

Eigen Values & Vector

• Rubber band stretch it In each case we do something that affects the shape or the orientation

When we investigate transformations we find that there are some directions which remain the same after a "deformation" has occurred



preserved direction of transformation → eigen vector

associated amount → eigen value

Eigen values are multipliers - they are numbers which

respect how much "Stretching" has taken place.

Eigen = German for self

Example



"Tallening"
⇒

50cm



I am 3.2 times 3.2, acting as
taller than when the eigen value
I was a baby

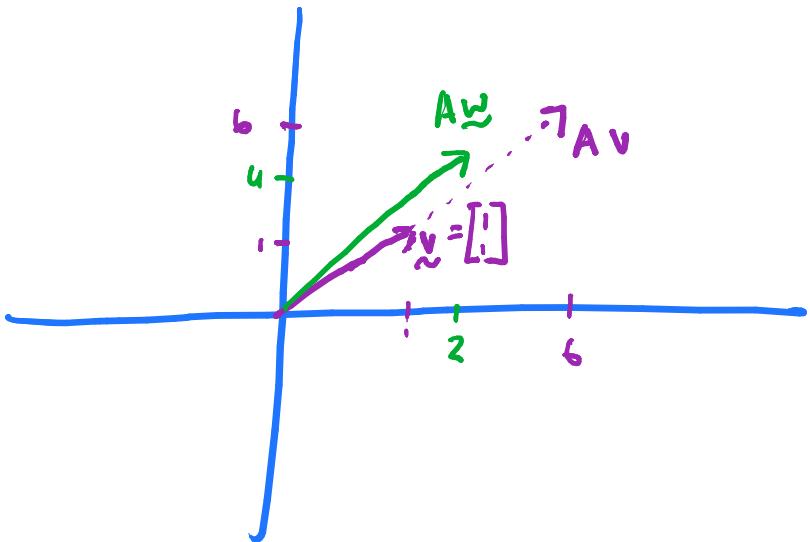
Def Let A be a $n \times n$ matrix ; A number (scalar) λ is called an eigen value of A if there is a non-zero vector v such that

$$Av = \lambda v$$

Vectors satisfying this equation are called eigen vectors of A belonging to λ

Example

$$A = \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}$$



$$Av = \lambda v$$

$$\bullet A \begin{bmatrix} v \\ i \end{bmatrix} = \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \end{bmatrix} = 6 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\bullet A \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \dots = \begin{bmatrix} 2 \\ 4 \end{bmatrix} \Rightarrow \lambda \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

so $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ is not an eigen vector

$$\bullet A \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 2 \\ -2 \end{bmatrix} = 2 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

so $\lambda=2$ eigen value ✓
with corresponding eigen vector $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$

Example

To show scaling factor eigenvalue can be negative z.c.
or even complex.

Suppose $A = \begin{bmatrix} 1 & 3 \\ 2 & 6 \end{bmatrix}$ + $\tilde{v} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}$

$$A \tilde{v} = 0 = 0 \tilde{v} \quad A \tilde{v} = \lambda \tilde{v}$$

So that $\lambda = 0$ is an eigenvalue with corresponding
eigen vector $\begin{bmatrix} 3 \\ -1 \end{bmatrix}$

The action of A in the direction \tilde{v} is to collapse
all vectors to \tilde{v} .

Example

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad v = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad A \tilde{v} = \lambda \tilde{v} ?$$

$$A \tilde{v} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} = (-1) \tilde{v}$$

So $\lambda = -1$ is an eigen value with corresponding eigenvector

$$\tilde{v} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

The action of A in the direction of \tilde{v} is a reflection!

1 Eigenvalues & eigenvectors

Poole Ch. 4

1.1 The Eigen Problem

Problem 1.1. Compute

$$A^{100}\mathbf{x}$$

$$\begin{bmatrix} 5 & 4 \\ 8 & -1 \end{bmatrix} \begin{bmatrix} 5 & 4 \\ 8 & -1 \end{bmatrix}^{\text{100 times}}$$

where

$$A = \begin{bmatrix} -5 & 4 \\ 8 & -1 \end{bmatrix} \quad \text{and} \quad \mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

This looks like a tedious task. However, note that

$$A\mathbf{x} = \begin{bmatrix} 3 \\ 6 \end{bmatrix} = 3\mathbf{x},$$

so that

$$A^{100}\mathbf{x} = 3^{100}\mathbf{x} = \begin{bmatrix} 3^{100} \\ 2 \times 3^{100} \end{bmatrix}.$$

Clearly there is something special about \mathbf{x} . The action of A on \mathbf{x} is to *scale* \mathbf{x} by a factor of 3. If we choose another vector, this does not (necessarily) occur. For example

$$A \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -5 \\ 8 \end{bmatrix} \neq \lambda \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

for any constant λ . So how could we *efficiently* compute

$$\begin{array}{l} \text{- Span } \text{Span} \\ \text{- Linear independent} \end{array} \quad A^{100} \begin{bmatrix} 1 \\ 0 \end{bmatrix}?$$

Suppose we pick a basis for \mathbf{R}^2 that includes \mathbf{x} —say $\{\mathbf{x}, \mathbf{y}\}$. Then

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} = c_1\mathbf{x} + c_2\mathbf{y}$$

for some constants c_1 and c_2 . Now

$$A \begin{bmatrix} 1 \\ 0 \end{bmatrix} = c_1A\mathbf{x} + c_2A\mathbf{y} = 3c_1\mathbf{x} + c_2A\mathbf{y}.$$

It would be nice if we could choose \mathbf{y} so that it also scales under A ; that is

$$A\mathbf{y} = \lambda\mathbf{y}$$

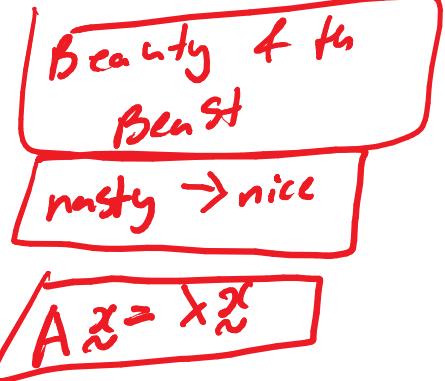
for some constant λ (it would be too much to hope that $\lambda = 3$). Is this possible? In this case, it is!

$$A \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -9 \\ 9 \end{bmatrix} = -9 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

and so $\mathbf{y} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, and $\lambda = -9$. We have

$$A \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 3c_1\mathbf{x} + (-9)c_2\mathbf{y},$$

Poole Ch. 4.1



and therefore

$$A^{100} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 3^{100} c_1 \mathbf{x} + (-9)^{100} c_2 \mathbf{y}.$$

We still need to determine c_1 and c_2 . These are the coordinates of $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ in the basis $\{\mathbf{x}, \mathbf{y}\}$:

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} = c_1 \begin{bmatrix} 1 \\ 2 \end{bmatrix} + c_2 \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

so we need to solve the linear system

$$[\mathbf{x} \quad \mathbf{y}] \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Solving this system gives $c_1 = \frac{1}{3}$, $c_2 = \frac{2}{3}$. Finally, we obtain

$$A^{100} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 3^{100} \frac{1}{3} \mathbf{x} + (-9)^{100} \frac{2}{3} \mathbf{y} = 3^{100} \frac{1}{3} \mathbf{x} + 3^{200} \frac{2}{3} \mathbf{y} \approx 2 \times 3^{199} \mathbf{y}$$

since $3^{200} \gg 3^{100}$.

Question 1.2. Can we always do this? That is, can we find a collection of vectors \mathbf{x}_i which scale under a given $n \times n$ matrix A

$$A\mathbf{x}_i = \lambda_i \mathbf{x}_i$$

which will form a basis for \mathbf{R}^n ?

1.2 Eigenvalues & eigenvectors

Definition 1.3. A non-zero $n \times 1$ vector \mathbf{x} is an *eigenvector* of an $n \times n$ matrix A with associated *eigenvalue* λ if

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

Rearranging this equation (remember $\lambda \mathbf{x} = \lambda I \mathbf{x}$), we have

$$(A - \lambda I)\mathbf{x} = \mathbf{0} \quad \text{Find } \lambda !$$

where I is the $n \times n$ identity matrix. If $A - \lambda I$ is non-singular then $\mathbf{x} = \mathbf{0}$ is the only solution to this equation. Since we require this equation to have a non-zero solution, we must have $A - \lambda I$ singular; that is,

$$\det(A - \lambda I) = 0. \Rightarrow \text{eigenvectors}$$

This equation is called the *characteristic equation* for A . This is a *non-linear* equation for λ . Once we solve this, we must then solve the *linear* equations

$$(A - \lambda_i I)\mathbf{x}_i = \mathbf{0} \Rightarrow \text{eigenvalues}$$

for each solution λ_i of the characteristic equation.

non singular
→ has an
inverse

singular
 $\det B = 0$

Def :
How do we
find them?
↓
| So
| $(A - \lambda I)x = 0$
| $Bx = 0$

Basic Strategy

① solve $\det(A - \lambda I) = 0$ for λ

② solve $(A - \lambda I) \vec{z} = 0$ for \vec{z}

- How many? For any $n \times n$ matrix A always find n eigenvalues
- find if some are repeated, might get $< n$ eigenvectors.

Exercise 1.4. Find the eigenvalues and eigenvectors of

$$A = \begin{bmatrix} -5 & 4 \\ 8 & -1 \end{bmatrix}$$

Notice that if \mathbf{x} is an eigenvector associated with λ then any non-zero multiple of \mathbf{x} is also an eigenvector associated with λ .

Why are eigenvalues and eigenvectors so important? Consider a system of first order ODEs, for unknowns $y_i = y_i(t)$,

$$\begin{aligned} y'_1 &= a_{11}y_1 + a_{12}y_1 + \cdots + a_{1n}y_n \\ y'_2 &= a_{21}y_1 + a_{22}y_1 + \cdots + a_{2n}y_n \\ &\vdots \\ y'_n &= a_{n1}y_1 + a_{n2}y_1 + \cdots + a_{nn}y_n \end{aligned}$$

where a_{ij} are all constants. In vector notation, this system has the form

$$\mathbf{y}' = A\mathbf{y}$$

with

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}.$$

Suppose \mathbf{x} is an eigenvector of A associated with eigenvalue λ and let

$$\mathbf{z} = e^{\lambda t}\mathbf{x}.$$

We have

$$\mathbf{z}' = \lambda e^{\lambda t}\mathbf{x} = e^{\lambda t}\lambda\mathbf{x} = e^{\lambda t}A\mathbf{x} = Ae^{\lambda t}\mathbf{x} = A\mathbf{z}$$

Since \mathbf{x} does *not* depend on t . Therefore \mathbf{z} is a *solution* to the system of differential equations $\mathbf{y}' = A\mathbf{y}$.

Question 1.5. Can we get *all* the solutions to this system by this method?

To answer this question, we must first find out how many *independent* eigenvectors A has.

1.3 The characteristic equation

Poole Ch. 4.3

Let

$$p(\lambda) = \det(A - \lambda I) = \det \begin{bmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} - \lambda & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{bmatrix}.$$

For an $n \times n$ matrix A , $p(\lambda)$ is a polynomial of degree n in λ . It is called the *characteristic polynomial of A* . The eigenvalues of A are the roots of the characteristic polynomial, $\lambda_i, i = 1, 2, \dots, n$. Therefore A has n eigenvalues.

Example

Find the eigenvalues & eigenvectors of

$$A = \begin{bmatrix} -5 & 4 \\ 8 & -1 \end{bmatrix}$$

Strategy:

- ① Solve $\det(A - \lambda I) = 0$ for λ (eigen values)
- ② Solve $(A - \lambda I)\tilde{x} = 0$ for \tilde{x} (eigen vectors)

① Find eigenvalues

$$\det(A - \lambda I) = 0$$

$$\begin{vmatrix} -5 - \lambda & 4 \\ 8 & -1 - \lambda \end{vmatrix} = 0$$

$$(-5 - \lambda)(-1 - \lambda) - 8 \times 4 = 0$$

$$\lambda^2 + 6\lambda - 27 = 0$$

$$(\lambda - 3)(\lambda + 9) = 0$$

$$\lambda = 3, \text{ or } \lambda = -9$$

② For $\lambda = 3$ eigen vector

$$(A - 3I)\tilde{x} = 0$$

$$\left[\begin{array}{cc|c} -8 & 4 & 0 \\ 8 & -4 & 0 \end{array} \right] \Rightarrow \left[\begin{array}{cc|c} -8 & 4 & 0 \\ 0 & 0 & 0 \end{array} \right]$$

↑ 1 pivot
↑ free

eigen vector: $x = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ set $x_2 = s$, where $s \in \mathbb{R}$

$$\tilde{x} = s \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Matlab $\rightarrow \|z\| = 1$

$$\lambda = -q \quad (A - qI) \tilde{x} = 0$$

$$\begin{bmatrix} 4 & 4 & | & 0 \\ 8 & 8 & | & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 4 & 4 & | & 0 \\ 0 & 0 & | & 0 \end{bmatrix}$$

choose eigen vector

$$\tilde{x}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \text{ say}$$

Eigen Spaces

Let A be a $n \times n$ matrix & let λ be an eigenvalue of A . The collection of all eigenvectors corresponding to λ , together with the zero vector, is called the eigen space of λ , E_λ .

Example

$$\begin{bmatrix} 4 & -1 \\ 2 & 1 \end{bmatrix}$$

$$\textcircled{1} \quad \det(A - \lambda I) = 0$$

$$\begin{vmatrix} 4-\lambda & -1-\lambda \\ 2 & 1-\lambda \end{vmatrix} = \dots = (\lambda^2 - 3)(\lambda - 2)$$

• eigen values $\lambda = 3, \lambda = 2$

$$\textcircled{2} \quad (A - \lambda I)\vec{x} = 0$$

for $\lambda = 2$

$$A - 2I = \dots = \begin{bmatrix} 2 & -1 \\ 2 & -1 \end{bmatrix} \Rightarrow \begin{bmatrix} 2 & -1 \\ 0 & 0 \end{bmatrix}$$

so if $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ is an eigen vector corresponding to eigen value 2

$$\text{then } x_2 = 2x_1$$

Eigen vectors are of the form $\vec{x} = \begin{bmatrix} x_1 \\ 2x_1 \end{bmatrix}$

i.e., non-zero multiples of $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$

so

$$E_2 = \text{null}(A - 2I)$$

$$= \text{span}\left\{\begin{bmatrix} 1 \\ 2 \end{bmatrix}\right\}$$

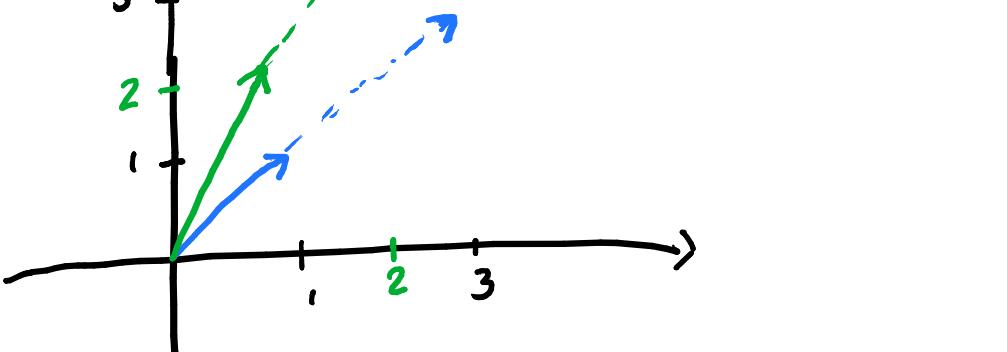
For $\lambda = 3$

$$(A - 3I) = \begin{bmatrix} 1 & -1 \\ 2 & -2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix}$$

so if $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ is an eigen vector associated to eigen value 3 then $x_1 = x_2$

Eigen vectors are of the form $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$E_3 = \text{span}\left\{\begin{bmatrix} 1 \\ 1 \end{bmatrix}\right\}$$



However, some of these roots may be repeated (and some complex); so the eigenvalues may not all be distinct. In terms of the eigenvalues, we may write

$$p(\lambda) = (\lambda_1 - \lambda)(\lambda_2 - \lambda) \cdots (\lambda_n - \lambda) = \prod_{i=1}^n (\lambda_i - \lambda).$$

In particular, if we set $\lambda = 0$ we obtain

$$\det A = p(0) = \lambda_1 \lambda_2 \cdots \lambda_n = \prod_{i=1}^n \lambda_i;$$

that is, the determinant of A is the product of its eigenvalues. The trace of a matrix A is the sum of its diagonal elements; that is

$$\text{tr } A = a_{11} + a_{22} + \cdots + a_{nn} = \sum_{i=1}^n a_{ii}.$$

The coefficient of λ^{n-1} in the factored form of the characteristic polynomial is

$$(-1)^{n-1}(\lambda_1 + \lambda_2 + \cdots + \lambda_n).$$

However this same coefficient computed from the determinant form of the characteristic polynomial is

$$(-1)^{n-1}(a_{11} + a_{22} + \cdots + a_{nn})$$

and therefore

$$\text{tr } A = \lambda_1 + \lambda_2 + \cdots + \lambda_n = \sum_{i=1}^n \lambda_i.$$

That is, the trace of A is the sum of its eigenvalues. Since the trace is easy to compute it acts as a useful check on our computations.

