mathbf{A}$ is invertible, then the eigenvalues of $\mathbf{A}^{-1}$ are $\lambda_1^{-1}, \ldots, \lambda_n^{-1}$.

Eigenvectors

- If λ is an eigenvalues of $\mathbf{A}$, then (by definition) we have $\det(\mathbf{A} - \lambda I_n) = 0$.
- In other words, the following equivalent statements hold:
    - The matrix $\mathbf{A} - \lambda I_n$ is singular;
    - The kernel space of $\mathbf{A} - \lambda I_n$ is nontrivial (i.e. not equal to the zero vector);
    - The system of equations $(\mathbf{A} - \lambda I_n)v = 0$ has a nontrivial solution;
    - There exists a nonzero vector v such that

$$\mathbf{A}v = \lambda v.$$

- Such a vector is called an *eigenvector* of $\mathbf{A}$.

Example (cont'd) i

Recall that we had

$$\mathbf{A} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix},$$

and we determined that 0.5 was an eigenvalue of $\mathbf{A}$.

We therefore have

$$\mathbf{A} - 0.5 I_2 = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}.$$

Example (cont'd) ii

As we can see, any vector v of the form $(x, -x)$ satisfies

$$(\mathbf{A} - 0.5I_2)v = (0, 0).$$

In other words, we not only get a single eigenvector, but a whole subspace of $\mathbb{R}^2$. By convention, we usually select as a represensative a vector of norm 1, e.g.

$$v = \left(\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}\right).$$

Example (cont'd) iii

Alternatively, instead of finding the eigenvector by inspection, we can use the reduced row-echelon form of $\mathbf{A} - 0.5I_2$, which is given by

$$\mathbf{A} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

Therefore, the solutions to $(\mathbf{A} - 0.5I_2)v$, with $v = (x, y)$ are given by a single equation, namely $y + x = 0$, or $y = -x$.

Eigenvalues and eigenvectors in R i

```
A <- matrix(c(1, 0.5, 0.5, 1), nrow = 2)

result <- eigen(A)

names(result)
```

```
## [1] "values"  "vectors"
```

```
result$values
```

```
## [1] 1.5 0.5
```

Eigenvalues and eigenvectors in R ii

```
result$vectors
```

```
##              [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

```
1/sqrt(2)
```

```
## [1] 0.7071068
```

Symmetric matrices i

- A matrix $\mathbf{A}$ is called *symmetric* if $\mathbf{A}^T = \mathbf{A}$.
- **Proposition 1**: If $\mathbf{A}$ is (real) symmetric, then its eigenvalues are real.

Proof: Let λ be an eigenvalue of $\mathbf{A}$, and let $v \neq 0$ be an eigenvector corresponding to this eigenvalue. Then we have

Symmetric matrices ii

$$\begin{aligned}
\lambda \bar{v}^T v &= \bar{v}^T (\lambda v) \\
&= \bar{v}^T (\mathbf{A} v) \\
&= (\mathbf{A}^T \bar{v})^T v \\
&= (\mathbf{A} \bar{v})^T v && (\mathbf{A} \text{ is symmetric}) \\
&= (\overline{\mathbf{A} v})^T v && (\mathbf{A} \text{ is real}) \\
&= \bar{\lambda} \bar{v}^T v.
\end{aligned}$$

Since we have $v \neq 0$, we conclude that $\lambda = \bar{\lambda}$, i.e. λ is real.

$\square$

Symmetric matrices iii

- **Proposition 2**: If $\mathbf{A}$ is (real) symmetric, then eigenvectors corresponding to distinct eigenvalues are orthogonal.

Proof: Let λ_1, λ_2 be distinct eigenvalues, and let $v_1 \neq 0, v_2 \neq 0$ be corresponding eigenvectors. Then we have

Symmetric matrices iv

$$\begin{aligned}\lambda_1 v_1^T v_2 &= (\mathbf{A}v_1)^T v_2 \\ &= v_1^T \mathbf{A}^T v_2 \\ &= v_1^T \mathbf{A} v_2 \qquad (\mathbf{A} \text{ is symmetric}) \\ &= v_1^T (\lambda_2 v_2) \\ &= \lambda_2 v_1^T v_2.\end{aligned}$$

Since $\lambda_1 \neq \lambda_2$, we conclude that $v_1^T v_2 = 0$, i.e. v_1 and v_2 are orthogonal. $\square$

Spectral Decomposition i

- Putting these two propositions together, we get the *Spectral Decomposition* for symmetric matrices.
- **Theorem**: Let $\mathbf{A}$ be an $n \times n$ symmetric matrix, and let $\lambda_1 \geq \cdots \geq \lambda_n$ be its eigenvalues (with multiplicity). Then there exist vectors $v_1, \ldots, v_n$ such that
  1. $\mathbf{A}v_i = \lambda_i v_i$, i.e. v_i is an eigenvector, for all i;
  2. If $i \neq j$, then $v_i^T v_j = 0$, i.e. they are orthogonal;
  3. For all i, we have $v_i^T v_i = 1$, i.e. they have unit norm;
  4. We can write $\mathbf{A} = \sum_{i=1}^n \lambda_i v_i v_i^T$.

Sketch of a proof:

Spectral Decomposition ii

1. We are saying that we can find n eigenvectors. This means that if an eigenvalue λ has multiplicity m (as a root of the characteristic polynomial), then the dimension of its *eigenspace* (i.e. the subspace of vectors satisfying $\mathbf{A}v = \lambda v$) is also equal to m. This is not necessarily the case for a general matrix $\mathbf{A}$.
2. If $\lambda_i \neq \lambda_j$, this is simply a consequence of Proposition 2. Otherwise, find a basis of the eigenspace and turn it into an orthogonal basis using the Gram-Schmidt algorithm.
3. This is one is straightforward: we are simply saying that we can choose the vectors so that they have unit norm.

4. First, note that if Λ is a diagonal matrix with $\lambda_1, \ldots, \lambda_n$ on the diagonal, and P is a matrix whose i-th column is v_i, then $\mathbf{A} = \sum_{i=1}^n \lambda_i v_i v_i^T$ is equivalent to

$$\mathbf{A} = P\Lambda P^T.$$

Then 4. is a consequence of the change of basis theorem: if we change the basis from the standard one to $\{v_1, \ldots, v_n\}$, then $\mathbf{A}$ acts by scalar multiplication in each direction, i.e. it is represented by a diagonal matrix Λ.

□

Examples i

We looked at

$$\mathbf{A} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix},$$

and determined that the eigenvalues where $1.5, 0.5$, with corresponding eigenvectors $\left(1/\sqrt{2}, 1/\sqrt{2}\right)$ and $\left(1/\sqrt{2}, -1/\sqrt{2}\right)$.

Examples ii