Note that if A is upper or lower triangular then the eigenvalues are simply the entries on the main diagonal. In these two cases, no computation is needed to find the eigenvalues.

Example 1.6. Let

$$A = \begin{bmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{bmatrix}.$$

Find eigen value & vector
of A

① $\det(A - \lambda I) = 0$

② $(A - \lambda I)x = 0$

The characteristic polynomial is

$$\begin{aligned} \det(A - \lambda I) &= \det \begin{bmatrix} -1 - \lambda & -2 & -2 \\ 1 & 2 - \lambda & 1 \\ -1 & -1 & -\lambda \end{bmatrix} \\ &= -(1 + \lambda)(1 - \lambda(2 - \lambda)) + 2(1 - \lambda) - 2(-1 + 2 - \lambda) \\ &= -(\lambda - 1)^2(\lambda + 1). \end{aligned}$$

← characteristic equation

Therefore the three eigenvalues of A are $-1, 1, 1$. As a check, note that

$$\text{tr } A = 1 = -1 + 1 + 1.$$



The eigenvalue 1 is *repeated*. The number of times that it is repeated is called the *algebraic multiplicity* of the eigenvalue. In this case, $\lambda_2 = 1$ has algebraic multiplicity 2.

① $\lambda = 1, 1, -1$
Algebraic multiplicity of $\lambda = 1$ is 2
Algebraic multiplicity of $\lambda = -1$ is 1

1.4 Eigenvectors

Once an eigenvalue λ_i is found, the corresponding eigenvector(s) \mathbf{x}_i are found by solving

$$(A - \lambda_i I)\mathbf{x}_i = \mathbf{0}.$$

Remember that this system will *not* have a unique solution. If it does, then λ_i was not an eigenvalue!

Continuing on our example: The eigenvector associated with $\lambda_1 = -1$ is given by the non-zero solutions of $(A + I)\mathbf{x} = \mathbf{0}$; that is

(2)

$$\left[\begin{array}{ccc|c} 0 & -2 & -2 & 0 \\ 1 & 3 & 1 & 0 \\ -1 & -1 & 1 & 0 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 1 & 2 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right]. \quad \text{row of zeros!}$$

The row reduced augmented matrix *must* have at least one row of zeros (and it can have up to the algebraic multiplicity rows of zeros). In this case we have one free parameter (since there is only one row of zeros). Letting the third component be s , we have

$$\mathbf{x} = \begin{bmatrix} 2s \\ -s \\ s \end{bmatrix} = s \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}. \quad \begin{aligned} x_3 &= s \\ x_2 &= \dots \\ x_1 &= \dots \end{aligned}$$

Thus the solution form a 1-dimensional subspace of \mathbb{R}^3 . We call this subspace the *eigenspace associated with the eigenvalue -1* and denote it by E_{-1} . An eigenvector is any non-zero element of this space (and thus will form a *basis* for E_{-1}).

We choose any convenient non-zero value for s . Let $s = 1$ and we obtain

eigen value $\lambda = -1$ $\xrightarrow{\mathbf{x}}$ $\mathbf{x}_1 = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$

with

$$E_{-1} = \{s\mathbf{x}_1\} = \text{span}(\mathbf{x}_1).$$

Since all the eigenvectors associated with -1 are multiples of \mathbf{x}_1 , we only have *one independent eigenvector*. The dimension of the eigenspace is called the *geometric multiplicity* of the eigenvalue. In this case, $\lambda_1 = -1$ has algebraic multiplicity 1 and geometric multiplicity 1.

Remark 1.7. Note that MATLAB would choose s so that $\|\mathbf{x}_1\| = 1$.

$$\begin{aligned} (\mathbf{A} - \lambda \mathbf{I})\mathbf{x} &\approx \mathbf{0} \\ (\mathbf{A} - 1 \cdot \mathbf{I})\mathbf{x} &\approx \mathbf{0} \end{aligned}$$

The eigenvectors associated with $\lambda_2 = 1$ satisfy $(\mathbf{A} - \mathbf{I})\mathbf{x} = \mathbf{0}$; that is

$$\left[\begin{array}{ccc|c} -2 & -2 & -2 & 0 \\ 1 & 1 & 1 & 0 \\ -1 & -1 & -1 & 0 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right]. \quad \begin{aligned} &\text{3 two free parameters} \\ &\text{e.g. free variables.} \end{aligned}$$

In this case we have *two* rows of zeros and so there are two free parameters. The solution is

$$\mathbf{x} = \begin{bmatrix} -s - t \\ s \\ t \end{bmatrix} = s \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} + t \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \quad \begin{aligned} x_3 &= t \\ x_2 &= s \\ x_1 &+ s + t = 0 \\ x_1 &= -s - t \end{aligned}$$

(row)
6
(2)

Algebraic & Geometric multiplicity

Definition

If λ appears k times as a repeated root of the characteristic equation of the matrix A , then we say λ is an eigen value of algebraic multiplicity k .

If λ is an eigen value of A then $\dim E_\lambda$ is the geometric multiplicity of λ .

Example

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & -2 & 1 \\ 0 & 0 & 3 \end{bmatrix} \quad \# \text{ upper triangle}$$

$$\textcircled{1} \quad \det(A - \lambda I) = 0$$

$$\begin{vmatrix} 1-\lambda & 1 & 0 \\ 0 & -2-\lambda & 1 \\ 0 & 0 & 3-\lambda \end{vmatrix} = 0$$

$$= (-\lambda)[(-2-\lambda)(3-\lambda) - 0 \cdot 1] - 1(0) + 0$$

$$= (-\lambda)(-\lambda)(3-\lambda) \quad \# \text{ since we have "upper triangle, we can just read from the matrix}$$

$\lambda = 1, -2, 3$

(2)

$$\begin{aligned} (A - \lambda I) \vec{x} &= \vec{0} \\ \lambda = -2 \quad [A + 2I] &= \begin{bmatrix} 3 & 1 & 6 \\ 0 & 0 & 1 \\ 0 & 0 & 5 \end{bmatrix} \\ &= \begin{bmatrix} 3 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

$$E_{-2} = \left\{ \begin{bmatrix} 1 \\ -3 \\ 0 \end{bmatrix} \right\} \text{ will do}$$

This eigen value -2 has:

- algebraic multiplicity "1"
- geometric multiplicity "1"

Example

$$A = \begin{bmatrix} 4 & 1 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

three our eigen values

$$\lambda = 4, 4, 3$$

$$\lambda = 3 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \text{ as a basis for } E_3$$

$$\lambda = 4 \quad (A - 4I) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

choose $\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

• For $\lambda = 3$ algebraic multiplicity = geometric ...
= 1 (both)

• For $\lambda = 4$ algebraic multiplicity = 2
geometric multiplicity = 1

$$\lambda = 1 \quad \begin{array}{l} \text{algebraic multiplicity 2} \\ \text{geometric multiplicity 2} \end{array}$$

The eigenspace E_1 is 2-dimensional. Setting $s = 1, t = 0$ and $s = 0, t = 1$ we obtain two linearly independent eigenvectors

$$\mathbf{x}_2 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{x}_3 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

associated with the eigenvalue 1 with

$$E_1 = \text{span}(\mathbf{x}_2, \mathbf{x}_3).$$

Thus the geometric multiplicity of $\lambda_2 = 1$ is 2. A different choice of free parameters (or a different Gaussian elimination) may yield two different eigenvectors. However these eigenvectors will always be able to be written in terms of $s\mathbf{x}_2 + t\mathbf{x}_3$ with appropriate choices for s and t .

1.5 More on multiplicity

In the above example, we had

$$\text{geometric multiplicity} = \text{algebraic multiplicity}$$

for both eigenvalues (1 in the case of $\lambda_1 = -1$, and 2 in the case of $\lambda_2 = 1$). Is this always the case? Consider

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}. \quad \text{(2)} \quad \begin{array}{l} \text{upper triangular} \\ \text{eigenvalues} \end{array} \quad \begin{array}{l} \text{Read my} \\ \text{straight off} \\ \text{the diagonal} \end{array}$$

The eigenvalue is $\lambda = 1$ (upper triangular matrix) with algebraic multiplicity 2. However

$$[A - I \quad \mathbf{0}] = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

which is already row reduced. We have only one row of zeros and so there is only one free parameter; that is,

$$\mathbf{x} = s \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

is the only independent eigenvector. Therefore the geometric multiplicity of the eigenvalue 1 is 1 while its algebraic multiplicity is 2. A matrix that has an eigenvalue whose

$$\text{geometric multiplicity} < \text{algebraic multiplicity}$$

is called *defective*. Defective matrices must have at least one repeated eigenvalue.

Eigenvectors are linearly independent

Suppose \mathbf{x}_1 and \mathbf{x}_2 are eigenvectors associated with λ_1 and λ_2 respectively. Furthermore, suppose $c_1\mathbf{x}_1 + c_2\mathbf{x}_2 = \mathbf{0}$. Now

$$A(c_1\mathbf{x}_1 + c_2\mathbf{x}_2) = c_1\lambda_1\mathbf{x}_1 + c_2\lambda_2\mathbf{x}_2 = \mathbf{0}.$$

So $c_1\mathbf{x}_1 = -c_2\mathbf{x}_2$, and therefore

$$c_2(\lambda_2 - \lambda_1)\mathbf{x}_2 = \mathbf{0}.$$

Since $\mathbf{x}_2 \neq \mathbf{0}$ (it is an eigenvector), we have $c_2 = 0$ if $\lambda_1 \neq \lambda_2$. In this case c_1 will also be zero and so \mathbf{x}_1 and \mathbf{x}_2 are *linearly independent*. What we have shown is eigenvectors associated with *distinct* eigenvalues are linearly independent.

Eigenvalues of the powers of a matrix

Suppose $A\mathbf{x} = \lambda\mathbf{x}$. Then

$$A^2\mathbf{x} = \lambda A\mathbf{x} = \lambda^2\mathbf{x}.$$

By induction, λ^n is an eigenvalue of A^n with corresponding eigenvector \mathbf{x} for any positive integer n . Moreover, if A is invertible then

$$\mathbf{x} = A^{-1}A\mathbf{x} = \lambda A^{-1}\mathbf{x},$$

so

$$A^{-1}\mathbf{x} = \lambda^{-1}\mathbf{x}.$$

Thus if A is invertible then λ^{-1} is an eigenvalue of A^{-1} with corresponding eigenvector \mathbf{x} (note that, since the determinant of A is the product of the eigenvalues, 0 is an eigenvalue of A if and only if A is singular).

The above arguments also apply to *polynomials* of matrices. Let

$$q(A) = a_k A^k + a_{k-1} A^{k-1} + \cdots + a_1 A + a_0 I$$

be a polynomial (in A) of degree k . We have

$$\begin{aligned} q(A)\mathbf{x} &= a_k A^k \mathbf{x} + a_{k-1} A^{k-1} \mathbf{x} + \cdots + a_1 A \mathbf{x} + a_0 I \mathbf{x} \\ &= a_k \lambda^k \mathbf{x} + a_{k-1} \lambda^{k-1} \mathbf{x} + \cdots + a_1 \lambda \mathbf{x} + a_0 \mathbf{x} \\ &= (a_k \lambda^k + a_{k-1} \lambda^{k-1} + \cdots + a_1 \lambda + a_0) \mathbf{x} \\ &= q(\lambda) \mathbf{x}. \end{aligned}$$

Therefore $q(\lambda)$ is an eigenvalue of $q(A)$ with corresponding eigenvector \mathbf{x} .

Transpose of a matrix

First note that, for any matrix A ,

$$\det(A^T - \lambda I) = \det(A - \lambda I)^T = \det(A - \lambda I).$$

Therefore the characteristic polynomial of A^T is the same as that of A . Thus, the *eigenvalues of A^T are the same as A* , even though the eigenvectors may differ.

Dominant eigenvalue

The *dominant eigenvalue*, λ_1 , of an $n \times n$ matrix A is the eigenvalue of largest absolute value. That is,

$$|\lambda_i| < |\lambda_1|$$

for all $i \neq 1$.

For example, the dominant eigenvalue of a matrix with eigenvalues $-4, -3, 1, 3$ is -4 . However, a matrix with eigenvalues $-4, -3, 1, 4$ has no dominant eigenvalue since the eigenvalues -4 and 4 have the same absolute value.

Note that if λ_1 is the dominant eigenvalue for A then λ_1^k will be the dominant eigenvalue for A^k .

In summary:

- Eigenvalues and Eigenvectors might be complex, even for a matrix that only contains real numbers.
- ✓ • Eigenvectors associated with distinct eigenvalues are linearly independent.
- The algebraic multiplicity of an eigenvalue λ_i is the number of times the factor $(\lambda - \lambda_i)$ is repeated in the characteristic equation.
- The geometric multiplicity of an eigenvalue λ_i is the number of rows of zeros in the row reduction of $[A - \lambda_i I \ 0]$. It gives the dimension of the associated eigenspace; that is, the number of linearly independent eigenvectors associated with λ_i .
- For any eigenvalue of a $n \times n$ matrix

$$1 \leq \text{geometric multiplicity} \leq \text{algebraic multiplicity} \leq n.$$

$n \times n$

Diagonalisation

just stuff

along diagonal of zeros

everywhere else.

- Triangular & diagonal matrices are nice!

- Eigenvalues are in full view.

- can we reduce a given square matrix to a triangular or even a diagonal one such that our eigenvalues are preserved

~~Gaussian Elimination~~

- But doesn't preserve eigenvalues

$$A^{\text{tri}} \tilde{x} = b$$

Diagonalise \rightarrow well behaved with respect to eigen values

- A & B are Similar (if they are $n \times n$)

$$A \sim B$$

If there exists an invertible $n \times n$ P

$$P^{-1}AP = B$$

Similar matrices have some very nice properties:

- $\det(A) = \det(B)$
- A is invertible \Leftrightarrow B is too
- A & B have the same rank.
- A & B have the same characteristic polynomial
- A & B have the same eigenvalues.

Diagonalisation

A $n \times n$ matrix is diagonalisable

if and only if

A has n linearly independent eigen vectors

$$A = PDP^{-1}$$

Eigen vectors as columns
diagonal matrix eigenvalues along value

This is how "close" A is to a diagonal matrix

method to diagonalise

A $n \times n$ matrix

- Find eigenvalues $\{\lambda_k\}$ of A
- Find basis of each eigenspace E_{λ_k}
- If there are enough eigen vectors (you'll need n)

$$A \sim D \quad (\text{Diagonal})$$

$$A = PDP^{-1}$$

Go back to example 1.6

$$A = \begin{bmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{bmatrix}$$

we found eigenvalues $\lambda = -1, 1, 1$
 we found eigenvectors $\lambda = 1$
 $\tilde{x}_1 = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$ $\tilde{x}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ $\tilde{x}_3 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$

$$\therefore P = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad P = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$A = P D P^{-1}$$

$$AP = PD$$

$$\begin{bmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -2 & -1 & -1 \\ 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -2 & -1 & -1 \\ 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

DIY Diagonalisation (Revision from last lecture)

- Find the eigen values $\{\lambda_k\}$ of $n \times n A$
- Find a basis for each eigenspace E_{λ_k}
- If there are "enough" vectors, create a diagonal matrix D of eigenvalues & P with the associated eigenvalues as columns

$A \sim D$

$P^{-1}AP = D$ is a diagonal matrix

or

$$A = PDP^{-1}$$

2 Diagonalisation

Poole Ch. 4.4

MATLAB has the function `eig` to compute eigenvalues/eigenvectors.

- $\mathbf{d} = \text{eig}(\mathbf{A})$ (or, simply, `eig(A)`) will return a *vector* of eigenvalues of A .
- $[\mathbf{P}, \mathbf{D}] = \text{eig}(\mathbf{A})$ will return a *diagonal matrix* D with the eigenvalues of A on the main diagonal and a matrix P whose *columns* are the eigenvectors of A (such that the j^{th} column corresponds to the eigenvalue d_{jj}).

The rationale for this format is the following. Let

$$\mathbf{P} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_n]$$

(that is, the \mathbf{x}_i are the columns of P which are the eigenvectors of A). Note that

$$A\mathbf{x}_j = d_{jj}\mathbf{x}_j$$

since the eigenvalue is stored in the $j j$ position in D . Then

$$AP = A [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_n] = [d_{11}\mathbf{x}_1 \ d_{22}\mathbf{x}_2 \ \cdots \ d_{nn}\mathbf{x}_n] = PD$$

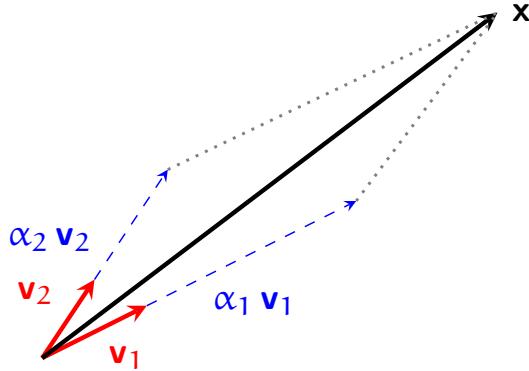
(or, if P is non-singular, $A = PDP^{-1}$).

2.1 Coordinates

Let \mathbf{v}_1 and \mathbf{v}_2 be two linearly independent vectors in \mathbf{R}^2 . Any vector in \mathbf{R}^2 may be written as a *linear combination* of \mathbf{v}_1 and \mathbf{v}_2 :

$$\mathbf{x} = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2.$$

(α_1, α_2) are called the *coordinates of \mathbf{x} relative to the basis $\{\mathbf{v}_1, \mathbf{v}_2\}$* .



In matrix form, this equation is

$$\mathbf{x} = [\mathbf{v}_1 \ \mathbf{v}_2] \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \mathbf{P} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}$$

where $P = [\mathbf{v}_1 \ \mathbf{v}_2]$ is a matrix whose columns are \mathbf{v}_1 and \mathbf{v}_2 . Thus the coordinates of \mathbf{x} may be found by solving this system (that is, row reducing $[P \ | \ \mathbf{x}]$). Formally, we have

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = P^{-1}\mathbf{x}.$$

Now assume that we choose $\mathbf{v}_1 = \mathbf{x}_1$ and $\mathbf{v}_2 = \mathbf{x}_2$ linearly independent eigenvectors of some 2×2 matrix A with associated eigenvalues λ_1 and λ_2 . Multiplication by A is given by

$$A\mathbf{x} = A(\alpha_1\mathbf{x}_1 + \alpha_2\mathbf{x}_2) = \alpha_1\lambda_1\mathbf{x}_1 + \alpha_2\lambda_2\mathbf{x}_2.$$

That is, the effect of multiplying by A is to multiply each coordinate of \mathbf{x} by the corresponding eigenvalue of A .

Thus

$$\begin{aligned} A\mathbf{x} &= [\mathbf{x}_1 \ \mathbf{x}_2] \begin{bmatrix} \lambda_1\alpha_1 \\ \lambda_2\alpha_2 \end{bmatrix} = [\mathbf{x}_1 \ \mathbf{x}_2] \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \\ &= P \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} P^{-1}\mathbf{x} = PDP^{-1}\mathbf{x} \end{aligned}$$

where

$$P = [\mathbf{x}_1 \ \mathbf{x}_2]$$

is the matrix whose columns are the eigenvectors of A and D is the *diagonal* matrix of the eigenvalues

$$D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}.$$

Since \mathbf{x} is *arbitrary*, we have

$$A = PDP^{-1}$$

in case where the 2×2 matrix A has *two* linearly independent eigenvectors.

2.2 Diagonalisation

The above example shows that *some* 2×2 matrices A can be decomposed into a product PDP^{-1} . This process is called the *diagonalisation* of A . The process can be (easily) generalised to $n \times n$ matrices A . If A has n linearly independent eigenvectors, x_i , then

$$A = PDP^{-1}$$

where

$$P = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_n] \quad \text{and} \quad D = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

For any two $n \times n$ matrices A and B , we say that A is *similar* to B if there exists a non-singular matrix P such that

$$B = P^{-1}AP.$$

Diagonalisation Thm

- A is a $n \times n$ square
- Distinct eigenvalues: $\lambda_1, \lambda_2, \dots, \lambda_k$ \leftarrow because some might be repeated

Then the following are equivalent:

- A is diagonalisable
- Union of bases of eigenspaces of A "enough"
contains n vectors
- Algebraic multiplicity of each eigenvalue "enough"
is equal to the geometric multiplicity
Defective

Example

Is A diagonalisable?

$$A = \begin{bmatrix} -3 & 4 \\ -1 & 1 \end{bmatrix}$$

$$\textcircled{1} \det(A - \lambda I) = 0$$

$$\therefore \lambda = -1, 1$$

$$\textcircled{2} \text{ find eigen vectors } (A - \lambda I)\vec{x} = 0$$

$$(A + I)\vec{x} = 0$$

$$(A + I) \rightarrow \begin{bmatrix} -2 & 4 \\ -1 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} -2 & 4 \\ 0 & 0 \end{bmatrix} \leftarrow \text{only going to get 1 eigenvector}$$

so eigenvalue -1 has geometric multiplicity 1

\therefore No not diagonalisable

Example

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 3 & 0 & -3 \\ 1 & 0 & -1 \end{bmatrix}$$

Do this work!

$$(\text{!}) \quad \lambda_1 = \lambda_2 = 0 \quad \text{Algebraic } m = 2$$

$$\lambda_3 = -2$$

yes!

E_0 has geometric m 2

$$E_{-2} \sim \sim \sim \sim \sim$$

If A is similar to B , we write

$$A \sim B.$$

Thus if A has n linearly independent eigenvectors then A is similar to the diagonal matrix D formed from the eigenvalues.

Note that the ordering of the eigenvalues are irrelevant (as long as we are consistent). For example, we also have

$$A = \tilde{P} \tilde{D} \tilde{P}^{-1}$$

where

$$\tilde{P} = [\mathbf{x}_n \ \mathbf{x}_{n-1} \ \cdots \ \mathbf{x}_1] \quad \text{and} \quad A = \begin{bmatrix} \lambda_n & 0 & \cdots & 0 \\ 0 & \lambda_{n-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_1 \end{bmatrix}$$

Earlier, we saw that

$$A = \begin{bmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{bmatrix}$$

had eigenvalues $-1, 1, 1$ with eigenvectors

$$\mathbf{x}_1 = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{x}_3 = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

respectively. Therefore

$$D = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad P = [\mathbf{x}_1 \ \mathbf{x}_2 \ \mathbf{x}_3] = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

and (check) $AP = PD$. (Checking it this way rather than $A = PDP^{-1}$ avoids computing the inverse P^{-1} .)

Question 2.1. So what matrices are diagonalisable?

Clearly, we need n linearly independent eigenvectors (if columns of P are linearly dependent then $\det P = 0$ and so P^{-1} does not exist). This occurs if (and only if) the geometric multiplicity equals the algebraic multiplicity for each eigenvalue.

A matrix A is diagonalisable if and only if it is not defective.

If A has distinct eigenvalues then it is not defective and so can be diagonalised. If it has repeated eigenvalues then it may or may not be diagonalisable.

Note that the diagonal matrix D and the matrix P may be *complex* even when A is *real*. For example, if

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

*Hard !
Appropriate
Example*

$a+i b$
 $a-i b$

*watch:

real matrices can
have a complex
vector (eigen-

then the characteristic polynomial is

$$p(\lambda) = \lambda^2 + 1$$

and so the eigenvalues are $\pm i$ (if A is real then complex eigenvalues must always occur in complex conjugate pairs). The eigenvector associated with i will be given by

$$[A - iI \mid \mathbf{0}] = \left[\begin{array}{cc|c} -i & -1 & 0 \\ 1 & -i & 0 \end{array} \right] \rightarrow \left[\begin{array}{cc|c} -i & -1 & 0 \\ 0 & -0 & 0 \end{array} \right].$$

That is,

$$\mathbf{x}_1 = \begin{bmatrix} i \\ 1 \end{bmatrix}.$$

$$A = PDP^{-1}$$

The eigenvector associated with $-i$ will be the complex conjugate

$$\mathbf{x}_2 = \bar{\mathbf{x}}_1 = \begin{bmatrix} -i \\ 1 \end{bmatrix}$$

(since if A is real then $A\mathbf{x} = \lambda\mathbf{x}$ implies $A\bar{\mathbf{x}} = \bar{\lambda}\bar{\mathbf{x}}$). We have (check!)

$$AP = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} i & -i \\ 1 & 1 \end{bmatrix} = PD = \begin{bmatrix} i & -i \\ 1 & 1 \end{bmatrix} \begin{bmatrix} i & 0 \\ 0 & -i \end{bmatrix}$$

and so

$$A = PDP^{-1}.$$

Supposing A is diagonalisable, $A = PDP^{-1}$, then

$$A^T = (P^{-1})^T D^T P^T = QDQ^{-1}$$

where $Q = P^{-T} \equiv (P^{-1})^T$. Thus A^T is diagonalisable and its eigenvectors are the columns of P^{-T} (remember the columns of P are the eigenvectors of A).

2.3 The Cayley-Hamilton Theorem

Earlier we observed that if λ was an eigenvalue of A then $q(\lambda)$ was an eigenvalue of $q(A)$. If A is diagonalisable, that is $A = PDP^{-1}$, then

$$A^2 = PDP^{-1}PDP^{-1} = PD^2P^{-1}.$$

Following this idea further, for any polynomial q we have

$$q(A) = q(PDP^{-1}) = Pq(D)P^{-1}$$

with

$$q(D) = \begin{bmatrix} q(\lambda_1) & 0 & \cdots & 0 \\ 0 & q(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & q(\lambda_n) \end{bmatrix}$$

Cayley-Hamilton Thrm

Lost my calculator
internet's broke.

Every square matrix satisfies its own characteristic equation!

Demonstration by example:

Let's use a 2×2 case

$$A = \begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix}$$

characteristic $C_A(\lambda)$

$$\det(A - \lambda I) = \begin{vmatrix} 1-\lambda & -1 \\ 2 & 3-\lambda \end{vmatrix}$$

= ..

$$= \lambda^2 - 4\lambda + 5 = 0$$

$A^2 - 4A + 5 =$ matrix full of zeroes

$$\begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix}^2 - 4 \begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix} + 5 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} =$$

$$\begin{bmatrix} -1 & -4 \\ 8 & 7 \end{bmatrix} - \begin{bmatrix} 4 & -4 \\ 8 & 12 \end{bmatrix} + \begin{bmatrix} 5 & 5 \\ 0 & 5 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

! use C-H Thrm to compute powers &
inverses of matrices

2×2 case

characteristic polynomial $C_A(\lambda) = \lambda^2 + a\lambda + b$

Then $A^2 + aA + bI = 0$ ← matrix full of 0's

So $\underbrace{A^2}_{\text{nasty}} = -aA - bI$ $\underbrace{\quad}_{\text{nice}}$

$$A^3 = A \cdot A^2$$

$$= A(-aA - bI)$$

$$= -aA^2 - bAI$$

$$= -a(-aA - bI) - bAI$$

$$= (a^2 - b)A + abI$$

Keep going! Any positive power of A we write as a linear combination of A & I .

Even A^{-1}

$$A^{-1} = -\frac{1}{b}A - \frac{a}{b}I$$

$$A^2 + aA + bI = 0$$

This observation has a remarkable consequence. If we choose q to be the characteristic polynomial p of A then

$$p(D) = \begin{bmatrix} p(\lambda_1) & 0 & \cdots & 0 \\ 0 & p(\lambda_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p(\lambda_n) \end{bmatrix} = O$$

where O is the zero matrix since each eigenvalue is a root of the characteristic polynomial. Therefore

$$p(A) = Pp(D)P^{-1} = P O P^{-1} = O.$$

If A isnt diagonalisable

This result also holds in the defective case (it requires a bit more proof¹) and is known as the *Cayley-Hamilton Theorem*.

Since

$$p(\lambda) = (-1)^n \lambda^n + p_{n-1} \lambda^{n-1} + \cdots + p_1 \lambda + p_0,$$

the Cayley-Hamilton Theorem implies (after some rearranging)

$$A^n = (-1)^{n-1} (p_{n-1} A^{n-1} + \cdots + p_1 A + p_0 I)$$

and, if A is invertible, some more rearranging (and multiplying by A^{-1}) gives

$$A^{-1} = \frac{1}{p_0} ((-1)^{n-1} A^{n-1} - p_{n-1} A^{n-2} - \cdots - p_2 A - p_1 I)$$

(remember that $p_0 = \det A$).

2.4 systems of linear differential equations

Hard! Appreciation

Returning to the example of the system of ODEs $\mathbf{y}' = A\mathbf{y}$, if A is diagonalisable then

and so

$$\mathbf{y}' = PDP^{-1}\mathbf{y}$$

$$P^{-1}\mathbf{y}' = DP^{-1}\mathbf{y}.$$

Since P^{-1} is a matrix of constants, we may rewrite this as

$$\mathbf{z}' = D\mathbf{z} \quad "A\mathbf{z} = \mathbf{b}" \quad - "decouple"$$

where $\mathbf{z} = P^{-1}\mathbf{y}$ (that is, $P\mathbf{z} = \mathbf{y}$). We thus have the system of n equations ODE's

$$\begin{aligned} z'_1 &= \lambda_1 z_1 \\ z'_2 &= \lambda_2 z_2 \\ &\vdots \\ z'_n &= \lambda_n z_n. \end{aligned}$$

• usefulness
here

¹"a bit more" is a slight understatement.

We say that the original system has been *decoupled*. Each of the component equations is independent. Therefore

$$z_i = c_i e^{\lambda_i t}$$

where c_i are constants determined by the initial conditions. It is tempting to write this as

diagonal

$$\mathbf{z} = e^{Dt} \mathbf{c}$$

where

Give some meaning to matrix "

" ↴

$$e^{Dt} = \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix} \quad \text{and} \quad \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}.$$

In terms of the original dependent variables, we have

$$\mathbf{y} = P\mathbf{z} = Pe^{Dt}\mathbf{c}$$

with \mathbf{c} given by initial data $\mathbf{y}(0)$:

$$\mathbf{c} = P^{-1}\mathbf{y}(0).$$

For solution to the system of ODEs

$$\mathbf{y}' = \begin{bmatrix} -1 & -2 & -2 \\ 1 & 2 & 1 \\ -1 & -1 & 0 \end{bmatrix} \mathbf{y}, \quad \mathbf{y}(0) = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

we first compute $\mathbf{c} = P^{-1}\mathbf{y}(0)$. It is much better to solve $P\mathbf{c} = \mathbf{y}(0)$ than to compute P^{-1} ! Therefore

$$[P \mid \mathbf{y}(0)] = \left[\begin{array}{ccc|c} 2 & -1 & -1 & 1 \\ -1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{array} \right] \rightarrow \left[\begin{array}{ccc|c} 2 & -1 & -1 & 1 \\ 0 & 1 & -1 & 1 \\ 0 & 0 & 4 & 0 \end{array} \right]$$

and so

$$\mathbf{c} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

This is our tried & true example

Now

$$e^{Dt} = \begin{bmatrix} e^{-t} & 0 & 0 \\ 0 & e^t & 0 \\ 0 & 0 & e^t \end{bmatrix} \quad \lambda_s = -1, +1, 1$$

and so the solution is

$$\mathbf{y} = Pe^{Dt}\mathbf{c} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} e^{-t} & 0 & 0 \\ 0 & e^t & 0 \\ 0 & 0 & e^t \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 2e^{-t} - e^t \\ e^t - e^{-t} \\ e^{-t} \end{bmatrix}$$

So we were tempted with the “notation” e^{Dt} to denote the matrix

$$\begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix}.$$

We are now justified in making this connection. Recall that the exponential is given by the Taylor series

$$e^t = 1 + t + \frac{1}{2!}t^2 + \frac{1}{3!}t^3 + \cdots$$

and so we *define*

$$\begin{aligned} e^{At} &= 1 + At + \frac{1}{2!}(At)^2 + \frac{1}{3!}(At)^3 + \cdots \\ &= 1 + At + \frac{1}{2!}A^2t^2 + \frac{1}{3!}A^3t^3 + \cdots \end{aligned}$$

The natural question is: how does one evaluate this expression?

If $A = PDP^{-1}$ then $A^k = PD^kP^{-1}$ and so

$$e^{At} = P \left(1 + Dt + \frac{1}{2!}D^2t^2 + \frac{1}{3!}D^3t^3 + \cdots \right) P^{-1}.$$

For a diagonal matrix

$$D^k = \begin{bmatrix} \lambda_1^k & 0 & \cdots & 0 \\ 0 & \lambda_2^k & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n^k \end{bmatrix}$$

and so

$$e^{Dt} = \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix}.$$

Therefore,

$$e^{At} = Pe^{Dt}P^{-1}.$$

Example 2.2. MATLAB has the command `expm` to compute the matrix exponential

(not `exp`). For example, e^A when $A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 1 & 3 \end{bmatrix}$ in MATLAB:

```
>> expm([2 0 0; 0 1 -1; 0 1 3])
```

```
ans =
```

$$\begin{array}{ccc} 7.3891 & 0 & 0 \\ 0 & 0.0000 & -7.3891 \\ 0 & 7.3891 & 14.7781 \end{array}$$

Note that this example is a defective matrix.