```r
v1 <- c(1/sqrt(2), 1/sqrt(2))
v2 <- c(1/sqrt(2), -1/sqrt(2))

Lambda <- diag(c(1.5, 0.5))
P <- cbind(v1, v2)

P %*% Lambda %*% t(P)
```

```
##      [,1] [,2]
## [1,]  1.0  0.5
## [2,]  0.5  1.0
```

Examples iii

```r
# Now let's look at a random matrix----
A <- matrix(rnorm(3 * 3), ncol = 3, nrow = 3)
# Let's make it symmetric
A[lower.tri(A)] <- A[upper.tri(A)]
A
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.2550974 -0.5047826  0.2166169
## [2,] -0.5047826 -0.2792298  0.0953815
## [3,]  0.2166169  0.0953815 -0.5734243
```

Examples iv

```r
result <- eigen(A, symmetric = TRUE)
Lambda <- diag(result$values)
P <- result$vectors

P %*% Lambda %*% t(P)
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.2550974 -0.5047826  0.2166169
## [2,] -0.5047826 -0.2792298  0.0953815
## [3,]  0.2166169  0.0953815 -0.5734243
```

Examples v

```r
# How to check if they are equal?
all.equal(A, P %*% Lambda %*% t(P))
```

```
## [1] TRUE
```

Positive-definite matrices

Let $\mathbf{A}$ be a real symmetric matrix, and let $\lambda_1 \geq \cdots \geq \lambda_n$ be its (real) eigenvalues.

1. If $\lambda_i > 0$ for all i, we say $\mathbf{A}$ is *positive definite*.
2. If the inequality is not strict, if $\lambda_i \geq 0$, we say $\mathbf{A}$ is *positive semidefinite*.
3. Similary, if $\lambda_i < 0$ for all i, we say $\mathbf{A}$ is *negative definite*.
4. If the inequality is not strict, if $\lambda_i \leq 0$, we say $\mathbf{A}$ is *negative semidefinite*.

Note: If $\mathbf{A}$ is *positive-definite*, then it is invertible!

Matrix Square Root i

- Let $\mathbf{A}$ be a positive semidefinite symmetric matrix.
- By the Spectral Decomposition, we can write

$$\mathbf{A} = P\Lambda P^T.$$

- Since $\mathbf{A}$ is positive-definite, we know that the elements on the diagonal of Λ are positive.
- Let $\Lambda^{1/2}$ be the diagonal matrix whose entries are the square root of the entries on the diagonal of Λ.
- For example:

$$\Lambda = \begin{pmatrix} 1.5 & 0 \\ 0 & 0.5 \end{pmatrix} \Rightarrow \Lambda^{1/2} = \begin{pmatrix} 1.2247 & 0 \\ 0 & 0.7071 \end{pmatrix}.$$

Matrix Square Root ii

- We define the square root $\mathbf{A}^{1/2}$ of $\mathbf{A}$ as follows:

$$\mathbf{A}^{1/2} := P\Lambda^{1/2}P^T.$$

- *Check*:

$$\begin{aligned}
\mathbf{A}^{1/2}\mathbf{A}^{1/2} &= (P\Lambda^{1/2}P^T)(P\Lambda^{1/2}P^T) \\
&= P\Lambda^{1/2}(P^TP)\Lambda^{1/2}P^T \\
&= P\Lambda^{1/2}\Lambda^{1/2}P^T \qquad (P \text{ is orthogonal}) \\
&= P\Lambda P^T \\
&= \mathbf{A}.
\end{aligned}$$

Matrix Square Root iii

- *Be careful*: your intuition about square roots of positive real numbers doesn't translate to matrices.
    - In particular, matrix square roots are **not** unique (unless you impose further restrictions).

Cholesky Decomposition

- The most common way to obtain a square root matrix for a positive definite matrix $\mathbf{A}$ is via the *Cholesky decomposition*.
- There exists a unique matrix L such that:
  - L is lower triangular (i.e. all entries above the diagonal are zero);
  - The entries on the diagonal are positive;
  - $\mathbf{A} = LL^T$.
- For matrix square roots, the Cholesky decomposition should be prefered to the eigenvalue decomposition because:
  - It is computationally more efficient;
  - It is numerically more stable.

Example i

```r
A <- matrix(c(1, 0.5, 0.5, 1), nrow = 2)

# Eigenvalue method
result <- eigen(A)
Lambda <- diag(result$values)
P <- result$vectors
A_sqrt <- P %*% Lambda^0.5 %*% t(P)

all.equal(A, A_sqrt %*% A_sqrt) # CHECK
```

```
## [1] TRUE
```

Example ii

```r
# Cholesky method
# It's upper triangular!
(L <- chol(A))
```

```
##      [,1]      [,2]
## [1,]    1 0.5000000
## [2,]    0 0.8660254
```

```r
all.equal(A, t(L) %*% L) # CHECK
```

```
## [1] TRUE
```

Power method

Introduction to numerical algebra

- As presented in these notes, we can find the eigenvalue decomposition by
  1. Finding the roots of a degree n polynomial.
  2. For each root, find the solutions to a system of linear equations.
- Problem: no exact formula for roots of a generic polynomial when $n > 4$.
  - So we need to find approximate solutions
- Other problem: approximation errors for eigenvalues propagate to eigenvectors
- **Need more stable algorithms**
- This is what numerical algebra is about. For a good reference, I recommend *Matrix Computations* by Golub and Van Loan.

Power Method i

- We'll discuss one approach to finding the leading eigenvector, i.e. the eigenvector corresponding to the largest eigenvalue (in absolute value).
- **Note**: We have to assume that the largest eigenvalue (in absolute value) is unique.
- *Algorithm*:
  1. Let v_0 be an initial vector with unit norm.
  2. At step k, define

$$v_{k+1} = \frac{\mathbf{A}v_k}{\|\mathbf{A}v_k\|},$$

  where $\|v\|$ is the norm of the vector v.

Power Method ii

3. Then the sequence v_k converges to the desired eigenvector.
4. The corresponding eigenvalue is defined by

$$\lambda = \lim_{k\to\infty} \frac{v_k^T \mathbf{A} v_k}{v_k^T v_k}.$$

- Comment: unless v_0 is orthogonal to the eigenvector we are looking for, we have theoretical guarantees of convergence.
    - In practice, we can pick v_0 randomly, since the probability a random vector is orthogonal to the eigenvector is zero.

Example i

```r
set.seed(123)

A <- matrix(rnorm(3*3), ncol = 3)
# Make A symmetric
A[lower.tri(A)] <- A[upper.tri(A)]

# Set initial value
v_current <- rnorm(3)
v_current <- v_current/norm(v_current, type = "2")
```

Example ii

```
# We'll perform 100 iterations
for (i in seq_len(100)) {
  # Save result from previous iteration
  v_previous <- v_current
  # Compute matrix product
  numerator <- A %*% v_current
  # Normalize
  v_current <- numerator/norm(numerator, type = "2")
}

v_current
```

Example iii

```
##               [,1]
## [1,] -0.3318109
## [2,]  0.5345952
## [3,]  0.7772448
```

```
# Corresponding eigenvalue
num <- t(v_current) %*% A %*% v_current
denom <- t(v_current) %*% v_current
num/denom
```