Markov chains

- Natural phenomena going through stages

Weather

day-day

Sunny } states
rainy } states

Rugby

win } states
lose } states
draw } states

game-by-game

System

evolves
through

Stages

at any state it
can be in one
of finite number
of states

State

Terms: • transition probabilities

• Transition matrix

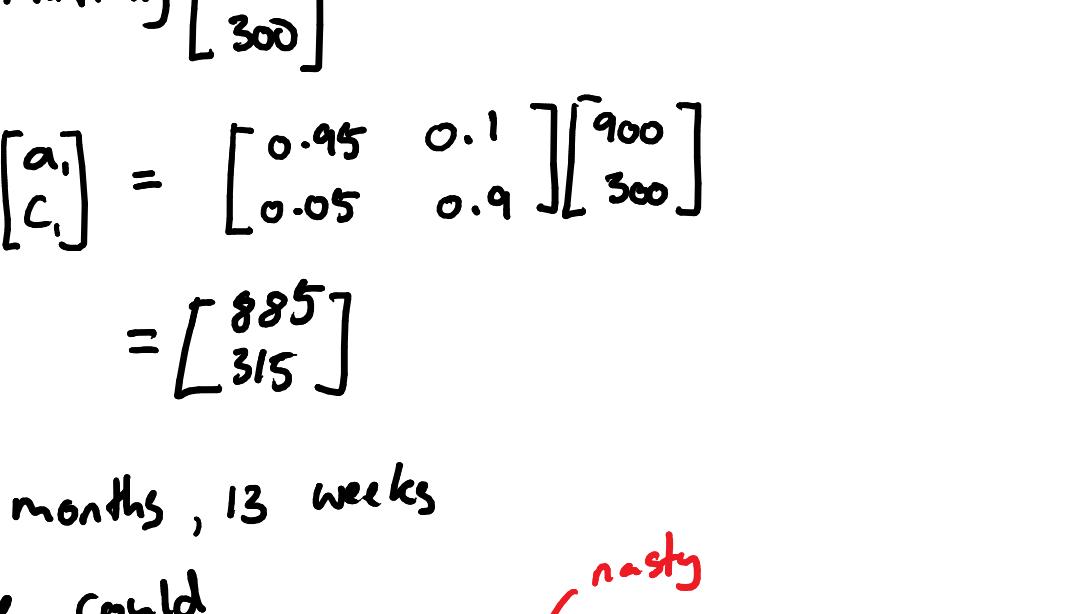
• Stochastic matrix

• State vectors

• Positive matrix

• Regular matrix

Car rental problem



- Turn this into a linear algebra problem

The relationships between the number of vans in each city this week (a_k, c_k) and next week (a_{k+1}, c_{k+1})

$$a_{k+1} = 0.95a_k + 0.10c_k$$

$$c_{k+1} = 0.05a_k + 0.90c_k$$

$$\begin{bmatrix} a_{k+1} \\ c_{k+1} \end{bmatrix} = \begin{bmatrix} 0.95 & 0.10 \\ 0.05 & 0.90 \end{bmatrix} \begin{bmatrix} a_k \\ c_k \end{bmatrix}$$

+ transition matrix of
this Markov process

↑ checks this adds to 1

a) Initially $\begin{bmatrix} 900 \\ 300 \end{bmatrix}$

$$\begin{bmatrix} a_1 \\ c_1 \end{bmatrix} = \begin{bmatrix} 0.95 & 0.1 \\ 0.05 & 0.9 \end{bmatrix} \begin{bmatrix} 900 \\ 300 \end{bmatrix}$$

$$= \begin{bmatrix} 885 \\ 315 \end{bmatrix}$$

b) 3 months, 13 weeks

we could

$$\begin{bmatrix} a_{13} \\ c_{13} \end{bmatrix} = \begin{bmatrix} 0.95 & 0.1 \\ 0.05 & 0.9 \end{bmatrix}^{13} \begin{bmatrix} 900 \\ 300 \end{bmatrix}$$

= 3 years later of hand computation

$$= \begin{bmatrix} 812 \\ 388 \end{bmatrix}$$

OR we could use eigenvalues & eigenvectors to diagonalise!

- Find eigenvalues:

$$(0.95 - \lambda)(0.9 - \lambda) - (0.1)(0.05) = 0$$

$$\dots = 0$$

largest & absolute value $(\lambda - 1)(\lambda - 0.85) = 0$

- It turns out that Markov processes always have $\lambda = 1$ as an eigenvalue. And it is always the biggest (dominant eigenvalue)

For $\lambda = 1$ $A - I = \begin{bmatrix} -0.05 & 0.1 \\ 0.05 & -0.1 \end{bmatrix} \xrightarrow{\text{row}} \begin{bmatrix} 1 & -2 \\ 0 & 0 \end{bmatrix}$

so an eigenvector $v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

- For $\lambda = 0.85$

$$A - 0.85I = \begin{bmatrix} 0.1 & 0.1 \\ 0.05 & 0.05 \end{bmatrix} \xrightarrow{\text{row}} \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$$

so an eigenvector

$$v_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \text{ (say!)}$$

we have "enough" eigenvectors! Diagonalisable ✓

$$D = \begin{bmatrix} 1 & 0 \\ 0 & 0.85 \end{bmatrix} \quad P = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix}$$

$$P^{-1} = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{2}{3} \end{bmatrix}$$

- Long term behaviour

$$x_k = A^k z_0 = (P D P^{-1})^k z_0 \xrightarrow{\text{PDD...DP}^{-1}}$$

$$= P D^k P^{-1} z_0$$

and so

$$= \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & -\frac{2}{3} \end{bmatrix} \begin{bmatrix} 900 \\ 300 \end{bmatrix}$$

$$= \begin{bmatrix} 800 \\ 400 \end{bmatrix}$$

so the system will eventually stabilize with 800

vans in Auckland & 400 vans in Christchurch

Note: OR you can simply solve the equation $A v = v$

(finding the dominant eigenvector, as we "know" the dominant eigenvalue is always $\lambda = 1$)

$$A \tilde{x} = \lambda \tilde{x}$$

3 Markov Chains

Poole Sec. 3.7, 4.6

Problem 3.1. Suppose that a car-hire company has a fleet of campervans based in Auckland and Christchurch, and that experience has shown that each week (on average):

- 5% of vans hired in Auckland each week are returned to the depot in Christchurch, and the rest are returned in Auckland;
- 10% of vans hired in Christchurch each week are returned to the depot in Auckland, and the rest are returned in Christchurch.

If there are 900 vans in Auckland now, and 300 vans in Christchurch, how many vans will be in each city:

initial conditions

1. next week?
2. in three months' time (that is, 13 weeks)?
3. in the long run?

We can represent this situation by a system of linear equations

$$\begin{aligned} a^{(k+1)} &= 0.95a^{(k)} + 0.10c^{(k)} \\ c^{(k+1)} &= 0.05a^{(k)} + 0.90c^{(k)} \end{aligned}$$

where $a^{(k)}$ is the number of vans in Auckland in week k , and $c^{(k)}$ is the number of vans in Christchurch in week k . In matrix form,

$$\mathbf{v}^{(k+1)} = A\mathbf{v}^{(k)}, \quad \mathbf{v}^{(k)} = \begin{bmatrix} a^{(k)} \\ c^{(k)} \end{bmatrix},$$

and A is the matrix

$$A = \begin{bmatrix} 0.95 & 0.10 \\ 0.05 & 0.90 \end{bmatrix}.$$

Clearly the answer to the first question is

$$\mathbf{v}^{(1)} = A\mathbf{v}^{(0)}$$

where

$$\mathbf{v}^{(0)} = \begin{bmatrix} 900 \\ 300 \end{bmatrix}.$$

Similarly, the answer to the second question is

$$\mathbf{v}^{(13)} = A\mathbf{v}^{(12)} = \dots = A^{13}\mathbf{v}^{(0)}.$$

In order to compute large powers of A , we have seen that it is efficient to first compute the eigenvalues and eigenvectors of A . If we assume that $A = PDP^{-1}$ is diagonalizable, then

$$\mathbf{v}^{(13)} = P D^{13} P^{-1} \mathbf{v}^{(0)}.$$

Using MATLAB, we have

```

>> A = [0.95 0.10; 0.05 0.90]
A =
    0.9500    0.1000
    0.0500    0.9000
>> v_0 = [900; 300]
v_0 =
    900
    300
>> [P, D] = eig(A)
P =
    0.8944   -0.7071
    0.4472    0.7071
D =
    1.0000      0
        0    0.8500

```

and so A is diagonalizable with

$$D = \begin{bmatrix} 1.00 & 0 \\ 0 & 0.85 \end{bmatrix}.$$

Thus

$$D^k = \begin{bmatrix} 1.00 & 0 \\ 0 & 0.85^k \end{bmatrix}$$

and since $0 < 0.85 < 1$ we have

$$D^k \rightarrow \begin{bmatrix} 1.00 & 0 \\ 0 & 0 \end{bmatrix}$$

as $k \rightarrow \infty$. Therefore, in answer to the third question, the long-term behaviour is

$$\mathbf{v}^{(\infty)} = P \begin{bmatrix} 1.00 & 0 \\ 0 & 0 \end{bmatrix} P^{-1} \mathbf{v}^{(0)}.$$

In order to compute this, we first solve $P\mathbf{z} = \mathbf{v}^{(0)}$ (that is, find the coordinates of $\mathbf{v}^{(0)}$ with respect to the “eigen-basis”). Thus

```

>> D_inf = [1 0; 0 0];
>> z = P \ v_0
z =
    894.4272
   -141.4214
>> v_inf = P * D_inf * z
z =
    800
    400

```

We see that, in the long run, there will be 800 campervans in Auckland and 400 in Christchurch. Note that the long-term behaviour is determined by the dominant eigenvalue.

This problem is an example of a *Markov chain*. A (finite) Markov chain represents the evolution of a system with a finite number n of states in *discrete* “time” steps. Thus the state of the system is described by a vector of length n , \mathbf{v} . The number of the population in state i is v_i . In the example above, we have two states; the Auckland depot and the Christchurch depot with $v_1^{(k)}$ being the number of campervans returned to the Auckland depot in week k .

For a Markov chain, the evolution must satisfy the following requirements:

- The total population remains fixed (no accidents!).
- The number in each state must be non-negative (we cannot have -100 campervans in Auckland!).
- The state that the system is in at the next step, $\mathbf{v}^{(k+1)}$, depends *only* on the present state, $\mathbf{v}^{(k)}$ (and not its evolutionary history).
- The probability of moving from state i to state j is constant—that is, does not depend on the step k .

The probabilities to move from state i to state j , a_{ij} , are called the *transition probabilities* of the Markov chain. Thus the evolution of a Markov chain is given by

$$\mathbf{v}^{(k+1)} = A\mathbf{v}^{(k)}$$

where $A = [a_{ij}]$ with $0 \leq a_{ij} \leq 1$ and

$$\sum_{i=1}^n a_{ij} = 1$$

(the a_{ij} are probabilities and columns must sum to 1). This is the requirement that the population remains fixed. The matrix A is called the *transition matrix* of the Markov chain. Any square matrix with these properties is called a *stochastic matrix*. The vectors $\mathbf{v}^{(k)}$ are called *state vectors*.

Question 3.2. What is the long-term behaviour of a Markov chain?

The behaviour of the campervan example is no accident.

Theorem 3.3. *Let A be a stochastic matrix. Then 1 is an eigenvalue of A .*

Proof. Recall that the eigenvalues of A^T are the same as A . Let

$$\mathbf{x}^T = [1 \ 1 \ \cdots \ 1].$$

For a stochastic matrix A ,

$$\mathbf{x}^T A = \mathbf{x}^T$$

since the column sums are 1. Therefore,

$$A^T \mathbf{x} = \mathbf{x}$$

and so 1 is an eigenvalue of A^T , and consequently 1 is also an eigenvalue of A . \square

Note that the method employed in this proof implies that A^k will be a stochastic matrix if A is stochastic, since

$$\mathbf{x}^T A^k = \mathbf{x}^T A A^{k-1} = \mathbf{x}^T A^{k-1} = \cdots = \mathbf{x}^T A = \mathbf{x}^T$$

and so the column sums of A^k are all 1. The fact that A^k is stochastic is obvious from the *context* with which it was introduced; but the mathematical fact that this is the case requires a proof (such as the above).

We saw with the above example that the eigenvalue 1 was also the dominant eigenvalue. This eigenvalue then determined the long-term behaviour of the Markov chain. For most Markov chains this is the case. A matrix is called *positive* if all its entries are positive (this is not the same as *positive definite*). Furthermore a square matrix A is called *regular* if A^k is positive for some power k . For example

$$B = \begin{bmatrix} 3 & 1 \\ 2 & 0 \end{bmatrix} \quad \begin{array}{l} (\text{not}) \\ \bullet \text{ a positive matrix} \end{array}$$

is not positive, but it is regular, since

$$B^2 = \begin{bmatrix} 11 & 3 \\ 6 & 2 \end{bmatrix}. \quad \begin{array}{l} \bullet \text{ regular matrix} \end{array}$$

Theorem 3.4. *Let A be a stochastic matrix with eigenvalue λ . Then*

- $|\lambda| \leq 1$
- if A is regular and $\lambda \neq 1$ then $|\lambda| < 1$.

For proof of this theorem, see Poole Theorem 4.31 (pp. 336–7).

Thus, in the regular case at least, 1 is the dominant eigenvalue. The non-regular case may have more than one eigenvalue with $|\lambda| = 1$ and so may not have a dominant eigenvalue. Returning to the regular case, if $A = PDP^{-1}$ is diagonalizable then

$$D^k \rightarrow D^{(\infty)}$$

as $k \rightarrow \infty$ where $D^{(\infty)}$ is a diagonal matrix with only 0 and 1 on its diagonal. In particular,

$$PD^{(\infty)} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_r \ \mathbf{0} \ \cdots \ \mathbf{0}]$$

where r is the algebraic (and geometric) multiplicity of the eigenvalue 1. Therefore the long-term behaviour of the Markov chain

$$\mathbf{v}^{(\infty)} = PD^{(\infty)}\mathbf{z} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_r \ \mathbf{0} \ \cdots \ \mathbf{0}] \mathbf{z} \in E_1$$

where $P\mathbf{z} = \mathbf{v}^{(0)}$. Thus the long-term behaviour of the Markov chain lies in the eigenspace of the dominant eigenvalue 1.

It can be shown (Poole Lemma 4.32) that if A is regular and diagonalizable then $r = 1$; the algebraic multiplicity of the eigenvalue 1 is 1. In this case

$$\mathbf{c}^{(\infty)} = c\mathbf{x}_1.$$

Since the total population

$$N = \|\mathbf{v}^{(0)}\|_1$$

is fixed, we must have $\|\mathbf{v}^{(\infty)}\|_1 = N$. Thus if we choose \mathbf{x}_1 to be the positive unit vector in the 1-norm (that is, a vector whose column sum is 1), then

$$\mathbf{v}^{(\infty)} = N\mathbf{x}_1.$$

Returning to the campervan example, we have $N = 1200$. MATLAB computed the eigenvectors so that their 2-norms are 1 so

$$\mathbf{x}_1 = \begin{bmatrix} 0.8944 \\ 0.4472 \end{bmatrix}.$$

Rescaling this so that its 1-norm is 1 we have

$$\mathbf{x}_1 = \begin{bmatrix} 0.6667 \\ 0.3333 \end{bmatrix}$$

and so

$$\mathbf{v}^{(\infty)} = 1200 \begin{bmatrix} 0.6667 \\ 0.3333 \end{bmatrix} = \begin{bmatrix} 800 \\ 400 \end{bmatrix}.$$

This example shows the *purpose* of the theory! Naively, we would have need to compute all the eigenvalues of A in order to find the dominant eigenvalue. However, if A is regular, the theory shows that we only need to compute the eigenspace E_1 in order to solve the problem, which is computationally much easier.

3.1 The Perron-Frobenius Theorem

The above observations on regular Markov chains are the consequence of a useful theorem called the Perron-Frobenius theorem which dates from 1912. We state a restricted version of it here (a more general statement is given in Poole Theorem 4.37).

Theorem 3.5 (Perron-Frobenius). *Let A be a positive $n \times n$ matrix. Then A has a real eigenvalue λ_1 with the following properties:*

- $\lambda_1 > 0$
- λ_1 has a corresponding positive eigenvector
- λ_1 is the dominant eigenvalue of A
- λ_1 has algebraic multiplicity 1.

(proof omitted)

We can apply this theorem to a regular Markov chain. In this case we know that A^k is positive for some k . Thus the Perron-Frobenius theorem guarantees that A^k has a dominant positive eigenvalue λ_1 and so A has a dominant positive eigenvalue $\lambda_1^{1/k}$ (which is unambiguous since $\lambda_1 > 0$). In the case of a Markov chain, A is stochastic so $\lambda_1 = 1$.

Note that this theorem also shows that we do *not* need the requirement that A be diagonalizable for the dominant eigenvalue to have algebraic multiplicity 1. Thus our arguments for the long-term behaviour of the Markov chain will work for *any* regular Markov chain.

Thrm 3.3

An alternative way to think about it.

- Let A be a stochastic matrix
Then 1 is an eigenvalue of A .

Column sums of A are 1

$$A^T = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix} = 1 \cdot \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}$$

Therefore $\lambda=1$ is an eigenvalue of A^T ,
and hence of A too.

(But the eigenvector will be different).

The steady state vector

A state vector \tilde{x} with the property

$$B\tilde{x} = \tilde{x}$$

is called a steady state vector.

(In fact one can prove that every Markov chain
has a unique steady state vector).

? Can we find this vector without doing any
iterations at all?

$$B\tilde{x} = \tilde{x}$$

$$B\tilde{x} = I\tilde{x}$$

$$(B-I)\tilde{x} = 0$$

This is just a homogeneous system of linear
equations in our case.

$$A = \begin{bmatrix} 0.15 & 0.1 \\ 0.05 & 0.9 \end{bmatrix} \quad [A - I] = \begin{bmatrix} 0.95-1 & 0.1 \\ 0.05 & 0.9-1 \end{bmatrix}$$

$\lambda=1$

Steady state vector $\begin{bmatrix} 800 \\ 400 \end{bmatrix}$

\vdots
 $= \begin{bmatrix} 0.5 & -1 \\ 0 & 0 \end{bmatrix}$ ← row of
zeros
 $x_2 = \text{free}$

Now suppose a steady state vector

$$\tilde{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} z \\ t \end{bmatrix} \text{ say!}$$

We need \tilde{x} to be a probability vector

$$\begin{aligned} 1 &= x_1 + x_2 \\ &= 2t + t \quad t = \frac{1}{3} \\ &= 3t \end{aligned}$$

$$\tilde{x} = \begin{bmatrix} \frac{2}{3} \\ \frac{1}{3} \end{bmatrix}$$

Now we need \tilde{x} to be an actual distribution

(of vars)

Example

(Markov Chain)

- Kim Kardashian has two sisters Khloe & Kourtney.
- She never goes out with Khloe twice in a row, however if she goes out with Kourtney she is 3 times likely to go out with Kourtney the next time as with Khloe. Initially she is equally likely to go out with either sister.

System Kim

Stages Dates outings

States Khloe
Kourtney

	Kh	KO
Kh	0	0.25
KO	1	0.75

columns add up to 1

$$\tilde{S}_m = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \text{ state vector for an outing } m$$

s_1^m outing with Kh

s_2^m outing with KO

$$\tilde{S}_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

$$T = \begin{bmatrix} 0 & 0.25 \\ 1 & 0.75 \end{bmatrix}$$

$$\tilde{S}_1 = T \tilde{S}_0 = \begin{bmatrix} 0 & 0.25 \\ 1 & 0.75 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0.125 \\ 0.875 \end{bmatrix}$$

$$\tilde{S}_2 = T \tilde{S}_1 = \begin{bmatrix} 0 & 0.25 \\ 1 & 0.75 \end{bmatrix} \begin{bmatrix} 0.125 \\ 0.875 \end{bmatrix} = \begin{bmatrix} 0.21875 \\ 0.78125 \end{bmatrix}$$

$\tilde{S}_3, \tilde{S}_4, \tilde{S}_5, \dots$ values $\begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}$ **iterating**

OR

=

Find the eigen space E_1

$$(T - I) \tilde{x} = 0$$

$$(T - I) \tilde{x} = 0$$

$$(T - I) = \begin{bmatrix} -1 & 0.25 \\ 1 & -0.25 \end{bmatrix}$$

:

$$\begin{bmatrix} -1 & 0.25 \\ 0 & 0 \end{bmatrix}$$

s_2 is free

$$\text{choose } \begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

↓

$$I = S_1 + S_2 = 4t + t = 5t$$

$$t = \frac{1}{5} \quad s_1 = \frac{1}{5}$$

$$s_2 = \frac{4}{5}$$

$$\begin{bmatrix} 0.2 \\ 0.8 \end{bmatrix}$$

3.2 Leslie models

Poole Sec. 3.7 & 4.6

In 1945, Leslie proposed the following model to describe age-specific population growth.

The females in a population are divided into *age classes*. If the life span of the species is l years and we use n age classes then each age class represents $\frac{l}{n}$ years. We begin with an initial age distribution

$$\mathbf{x}^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$$

where $x_i^{(0)}$ is the number of females in age class i (that is, the number of females aged between $\frac{(i-1)l}{n}$ and $\frac{il}{n}$).

As time progresses, the number of females in each age class changes because of three biological processes:

- births;
- deaths;
- aging.

language extinction

The problem is to predict the age distribution at some future time.

We divide into discrete times t_0, t_1, \dots . The Leslie model requires the duration between successive observations to be the same as the duration of the age intervals, $\frac{l}{n}$. thus

$$t_0 = 0, \quad t_1 = \frac{l}{n}, \quad t_2 = \frac{2l}{n}, \quad \dots \quad t_k = \frac{kl}{n}.$$

The birth and death processes between two successive observation times can be described by

birth rate b_i , $i = 1, 2, \dots, n$	average number of daughters born to each female while she is in age class i ;
survival rate s_i , $i = 1, 2, \dots, n - 1$	fraction of females in age class i that survive and pass into age class $i + 1$.

Let $\mathbf{x}^{(k)}$ be the population distribution at time t_k . $x_1^{(k)}$, the number of females whose age is between 0 and $\frac{l}{n}$, will be the number of females born in the previous season. Therefore

$$x_1^{(k)} = b_1 x_1^{(k-1)} + b_2 x_2^{(k-1)} + \dots + b_n x_n^{(k-1)}$$

since the females in age class i at time t_{k-1} will produce $b_i x_i^{(k-1)}$ daughters.

The number of females in age class $i + 1$ will be the number that survived from age class i in the previous season. Thus

$$x_{i+1}^{(k)} = s_i x_i^{(k-1)} \quad i = 1, 2, \dots, n - 1.$$

Therefore

$$\mathbf{x}^{(k)} = L\mathbf{x}^{(k-1)}$$

where

$$L = \begin{bmatrix} b_1 & b_2 & b_3 & \cdots & b_{n-1} & b_n \\ s_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & s_1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & s_{n-1} & 0 \end{bmatrix}$$

L is called a *Leslie matrix*. The population distribution after k seasons will be

$$\mathbf{x}^{(k)} = L\mathbf{x}^{(k-1)} = L^2\mathbf{x}^{(k-2)} = \cdots = L^k\mathbf{x}^{(0)}.$$

Again we see that we are interested in the behaviour of powers of a matrix.

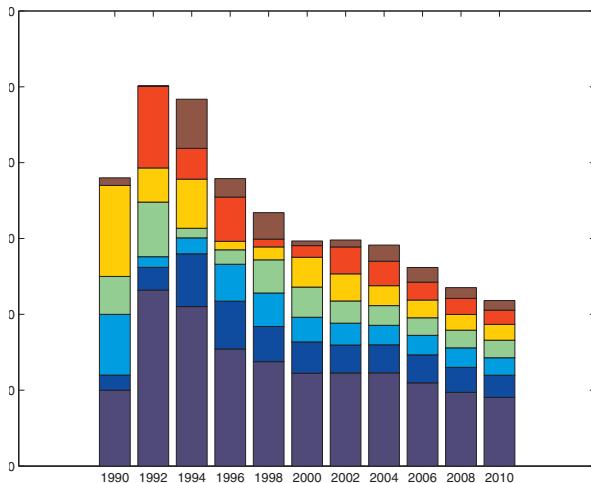
Exercise 3.6. Woodland caribou are primarily found in the western provinces of Canada and the American northwest. The average lifespan of a female is 14 years. The population in the Jasper National Park in Alberta was surveyed in 1990. The birth, survival rates and the age distribution found in the 1990 survey are:

Age (years)	Birth Rate	Survival Rate	Number (1990 survey)
0–2	0.0	0.3	10
2–4	0.2	0.7	2
4–6	0.9	0.9	8
6–8	0.9	0.9	5
8–10	0.9	0.9	12
10–12	0.8	0.6	0
12–14	0.3		1

The Leslie matrix and initial age distribution are

$$L = \begin{bmatrix} 0.0 & 0.2 & 0.9 & 0.9 & 0.9 & 0.8 & 0.3 \\ 0.3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.6 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{x}^{(0)} = \begin{bmatrix} 10 \\ 2 \\ 8 \\ 5 \\ 12 \\ 0 \\ 1 \end{bmatrix}$$

If one directly computes the age distribution out to 2010 say, one obtains

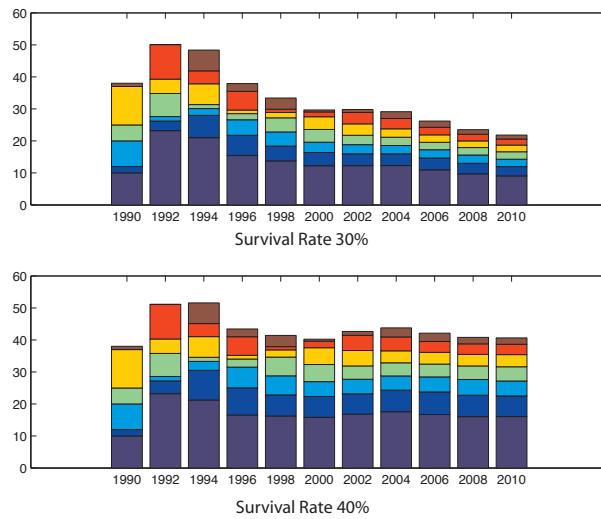


The typical problem that we might want to address is: what is the future of this population? That is, we wish to compute

$$\lim_{k \rightarrow \infty} \mathbf{x}^{(k)}.$$

From a conservation point of view, we would want to know if this population is in danger of becoming extinct. If so, we would then want to know what measures would enhance the survival of the population. If, on the other hand, we were trying to manage a pest then we would want the population to decline and what factors would most influence this decline.

It appears that the woodland caribou at Jasper National Park are in decline. What can be done to reverse this decline? Birth rates are more or less fixed by the biology of the animal (though ensuring an adequate food supply helps). What can be changed is the survival rates. In this case there is high mortality among young female caribou. Only 30% of the female population survive the first two years. What would happen if this survival rate could be increased to 40%? Repeating the above computations with $s_1 = 0.40$ we obtain



It appears that the decline has been arrested. But is it? It could be that the

decline may just be delayed. Increasing the survival rate of young caribou to 40% is likely to be a costly exercise. We need to have a better way of seeing the long term trends so that we may justify this cost.

It would be nice if we could apply the Perron-Frobenius theorem to this model. However L is clearly not positive (it has zero entries) and it is not clear if L^k is positive for some k . In fact, if *all* the birth rates are zero, then L is lower triangular with zeros on the main diagonal and so 0 is the only eigenvalue and has algebraic multiplicity n . Of course, from the context of the problem, at least one of the birth rates must be non-zero. Moreover all the survival rates must also be positive. Unfortunately, even under these conditions, L is not necessarily regular. A more general statement of the Perron-Frobenius theorem states that the theorem applies to irreducible matrices. However L is not guaranteed to be irreducible. So we must look for another approach.

Descartes' Rule of Signs

There is a useful result, called Descartes' Rule of Signs, about the number of positive roots of a polynomial. The rule states that

if the terms of a polynomial with real coefficients are ordered by descending variable exponent, then the number of positive roots of the polynomial is either equal to the number of sign differences between consecutive non-zero coefficients, or is less than it by a multiple of 2.

Multiple roots of the same value are counted separately.

Consider, for example, the polynomial

$$x^3 + x^2 - x - 1.$$

We see that this polynomial has one change of sign (the successive signs are ++, +- and --). Thus we can conclude that the polynomial has one positive root. For the polynomial

$$-x^3 + x^2 + x - 1$$

we have two changes of sign. Thus this polynomial has either two positive roots or no (2 - 2) positive roots.

Expanding the determinant along the last column and using induction, the characteristic polynomial of a Leslie matrix is

$$p(\lambda) = \det(L - \lambda I) = (-1)^n (\lambda^n - b_1 \lambda^{n-1} - b_2 s_1 \lambda^{n-2} - \cdots - b_n s_1 s_2 \dots s_{n-1}).$$

Since at least one of the birth rates must be positive (and the survival rates are all positive), there is only one change of sign in $p(\lambda)$. Therefore L has a unique positive eigenvalue, λ_1 . Direct calculation shows that

$$\mathbf{x}_1 = \begin{bmatrix} \lambda_1^{n-1} \\ s_1 \lambda_1^{n-2} \\ s_1 s_2 \lambda_1^{n-3} \\ \vdots \\ s_1 s_2 \dots s_{n-1} \end{bmatrix}$$

is a positive eigenvector associated with λ_1 .

In fact, it can be shown that if two *consecutive* birth rates are non-zero then λ_1 is the dominant eigenvalue of L . So, if $\lambda_1 < 1$ then the population is doomed to extinction without intervention. Conversely, if $\lambda_1 > 1$ then the population is expanding.

Harvesting a population

A harvesting policy in which an animal population is periodically harvested is said to be *sustainable* if the yield of each harvest is the same and the age distribution of the population remaining after each harvest is the same. Thus, the animal population is not depleted by a sustainable harvesting policy; only the excess growth is removed.

Let \mathbf{x} be the age distribution at the beginning of a season (growth period). $L\mathbf{x}$ will be the age distribution at the end of the growth season. We will assume that this is when harvesting occurs. Suppose that h_i is the fraction of females from age class i that is harvested. Clearly we require $0 \leq h_i \leq 1$ for each i . Let

$$H = \begin{bmatrix} h_1 & 0 & \cdots & 0 \\ 0 & h_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_n \end{bmatrix}.$$

The number of females harvested will be $HL\mathbf{x}$ and so the population after harvesting will be

$$L\mathbf{x} - HL\mathbf{x} = (I - H)L\mathbf{x}.$$

This will also be the age distribution at the beginning of the next growth period and so, for a sustainable harvesting population, we require

$$(I - H)L\mathbf{x} = \mathbf{x}.$$

Let

$$L_h = (I - H)L = \begin{bmatrix} (1 - h_1)b_1 & (1 - h_1)b_2 & \cdots & (1 - h_1)b_n \\ (1 - h_2)s_1 & 0 & \cdots & 0 \\ 0 & (1 - h_3)s_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}$$

The matrix L_h is also a Leslie matrix with *effective* birth rates $(1 - h_1)b_i$ and *effective* survival rates $(1 - h_2)s_1, (1 - h_3)s_2, \dots, (1 - h_n)s_{n-1}$.

There are many different strategies that one might consider. With some populations, it is difficult to catch animals of a specific age. For example, in harvesting a fishery, one might reasonably assume that the fish are caught at random and thus the same fraction of each age class is harvested. Therefore we set

$$h = h_1 = h_2 = \cdots = h_n.$$

This case is called *uniform harvesting*. The condition for a sustainable harvest is then

$$L\mathbf{x} = \left(\frac{1}{1 - h} \right) \mathbf{x}$$

and so

$$\lambda_1 = \frac{1}{1-h}$$

where λ_1 is the dominant eigenvalue of L . Thus

$$h = \frac{\lambda_1 - 1}{\lambda_1}.$$

Clearly if $\lambda_1 < 1$ then it is impossible to sustainable harvest the population.

In some populations only the youngest females are of any economic value and so we harvest only from the youngest age class. Thus we set

$$h_1 = h, \quad h_2 = h_3 = \cdots = h_n = 0.$$

For a maximal harvest that is sustainable, we require $\lambda_1 = 1$. The characteristic polynomial of L_h is

$$(-1)^n \left(\lambda^n - (1-h) (b_1 \lambda^{n-1} + b_2 s_1 \lambda^{n-2} + \cdots + b_n s_1 s_2 \dots s_{n-1}) \right).$$

Therefore $\lambda_1 = 1$ if and only if

$$1 - (1-h) (b_1 + b_2 s_1 + \cdots + b_n s_1 s_2 \dots s_{n-1}) = 0;$$

that is,

$$h = \frac{R-1}{R}$$

where

$$R = b_1 + b_2 s_1 + \cdots + b_n s_1 s_2 \dots s_{n-1}$$

R is called the *net reproduction rate*.

For a certain species of domestic sheep in New Zealand with a growth period of one year, the following birth and survival rates were found.²

Age	Birth Rate	Survival Rate
0–1	0.000	0.845
1–2	0.045	0.975
2–3	0.391	0.965
3–4	0.472	0.950
4–5	0.484	0.926
5–6	0.546	0.895
6–7	0.543	0.850
7–8	0.502	0.786
8–9	0.468	0.691
9–10	0.459	0.561
10–11	0.433	0.370
11–12	0.421	

²G. Caughley, Parameters for Seasonally Breeding Populations, *Ecology*, vol 48, 1967, 834–839

Using MATLAB, we find

$$\lambda_1 = 1.176.$$

Thus, with a uniform harvesting policy, the harvesting fraction is

$$h = \frac{\lambda_1 - 1}{\lambda_1} = 0.149$$

and so we can sustainably harvest 14.9% of the population.

On a sheep farm, it is possible to selectively harvest. Lambs have the greatest economic value (go to a supermarket and compare prices of lamb, hogget and mutton!). Thus there is an economic incentive to harvest only the youngest age group (i.e. the lambs). The net reproduction rate for this population is

$$R = 2.514$$

and so, if we only harvest the lambs, then

$$h = \frac{R - 1}{R} = 0.602$$

Thus we can sustainably harvest 60.2% of the lambs if we do not harvest any other age group. This seems almost too good to be true! But, remember that we are only harvesting the lambs here and not the whole population and so the figures 14.9% and 60.2% are not directly comparable. We need to compute the proportion of lambs in the entire population. Since $\lambda_1 = 1$,

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ s_1 \\ s_1 s_2 \\ \vdots \\ s_1 s_2 \dots s_{n-1} \end{bmatrix}$$

is an eigenvector associated with 1. Moreover it gives the long-term age distribution of the population at the beginning of a season and $L\mathbf{x}_1$ will give the age distribution *immediately* before harvest. Thus the fraction of sheep that are lambs before harvest will be

$$\frac{R}{\|L\mathbf{x}_1\|_1} = 0.2950.$$

Therefore 29.5% of the population are lambs immediately before harvest, and so the harvest is 60.2% of 29.5%, or 17.8% of the total population.