```
##             [,1]
## [1,] -1.75374
```

Example iv

```r
# CHECK results
result <- eigen(A, symmetric = TRUE)
result$values[which.max(abs(result$values))]
```

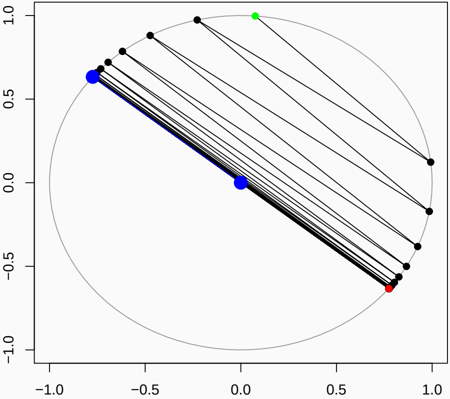
```
## [1] -1.75374
```

```r
result$vectors[,which.max(abs(result$values))]
```

```
## [1]  0.3318109 -0.5345952 -0.7772448
```

- Note that we did not get the same eigenvector: they differ by -1.

Visualization



Blue is the objective; the sequence goes from green to red.

Singular Value Decomposition

Singular Value Decomposition i

- We saw earlier that real symmetric matrices are *diagonalizable*, i.e. they admit a decomposition of the form $P\Lambda P^T$ where
  - Λ is diagonal;
  - P is orthogonal, i.e. $PP^T = P^TP = I$.
- For a general $n \times p$ matrix $\mathbf{A}$, we have the *Singular Value Decomposition* (SVD).
- We can write $\mathbf{A} = UDV^T$, where
  - U is an $n \times n$ orthonal matrix;
  - V is a $p \times p$ orthogonal matrix;
  - D is an $n \times p$ diagonal matrix.

Singular Value Decomposition ii

- We say that:
  - the columns of U are the *left-singular vectors* of $\mathbf{A}$;
  - the columns of V are the *right-singular vectors* of $\mathbf{A}$;
  - the nonzero entries of D are the *singular values* of $\mathbf{A}$.

Existence proof

- First, note that both $\mathbf{A}^T\mathbf{A}$ and $\mathbf{A}\mathbf{A}^T$ are symmetric.
- Therefore, we can write:
  - $\mathbf{A}^T\mathbf{A} = P_1\Lambda_1 P_1^T$;
  - $\mathbf{A}\mathbf{A}^T = P_2\Lambda_2 P_2^T$.
- Moreover, note that $\mathbf{A}^T\mathbf{A}$ and $\mathbf{A}\mathbf{A}^T$ have the **same** nonzero eigenvalues.
- Therefore, if we choose Λ_1 and Λ_2 so that the elements on the diagonal are in descending order, we can choose
  - $U = P_2$;
  - $V = P_1$;
  - The main diagonal of D contains the nonzero eigenvalues of $\mathbf{A}^T\mathbf{A}$ in descending order.

Example i

```r
set.seed(1234)
A <- matrix(rnorm(3 * 2), ncol = 2, nrow = 3)
result <- svd(A)
names(result)
```

```
## [1] "d" "u" "v"
```

```r
result$d
```

```
## [1] 2.8602018 0.6868562
```

Example ii

```
result$u
```

```
##              [,1]         [,2]
## [1,] -0.9182754 -0.359733536
## [2,]  0.1786546 -0.003617426
## [3,]  0.3533453 -0.933048068
```

```
result$v
```

```
##           [,1]       [,2]
## [1,] 0.5388308 -0.8424140
## [2,] 0.8424140  0.5388308
```

Example iii

```r
D <- diag(result$d)
all.equal(A, result$u %*% D %*% t(result$v)) #CHECK
```

```
## [1] TRUE
```

Example iv

```
# Note: crossprod(A) == t(A) %*% A
#  tcrossprod(A) == A %*% t(A)
U <- eigen(tcrossprod(A))$vectors
V <- eigen(crossprod(A))$vectors

D <- matrix(0, nrow = 3, ncol = 2)
diag(D) <- result$d

all.equal(A, U %*% D %*% t(V)) # CHECK
```

```
## [1] "Mean relative difference: 1.95887"
```

Example v

```r
# What went wrong?
# Recall that eigenvectors are unique
# only up to a sign!

# These elements should all be positive
diag(t(U) %*% A %*% V)
```

```
## [1] -2.8602018  0.6868562
```

Example vi

```r
# Therefore we need to multiply the
# corresponding columns of U or V
# (but not both!) by -1
cols_flip <- which(diag(t(U) %*% A %*% V) < 0)
V[,cols_flip] <- -V[,cols_flip]

all.equal(A, U %*% D %*% t(V)) # CHECK
```

```
## [1] TRUE
```