This is the figure that should be compared to the 14.9% from the uniform harvesting policy. We see that we get at least two benefits from harvesting only lambs. We increase the harvest from 14.9% to 17.8% of the total population and we harvest the economically more valuable part of the population.

The power method

Dominant eigen values/vectors \rightarrow long term behaviour

long term behaviour \rightarrow dominant eigenvalues/vectors

$$① \tilde{x}, A\tilde{x}, A^2\tilde{x}, \dots, A^k\tilde{x}, \dots$$

generally tends to a vector parallel to the eigenvector belonging to dominant eigenvalue of A

$$② (\text{shifting})$$

$$Av = \lambda v \Leftrightarrow (A - \alpha I)v = (\lambda - \alpha)v$$

i.e., by shifting λ by a suitable amount we can create a new dominant eigenvalue.

$$③ \text{ If } A \text{ is invertible}$$

$$Av = \lambda v \Leftrightarrow A^{-1}v = \left(\frac{1}{\lambda}\right)v$$

Now the dominant eigenvalue of A^{-1}
 = e-value of A^{-1} furthest from 0
 = (e-value of A closest to 0)⁻¹

so we can find another eigenvalue

Example

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 8 \end{bmatrix}$$

Start with the usual $x = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$

Theoretically $y = A^{-1}x$ is the first step.

But finding A^{-1} is almost never a good idea.

In this case it is better to use Gaussian elimination to find y as the solution to

since we are going to this repeatedly

$A = LU$ would be the way to go by hand.

$$y = y / \text{norm}(y)$$

Ans

So after a few iterations $\lambda = -7.3598$ as true

e-value of A^{-1}

so smallest e-value of A is $\frac{1}{\lambda} = -0.1359$

Idea 4 Find (somehow) an approximate e-value $\tilde{\lambda}$ of A .

Shifting: $B = A - \tilde{\lambda}I$

gives a matrix with an e-value close to 0.

Shifting back gives a (much) improved value for $\tilde{\lambda}$.

Now we need a way to find approximate eigenvalues

quickly

\rightarrow onto Gershgorin's

4 The Power & Inverse Power Methods Poole Ch. 4.5

The only method that we have, at the moment, to compute eigenvalues is to solve the characteristic equation. In practice, this approach is only feasible for small matrices. The power method is an *iterative* method which can be applied to a matrix *that has a dominant eigenvalue*.

4.1 The Power Method

Suppose A is diagonalizable (this assumption is not essential but simplifies the argument) and has a dominant eigenvalue λ_1 . Since A is diagonalizable then the eigenvectors \mathbf{x}_i will form a basis and so, for any \mathbf{y} , we have

$$\mathbf{y} = c_1 \mathbf{x}_1 + c_2 \mathbf{x}_2 + \cdots + c_n \mathbf{x}_n$$

for constants c_i . Now

$$\begin{aligned} A^k \mathbf{y} &= c_1 \lambda_1^k \mathbf{x}_1 + c_2 \lambda_2^k \mathbf{x}_2 + \cdots + c_n \lambda_n^k \mathbf{x}_n \\ &= \lambda_1^k \left(c_1 \mathbf{x}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k \mathbf{x}_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{x}_n \right) \end{aligned}$$

since the dominant eigenvalue λ_1 cannot be 0.

However, since λ_1 is the dominant eigenvalue of A ,

$$\left| \frac{\lambda_i}{\lambda_1} \right| < 1$$

for $i = 2, 3, \dots, n$. Therefore

$$\left(\frac{\lambda_i}{\lambda_1} \right) \rightarrow 0$$

as $k \rightarrow \infty$. Thus

$$A^k \mathbf{y} \rightarrow \lambda_1^k c_1 \mathbf{x}_1$$

as $k \rightarrow \infty$; that is, A^k approaches a non-zero multiple of the eigenvector \mathbf{x}_1 provided $c_1 \neq 0$. This suggests that we can approximate an eigenvector associated with λ_1 by repeatedly multiplying *almost any* vector \mathbf{y} by A .

Example 4.1. Let

$$A = \begin{bmatrix} -2 & -3 \\ -3 & -2 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Now

$$A\mathbf{y} = \begin{bmatrix} -2 \\ -3 \end{bmatrix}, \quad A^2\mathbf{y} = A \begin{bmatrix} -2 \\ -3 \end{bmatrix} = \begin{bmatrix} 13 \\ 12 \end{bmatrix}, \quad \dots \quad A^5\mathbf{y} = \begin{bmatrix} -1562 \\ -1563 \end{bmatrix}, \quad \dots$$

We see that $A^k\mathbf{y}$ is becoming parallel to the vector

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

(remember that any non-zero multiple of an eigenvector is also an eigenvector). Finally

$$A \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -5 \\ -5 \end{bmatrix} = -5 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

and so $\lambda_1 = -5$.

Remark 4.2. 1. In practice we may have a rough idea of the eigenvector \mathbf{x}_1 and so can use this as our initial estimate \mathbf{y} . In any case, we need to choose \mathbf{y} such that $c_1 \neq 0$; that is

$$\mathbf{y} \notin \text{span}(\mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n).$$

However, even if we are unlucky and choose \mathbf{y} such that $c_1 = 0$ then round-off error will normally introduce some non-zero component of \mathbf{x}_1 to our initial guess. In this case round-off error is useful!

2. Repeated multiplication of \mathbf{y} by A may produce some very large numbers ($|\lambda_1| > 1$) or some very small numbers ($|\lambda_1| < 1$). This will eventually cause overflow or underflow problems with finite precision arithmetic. In order to avoid this problem, we must scale our vector after each iteration.

There are many possible scaling strategies that one could choose. For example, we could choose to make each iterate a *unit* vector; that is, define

$$\mathbf{y}^{(1)} = \frac{A\mathbf{y}}{\|A\mathbf{y}\|}, \quad \mathbf{y}^{(2)} = \frac{A\mathbf{y}^{(1)}}{\|A\mathbf{y}^{(1)}\|}, \quad \mathbf{y}^{(3)} = \frac{A\mathbf{y}^{(2)}}{\|A\mathbf{y}^{(2)}\|}, \quad \dots$$

We choose a norm that is easy to compute. The ∞ -norm

$$\|\mathbf{y}\|_\infty = \max \{|y_1|, |y_2|, \dots, |y_n|\}$$

fits that bill nicely. Thus the $k + 1$ -th iterate will be

$$\mathbf{y}^{(k+1)} = \frac{A\mathbf{y}^{(k)}}{\|A\mathbf{y}^{(k)}\|_\infty}.$$

Repeating our earlier example, we see

$$A\mathbf{y} = \begin{bmatrix} -2 \\ -3 \end{bmatrix}, \quad \|A\mathbf{y}\|_\infty = 3$$

and so

$$\mathbf{y}^{(1)} = \frac{A\mathbf{y}}{\|A\mathbf{y}\|_\infty} = \begin{bmatrix} -\frac{2}{3} \\ -1 \end{bmatrix}.$$

The next iterate will be

$$A\mathbf{y}^{(1)} = \begin{bmatrix} 4.3333 \\ 4 \end{bmatrix}, \quad \|A\mathbf{y}^{(1)}\|_\infty = 4.3333.$$

That is,

$$\mathbf{y}^{(2)} = \begin{bmatrix} 1 \\ 0.9231 \end{bmatrix}.$$

Continuing, we obtain

$$\mathbf{y}^{(3)} = \begin{bmatrix} -0.9841 \\ -1 \end{bmatrix}, \quad \mathbf{y}^{(4)} = \begin{bmatrix} 1 \\ 0.9968 \end{bmatrix}, \quad \mathbf{y}^{(5)} = \begin{bmatrix} -0.9994 \\ -1 \end{bmatrix}.$$

Since a non-zero multiple of an eigenvector is also an eigenvector, $\mathbf{y}^{(k)}$ will converge to an eigenvector of λ_1 as $k \rightarrow \infty$. In order to compute the eigenvalue, note that if \mathbf{x} is an eigenvector associated with λ_1 then

$$\frac{(A\mathbf{x}) \cdot \mathbf{x}}{\mathbf{x} \cdot \mathbf{x}} = \frac{(\lambda\mathbf{x}) \cdot \mathbf{x}}{\mathbf{x} \cdot \mathbf{x}} = \frac{\lambda_1(\mathbf{x} \cdot \mathbf{x})}{\mathbf{x} \cdot \mathbf{x}} = \lambda_1.$$

Definition 4.3. Given a (Hermitian) matrix A and a vector \mathbf{y} , the *Rayleigh quotient* is

$$R(\mathbf{y}) \equiv \frac{(A\mathbf{y}) \cdot \mathbf{y}}{\mathbf{y} \cdot \mathbf{y}}$$

Note that

$$R(\mathbf{y}^{(k)}) \rightarrow \lambda_1$$

As $k \rightarrow \infty$. In the above example, we obtain

$$\begin{aligned} R(\mathbf{y}^{(1)}) &= -4.7692 & R(\mathbf{y}^{(2)}) &= -4.9902 \\ R(\mathbf{y}^{(3)}) &= -4.9996 & R(\mathbf{y}^{(4)}) &= -5.0000 \end{aligned}$$

This describes the so-called (scaled) *power method* to compute the dominant eigenvalue (and eigenvector) of a matrix A . The rate of convergence of this method is, at worst,

$$\left| \frac{\lambda_2}{\lambda_1} \right|$$

where λ_2 is the eigenvalue of second largest absolute value.

Note that had we chosen to normalise the iterates by using the 2-norm then

$$\mathbf{y}^{(k)} \cdot \mathbf{y}^{(k)} = \|\mathbf{y}^{(k)}\|_2^2 = 1$$

and so

$$R(\mathbf{y}^{(k)}) = (A\mathbf{y}^{(k)}) \cdot \mathbf{y}^{(k)}.$$

As with any iterative method, we need a stopping condition. If we choose a condition based on the Rayleigh quotients such as

$$|R(\mathbf{y}^{(k+1)}) - R(\mathbf{y}^{(k)})| < \text{tolerance}$$

then it is more efficient to scale the iterates using the 2-norm.

What about the other eigenvalues of A ? First note that if λ is an eigenvalue of A then $\lambda - q$ will be an eigenvalue of $A - qI$. Thus if λ_1 is the dominant eigenvalue of A then

$$0, \lambda_2 - \lambda_1, \lambda_3 - \lambda_1, \dots, \lambda_n - \lambda_1$$

will be the eigenvalues of $B = A - \lambda_1 I$. If this matrix has a dominant eigenvalue, $\lambda_2 - \lambda_1$ say, then we can find it by applying the power method to B and therefore find a second eigenvalue λ_2 of A .

In the above example, we found $\lambda_1 \approx -5$ and so

$$B = A - \lambda_1 I = \begin{bmatrix} 3 & -3 \\ -3 & 3 \end{bmatrix}.$$

Therefore

$$\mathbf{y}^{(1)} = \frac{B\mathbf{y}}{\|B\mathbf{y}\|_\infty} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

and

$$\mathbf{y}^{(2)} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

There is no need to compute any more iterates as they will all be the same. Therefore the eigenvector is

$$\mathbf{x}_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

and

$$R(\mathbf{y}^{(1)}) = 6.$$

We therefore conclude that

$$\lambda_2 \approx 6 + \lambda_1 = 1.$$

We could continue with this approach and now apply the power method to $A - \lambda_2 I$. However the dominant eigenvalue (if it exists) of this matrix may be $\lambda_1 - \lambda_2$ in which case we would not obtain a new eigenvalue of A . Fortunately there is a better approach.

4.2 The Inverse Power Method

Recall that if A is invertible then λ^{-1} will be an eigenvalue of A^{-1} if λ is an eigenvalue of A . If A^{-1} has a dominant eigenvalue μ_1 then it will correspond to the eigenvalue of *smallest* magnitude $\lambda_n = \mu_1^{-1}$ of A . Therefore we can find the least eigenvalue of A by applying the power method to A^{-1} .

However, iterating $A^{-1}\mathbf{y}$ is *not* efficient. Instead we solve the linear system

$$A\mathbf{z} = \mathbf{y}^{(k)}$$

and then scale

$$\mathbf{y}^{(k+1)} = \frac{\mathbf{z}}{\|\mathbf{z}\|}.$$

Since the coefficient matrix for the linear system, A , is unchanged in each iteration, we do not need to do a row reduction for each iteration (and this is the most time consuming step of Gauss elimination). Therefore we first perform a *LU* decomposition of A :

$$A = LU.$$

In each iteration we do a *forward substitution*

$$L\mathbf{w} = \mathbf{y}^{(k)}$$

and a *back substitution*

$$U\mathbf{z} = \mathbf{w}$$

in order to compute the unscaled iterate \mathbf{z} . In terms of the flop count, the inverse power method requires

- $\mathcal{O}(\frac{2}{3}n^3)$ flops to compute the *LU* decomposition of A . This only needs to be done once.
- $\mathcal{O}(2n^2)$ flops *per iteration* for the forward and back substitutions.

By contrast, the power method requires $\mathcal{O}(2n^2)$ flops per iteration (multiplying a matrix by a vector) with no “initial set up costs”. The extra work required for the inverse power method is the initial *LU* decomposition. However the *convergence* of the inverse power method is faster.

Example 4.4. Applying the inverse power method to our example, we obtain (using MATLAB)

$$L = \begin{bmatrix} 0.6667 & 1.0000 \\ 1.0000 & 0 \end{bmatrix}, \quad U = \begin{bmatrix} -3.0000 & -2.0000 \\ 0 & -1.6667 \end{bmatrix}$$

(partial pivoting has caused a row swap in L). Our initial guess is $\mathbf{y}^{(0)} = [0 \ 1]^T$ (which, by pure coincidence, is also the solution to $L\mathbf{w} = \mathbf{y}^{(0)}$). The iterates are

Iteration	\mathbf{w}	\mathbf{z}	$R(\mathbf{y}^{(k-1)})$	$\mathbf{y}^{(k)}$
1	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0.4000 \\ -0.6000 \end{bmatrix}$		$\begin{bmatrix} 0.6667 \\ -1.0000 \end{bmatrix}$
2	$\begin{bmatrix} -1.0000 \\ 1.3333 \end{bmatrix}$	$\begin{bmatrix} 0.8667 \\ -0.8000 \end{bmatrix}$	0.9538	$\begin{bmatrix} 1.0000 \\ -0.9231 \end{bmatrix}$
3	$\begin{bmatrix} -0.9231 \\ 1.6154 \end{bmatrix}$	$\begin{bmatrix} 0.9538 \\ -0.9692 \end{bmatrix}$	0.9981	$\begin{bmatrix} 0.9841 \\ -1.0000 \end{bmatrix}$
4	$\begin{bmatrix} -1.0000 \\ 1.6508 \end{bmatrix}$	$\begin{bmatrix} 0.9937 \\ -0.9905 \end{bmatrix}$	0.9999	$\begin{bmatrix} 1.0000 \\ -0.9968 \end{bmatrix}$

We conclude that the least eigenvalue of A is

$$\lambda_2 \approx \frac{1}{0.9999} = 1.0001.$$

The idea of shifting is much more powerful with the inverse power method. Note that if q is *not* an eigenvalue of A then $A - qI$ *must* be invertible. The eigenvalues of $(A - qI)^{-1}$ will be

$$\frac{1}{\lambda - q}$$

where λ is an eigenvalue of A . If we choose q *close* to λ then $1/(\lambda - q)$ will be the *dominant* eigenvalue of $(A - qI)^{-1}$. Moreover if we choose q *very close* to λ then the magnitude of $1/(\lambda - q)$ will be much larger than that of the next eigenvalue and so the convergence will be very fast. However, in order to exploit these observations, we need a method to estimate the location of the eigenvalues.

insight knowledge + approximation of eigenvalues \rightarrow speed of which things converge depends on the initial vector

centered on the diagonal

5 Gershgorin's (Disc) Theorem \rightarrow "inside knowledge" of where to start looking!

Let A be an $n \times n$ matrix with real or complex entries and let

How big the disk should be

"radius" of a circle

$$r_i = \sum_{j \neq i} |a_{ij}|$$

(not on the diagonal)

for $i = 1, 2, \dots, n$. That is, r_i is the sum of the absolute values of the off-diagonal elements in the i^{th} row of A . The closed disc (in the complex plane)

$$\xrightarrow{\text{Disk}} D_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i\}$$

is called a *Gershgorin disc*. Similarly, we define Gershgorin discs based on the columns of A . Let

$$s_i = \sum_{j \neq i} |a_{ji}|.$$

Center are the elements on the diagonals!

The closed discs

$$C_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq s_i\}$$

are also called *Gershgorin discs*. Gershgorin discs are easy to compute. Their usefulness comes from the following remarkable result.

Theorem 5.1 (Gershgorin's Theorem, 1931). *Let A be a $n \times n$ (real or complex) matrix. Then every eigenvalue of A lies in at least one row-based Gerschgorin disc D_i .*

Proof. Let λ be an eigenvalue of A with corresponding eigenvector \mathbf{x} . Let x_i be the entry in \mathbf{x} with largest absolute value (which will necessarily be non-zero). Since $A\mathbf{x} = \lambda\mathbf{x}$, we have

$$\begin{bmatrix} a_{i1} & a_{i2} & \cdots & a_{in} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \lambda x_i.$$

Thus

$$\sum_{j=1}^n a_{ij}x_j = \lambda x_i$$

or

$$(\lambda - a_{ii})x_i = \sum_{j \neq i} a_{ij}x_j.$$

Note that $x_i \neq 0$ and so

$$|\lambda - a_{ii}| = \frac{\left| \sum_{j \neq i} a_{ij}x_j \right|}{|x_i|} \leq \frac{\sum_{j \neq i} |a_{ij}x_j|}{|x_i|} = \frac{\sum_{j \neq i} |a_{ij}| |x_j|}{|x_i|} \leq \sum_{j \neq i} |a_{ij}| = r_i$$

since $|x_j| \leq |x_i|$. Therefore $\lambda \in D_i$. \square

Example

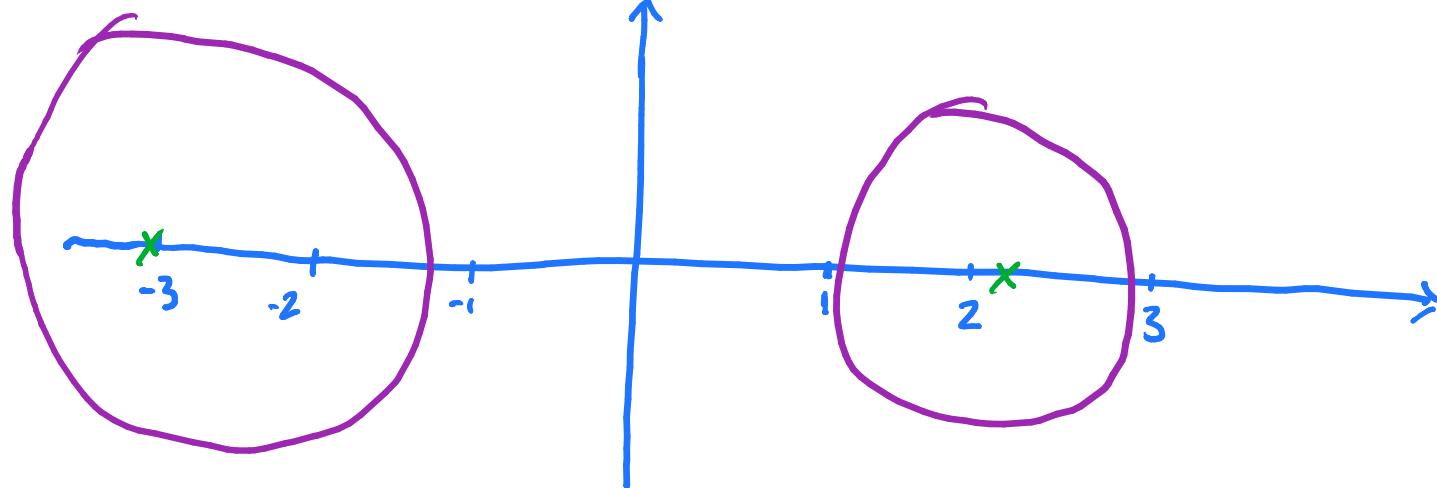
$$A = \begin{bmatrix} 2 & 1 \\ 2 & -3 \end{bmatrix}$$

centers radii
↑ ↑
radii centers

Gershgorin disks:

centers 2 -3

Radii 1 2



My eigenvalues will be in each of these disks.

Actually: Characteristic polynomial = ... = $\lambda^2 + \lambda - 8$

so $\lambda \approx 2.37, -3.37$

Corollary 5.2. Each eigenvalue of A lies in at least one column-based Gershgorin disc C_k .

Proof. Since the eigenvalues of A^T are the same as those of A , apply Gershgorin's Theorem to A^T . \square

Example 5.3. Let

$$A = \begin{bmatrix} 1 & 0.8 \\ -0.4 & 2 \end{bmatrix}. \quad \text{\# complex eigen values}$$

The row-based Gershgorin discs are

$$D_1 : |z - 1| \leq 0.8$$

$$D_2 : |z - 2| \leq 0.4$$

Here we are doing by rows

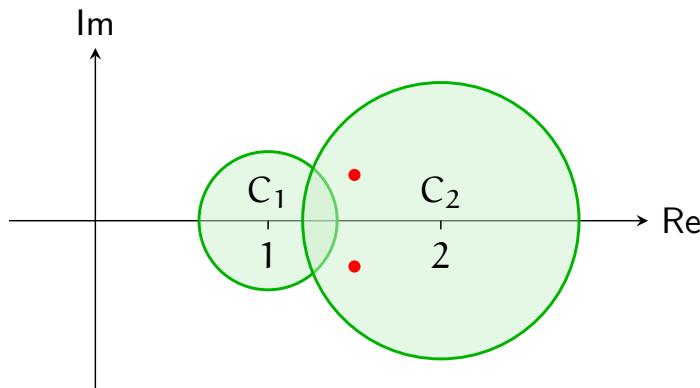
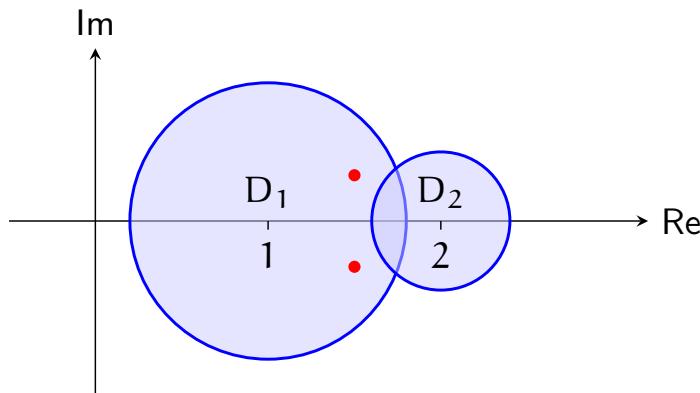
and the column-based Gershgorin discs are

$$C_1 : |z - 1| \leq 0.4$$

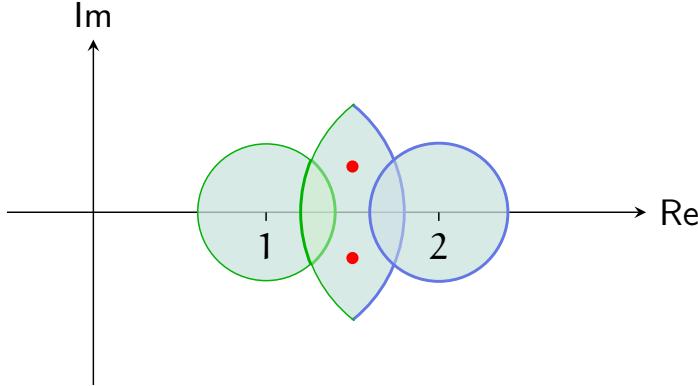
$$C_2 : |z - 2| \leq 0.8$$

we can also look at the columns

Therefore the eigenvalues of A must lie in both $D_1 \cup D_2$ and $C_1 \cup C_2$ (the red points indicate the actual eigenvalues of A).



Gershgorin's Theorem combined with the corollary guarantees that the eigenvalues will be located in the region $(D_1 \cup D_2) \cap (C_1 \cup C_2)$



There is an extension of Gershgorin's Theorem.

Theorem 5.4. *If the union of k row-based discs is disjoint from the union of the remaining $n - k$ row-based discs then the former contains exactly k eigenvalues and the latter contains the remaining $n - k$ eigenvalues.*

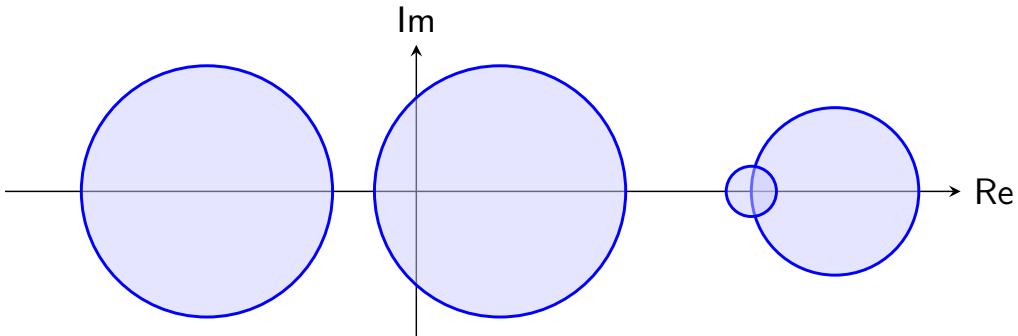
Clearly, by applying this result to A^T , there is a similar result for column-based discs. In particular, if one of the Gershgorin discs is disjoint from all others then it must contain exactly one eigenvalue. The above example shows that if the Gershgorin discs overlap then a given disc may not contain any eigenvalues (in this example, both eigenvalues lie in D_1 and C_2 with no eigenvalues in D_2 or C_1).

In the case where A is a *real* matrix, complex eigenvalues must occur in conjugate pairs. Therefore they must lie in the *same* Gershgorin disc. Thus for A to have complex eigenvalues, it must have an overlapping pair of Gershgorin discs. Consequently the eigenvalue that occurs in a disc that is disjoint from all other discs *must be real*.

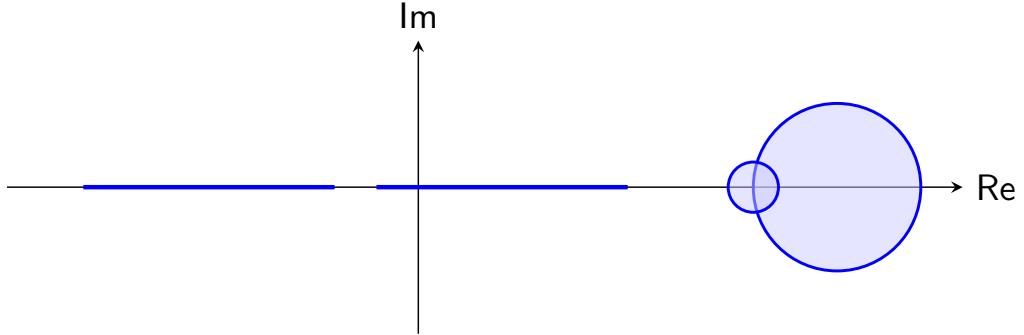
Example 5.5. Let

$$A = \begin{bmatrix} 10 & -1 & 0 & 1 \\ 0.2 & 8 & 0.2 & 0.2 \\ 1 & 1 & 2 & 1 \\ -1 & -1 & -1 & -5 \end{bmatrix}$$

The row-based discs are D_1 centered at 10 with radius 2, D_2 centered at 8 with radius 0.6, D_3 centered at 2 with radius 3 and D_4 centered at -5 with radius 3.

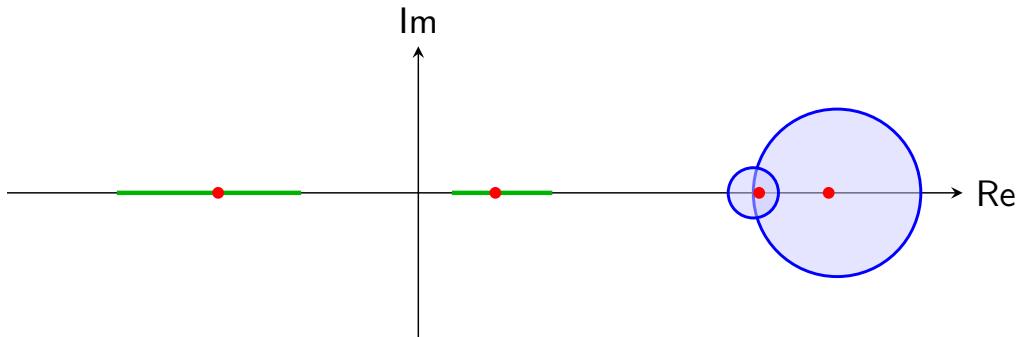


Since both D_3 and D_4 are disjoint from the other discs, they each must contain exactly one eigenvalue. Moreover, since A is real, these eigenvalues must be real.



The column-based Gershgorin discs will be C_1 centered at 10 with radius 2.2, C_2 centered at 8 with radius 3, C_3 centered at 2 with radius 1.2 and C_4 centered at -5 with radius 2.2. Since the radii of C_3 and C_4 are smaller than the radii of D_3 and D_4 , this will reduce the range that these real eigenvalues can have. The radii of C_1 and C_2 are larger than D_1 and D_2 and so these discs give us no new information.

The actual eigenvalues are indicated by the red points:



In this case all the eigenvalues are real. Note that we can also conclude from the Gershgorin diagram that this matrix is invertible since 0 is not contained in any column-based disc.

Numerical Methods for Computing Eigenvalues

Gershgorin's Theorem implies that if the off-diagonal entries of a square matrix A have small norms, the eigenvalues of A are “near” the diagonal elements. The extreme case is that of a diagonal matrix (and so the off-diagonal elements have zero norm) in which case the diagonal elements are the eigenvalues. This observation gives a strategy to design algorithms that compute eigenvalues; we try to find a matrix similar to the given matrix (and therefore has the same eigenvalues) whose off-diagonal elements are small. There are various methods based on this strategy (Householder, Givens and QR).

Example

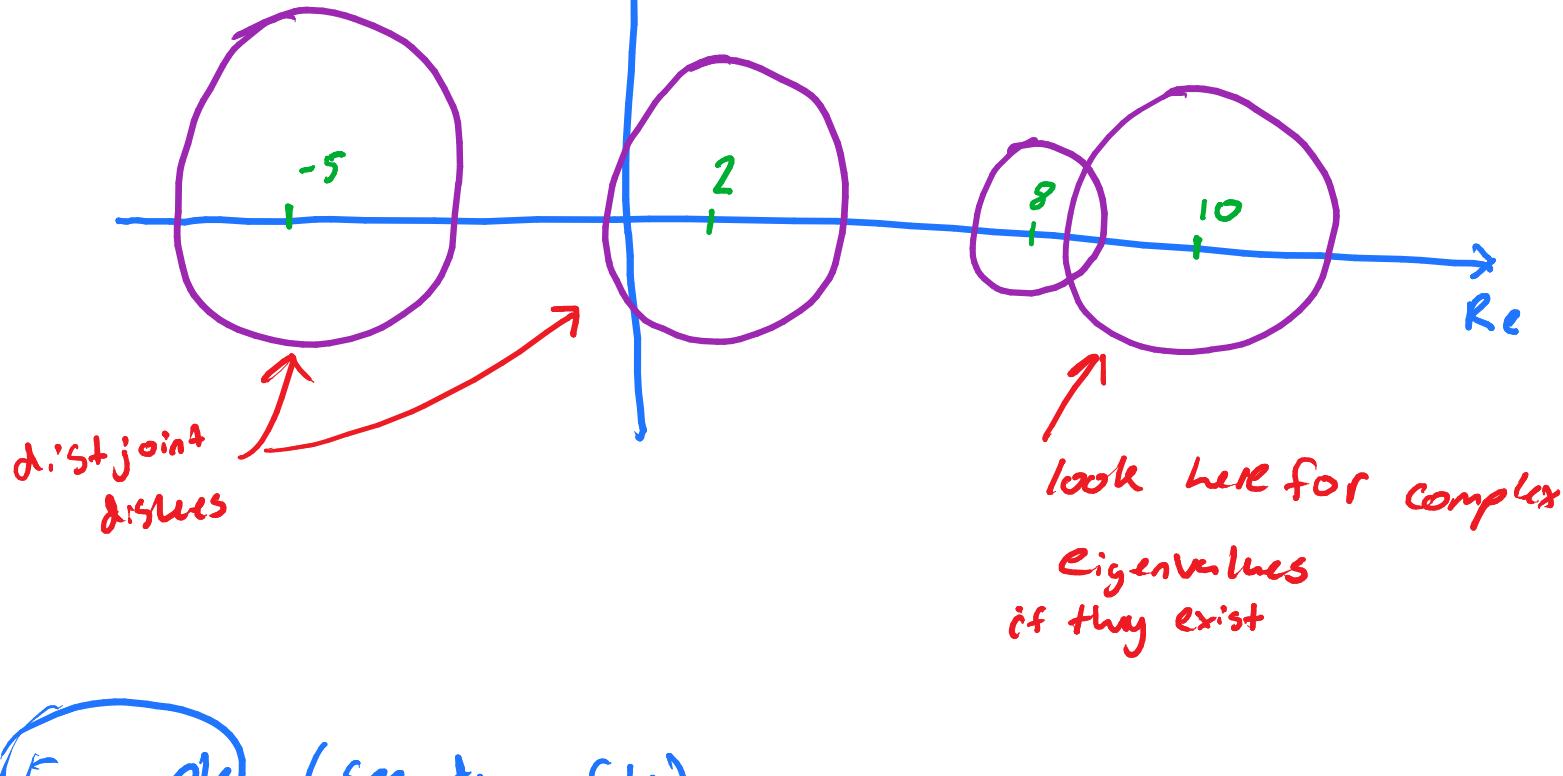
$$A = \begin{bmatrix} 10 & -1 & 0 & 1 \\ 0.2 & 8 & 0.2 & 0.2 \\ 1 & 1 & 2 & 1 \\ -1 & -1 & -1 & -5 \end{bmatrix}$$

Disc 1: Center 10 radius $= | -1 | + 0 + 1 = 2$

Disc 2: Center 8 radius $= 0.2 + 0.2 + 0.2 = 0.6$

Disc 3: Center 2 radius $= 1 + 1 + 1 = 3$

Disc 4: center -5 radius $= | 1 | + | (-1) | + | (-1) | = 3$



Example (see diary file)

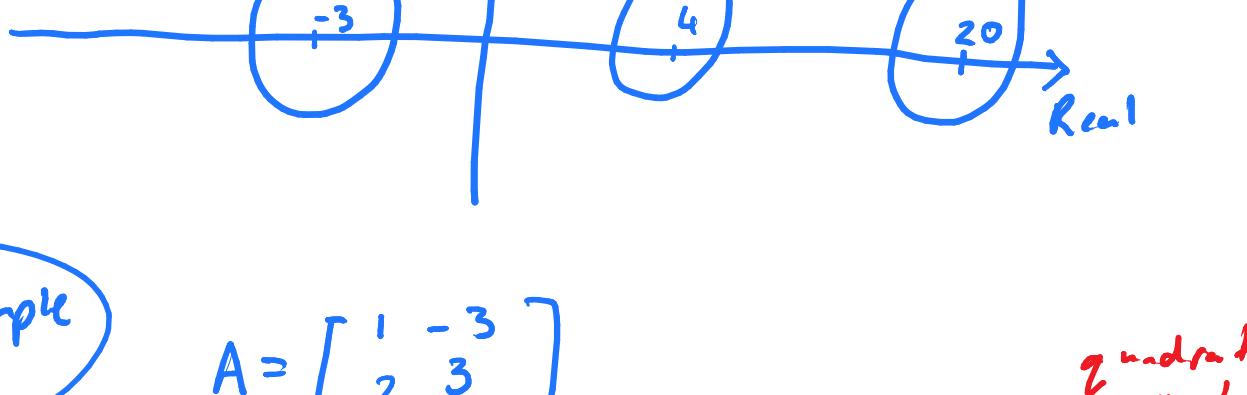
$$A = \begin{bmatrix} 20 & 1 & 0 \\ 1 & -3 & -1 \\ 0 & -1 & 4 \end{bmatrix}$$

The "closer" the matrix is to being diagonal the better our approximations

Matlab $\text{eig}(A) = -3.1824, 4.1389, 20.0435$

Gershgorin's disks

Disk 1: centers	20	radii $= 1 + 0 = 1$
Disk 2:	-3	$= 1 + -1 = 2$
Disk 3:	4	$= 0 + -1 = 1$



Example

$$A = \begin{bmatrix} 1 & -3 \\ 2 & 3 \end{bmatrix}$$

quadratic equation

Real matrix characteristic equation $= \lambda^2 - 4\lambda + 9$

Disk 1 center 1

$$\lambda = 2 + 2.23i$$

Disk 2 center 3

$$\lambda = 2 - 2.23i$$

$$\text{radius} = | -3 | = 3$$

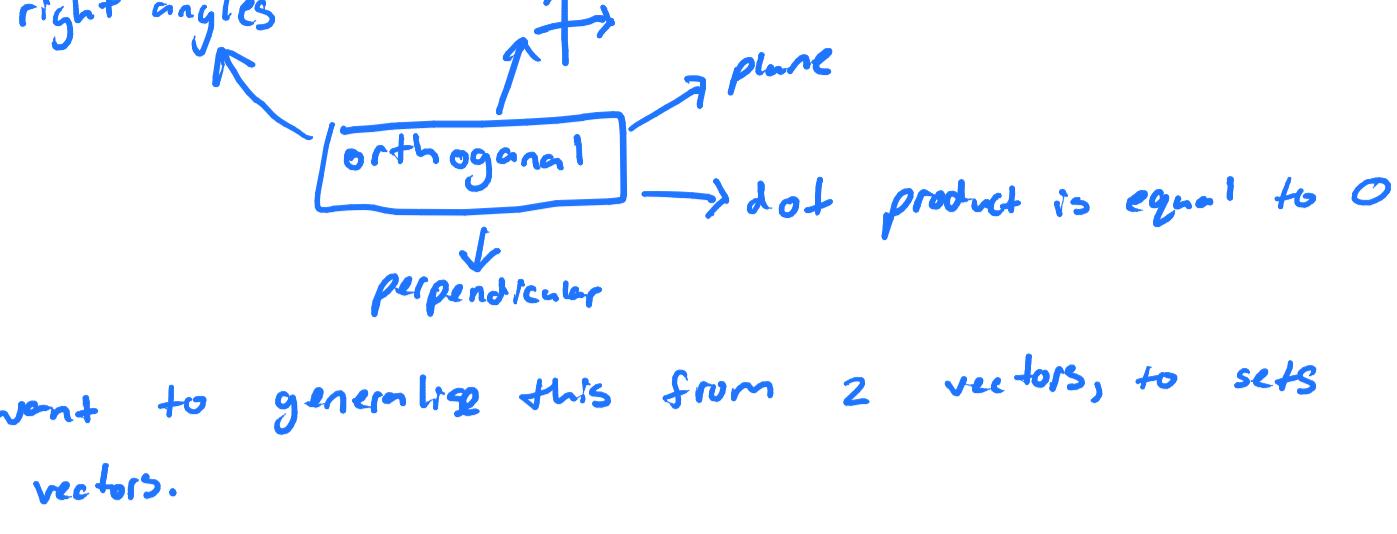
$$= | 2 | = 2$$

• overlapping disks
complex eigenvalues



Orthogonality in \mathbb{R}^n

(of finding nice bases!)



We want to generalize this from 2 vectors, to sets of vectors.

- Two properties:
 - Any 2 distinct vectors in the set are orthogonal
 - each vector in the set is a unit vector
(length = 1)

Glossary

- orthogonal
- orthonormal

Motivation

If we have a basis $\{\tilde{v}_1, \dots, \tilde{v}_k\}$ for a subspace W of \mathbb{R}^n , then we know every vector \tilde{v} in W can be expressed as a linear combination of $\tilde{v}_1, \dots, \tilde{v}_k$

$$\tilde{v} = c_1 \tilde{v}_1 + c_2 \tilde{v}_2 + \dots + c_k \tilde{v}_k$$

But it can be very messy to find two coefficients c_1, \dots, c_k

But the problem becomes much easier if the basis vectors are all orthogonal to one another.

Definition

We say the set $\{\tilde{v}_1, \dots, \tilde{v}_k\}$ is orthogonal if every pair of (different!) vectors in the set are orthogonal

$$\tilde{v}_i \cdot \tilde{v}_j = 0 \quad (i \neq j)$$

Example

① $\{\hat{i}, \hat{j}, \hat{k}\}$ is an orthogonal set in \mathbb{R}^3

$$② \tilde{v}_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}; \tilde{v}_2 = \begin{bmatrix} 2 \\ -1 \end{bmatrix}; \tilde{v}_3 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

is another orthogonal set in

\mathbb{R}^3

Check!

$$\tilde{v}_1 \cdot \tilde{v}_2 = 0 = \tilde{v}_2 \cdot \tilde{v}_3 = \tilde{v}_1 \cdot \tilde{v}_3$$

$\tilde{v}^T \tilde{v}$

$(\quad)(\quad)$

$$③ \tilde{v}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tilde{v}_2 = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \tilde{v}_3 = \begin{bmatrix} 1 \\ 4 \end{bmatrix}$$

Do not form an orthogonal set

c's (coefficients)

will be hard to

find

Orthogonal vectors have to be linearly independent!

$$c_1 \tilde{v}_1 + \dots + c_k \tilde{v}_k = 0$$

c's are all (0)

Take the dot product of both sides with \tilde{v}_1

$$c_1 \tilde{v}_1 \cdot \tilde{v}_1 + c_2 \tilde{v}_2 \cdot \tilde{v}_1 + c_3 \tilde{v}_3 \cdot \tilde{v}_1 + \dots + c_k \tilde{v}_k \cdot \tilde{v}_1 = 0$$

$$c_1 \tilde{v}_1 \cdot \tilde{v}_1 = 0$$

$$c_1 = 0$$

and similarly taking the dot products

with $\tilde{v}_2, \tilde{v}_3, \dots, \tilde{v}_k$, we get $c_2 = \dots = c_k = 0$

Orthogonal sets are "nice" basis. How else are they nice?

Orthogonality in \mathbb{R}^n

- Orthogonal bases are nice bases!

- Today we will see a few more reasons why?

• Burning question - Can we convert any bases into an orthogonal one?

(spoiler alert: Yes! And it is thanks to two Germans,

(Gram & Schmidt)

Orthogonal sets are candidates for nice bases.

Result Let $\{\underline{v}_1, \dots, \underline{v}_k\}$ be an orthogonal basis for a subspace W in \mathbb{R}^n & let \underline{w} be any vector in W .

then the scalars c_1, \dots, c_k such that

$$\underline{w} = c_1 \underline{v}_1 + c_2 \underline{v}_2 + \dots + c_k \underline{v}_k$$

are given by

$$c_i = \frac{\underline{v}_i \cdot \underline{w}}{\underline{v}_i \cdot \underline{v}_i} = \frac{\underline{v}_i^T \underline{w}}{\underline{v}_i^T \underline{v}_i}$$

Proof

$$\begin{aligned} \underline{v}_i \cdot \underline{w} &= \underline{v}_i \cdot (c_1 \underline{v}_1 + \dots + c_k \underline{v}_k) \\ &= c_1 \underline{v}_i \cdot \underline{v}_1 + c_2 \underline{v}_i \cdot \underline{v}_2 + \dots + c_k \underline{v}_i \cdot \underline{v}_k \end{aligned}$$

↑ because orthogonality

$$= c_i \underline{v}_i \cdot \underline{v}_i$$

$$\Rightarrow c_i = \frac{\underline{v}_i \cdot \underline{w}}{\underline{v}_i \cdot \underline{v}_i}$$

Example

Express $\tilde{w} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ as a linear combination of

$$\tilde{v}_1 = \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}, \quad \tilde{v}_2 = \begin{bmatrix} 2 \\ -1 \\ -2 \end{bmatrix} \text{ & } \tilde{v}_3 = \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix}$$

$$\tilde{w} = c_1 \tilde{v}_1 + c_2 \tilde{v}_2 + c_3 \tilde{v}_3$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = c_1 \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} + c_2 \begin{bmatrix} 2 \\ -1 \\ -2 \end{bmatrix} + c_3 \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix}$$

or

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 & 2 & 1 \\ 2 & -1 & -2 \\ 1 & -2 & 2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \leftarrow ?$$

$$c_1 = \frac{\tilde{v}_1 \cdot \tilde{w}}{\tilde{v}_1 \cdot \tilde{v}_1} = \frac{\begin{bmatrix} 2 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}}{\begin{bmatrix} 2 & 2 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix}} = \dots = \frac{2}{9}$$

$$c_2 = \frac{\tilde{v}_2 \cdot \tilde{w}}{\tilde{v}_2 \cdot \tilde{v}_2} = \dots = \frac{2}{9}$$

$$c_3 = \frac{\tilde{v}_3 \cdot \tilde{w}}{\tilde{v}_3 \cdot \tilde{v}_3} = \dots = \frac{1}{9}$$

→ And even nicer if each $\tilde{v}_i \cdot \tilde{v}_i = 1$
i.e., every vector has unit length!

Enter the **orthonormal basis** — make sure all our orthogonal vectors have length 1.

Examples of orthonormal bases

① $\{\tilde{i}, \tilde{j}, \tilde{k}\}$ form an orthonormal basis in \mathbb{R}^3

② "Normalise" our vectors $\tilde{v}_1, \tilde{v}_2, \tilde{v}_3$ above!

$$\begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} = \|\tilde{v}_1\| \text{ is } = \sqrt{2^2 + 2^2 + 1^2} = \sqrt{9} \text{ length } = 3$$

$$\begin{bmatrix} 2 \\ -1 \\ -2 \end{bmatrix}, \quad q_1 = \begin{bmatrix} \frac{2}{3} \\ \frac{2}{3} \\ \frac{1}{3} \end{bmatrix}, \quad q_2 = \begin{bmatrix} \frac{2}{3} \\ -\frac{1}{3} \\ -\frac{2}{3} \end{bmatrix}, \quad q_3 = \begin{bmatrix} \frac{1}{3} \\ -\frac{2}{3} \\ \frac{2}{3} \end{bmatrix}$$

Orthogonal matrix

unfortunate choice of a name.

Technically "Orthonormal matrix"

Suppose we have n orthonormal vectors in \mathbb{R}^n .

Let

$$Q = [q_1 \mid q_2 \mid \dots \mid q_n]$$

A very important property of orthogonal matrices is

$$Q^T Q = I$$

This is because

$$q_i \cdot q_j = q_i^T q_j = 0 \quad \forall i \neq j$$

since all vectors are orthogonal to one another

Any

$$q_i \cdot q_i = q_i^T q_i = 1$$

because they are orthonormal.

$$\begin{bmatrix} q_1^T \\ q_2^T \\ \vdots \\ q_n^T \end{bmatrix} \begin{bmatrix} q_1 & q_2 & \dots & q_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & & & \\ 0 & & \ddots & & \\ 0 & & & 1 & \\ \vdots & & & & \vdots \end{bmatrix}$$

$$Q^T Q = I$$

6 Orthogonality & Basis

Poole Ch. 5

The standard basis for \mathbf{R}^n has some very nice properties. Among these properties are the following:

Definition 6.1. A set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ in \mathbf{R}^n is called an *orthogonal set* if

$$\mathbf{v}_i \cdot \mathbf{v}_j \equiv \mathbf{v}_i^T \mathbf{v}_j = 0 \quad \text{whenever} \quad i \neq j.$$

If, in addition,

$$\mathbf{v}_i \cdot \mathbf{v}_i = \mathbf{v}_i^T \mathbf{v}_i = 1 \quad \text{for} \quad i = 1, 2, \dots, k$$

then the set is called an *orthonormal set*.

Clearly the standard basis for \mathbf{R}^n is an orthonormal set. Orthogonal (and therefore orthonormal) sets are *always* linearly independent.

Theorem 6.2. If $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ is an orthogonal set of non-zero vectors then these vectors are linearly independent.

Proof. Suppose $c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_k \mathbf{v}_k = \mathbf{0}$. Then

$$\mathbf{v}_i^T (c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2 + \dots + c_k \mathbf{v}_k) = \mathbf{v}_i^T \mathbf{0} = 0.$$

Thus

$$0 = c_1 \mathbf{v}_i^T \mathbf{v}_1 + c_2 \mathbf{v}_i^T \mathbf{v}_2 + \dots + c_k \mathbf{v}_i^T \mathbf{v}_k = c_i \mathbf{v}_i^T \mathbf{v}_i$$

since $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ is an orthogonal set. Since \mathbf{v}_i is a non-zero vector, we have $c_i = 0$ and so the set is linearly independent. \square

Definition 6.3. An *orthogonal (orthonormal) basis* for a subspace W of \mathbf{R}^n is a basis of W that is an orthogonal (orthonormal) set.

Since linear independence is guaranteed, any set of n non-zero orthogonal vectors will form a basis for \mathbf{R}^n .

Example 6.4. 1. Show that the vectors

$$\mathbf{v}_1 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} \quad \mathbf{v}_2 = \begin{bmatrix} 2 \\ 2 \\ -1 \end{bmatrix} \quad \mathbf{v}_3 = \begin{bmatrix} -1 \\ -1 \\ 4 \end{bmatrix}$$

form an orthogonal basis for \mathbf{R}^3 .

2. Find an orthogonal basis for the subspace W of \mathbf{R}^3 given by

$$W = \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} : x + y - z = 0 \right\}.$$

The usefulness of orthogonal basis comes from the following observation. Suppose we wish to solve $A\mathbf{x} = \mathbf{b}$. Now

$$A\mathbf{x} = [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_n] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \cdots + x_n\mathbf{a}_n = \mathbf{b}$$

where \mathbf{a}_i are the columns of A . Normally, in order to solve $A\mathbf{x} = \mathbf{b}$ would require a row reduction. However, if $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$ form an orthogonal basis then

$$\mathbf{a}_i \cdot (x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \cdots + x_n\mathbf{a}_n) = x_i\mathbf{a}_i \cdot \mathbf{a}_i$$

and so

$$x_i = \frac{\mathbf{a}_i \cdot \mathbf{b}}{\mathbf{a}_i \cdot \mathbf{a}_i} = \frac{\mathbf{a}_i^T \mathbf{b}}{\mathbf{a}_i^T \mathbf{a}_i}.$$

This observation is extremely useful! For an arbitrary basis $\mathcal{B} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$, we need to solve the system

$$[\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_k \mid \mathbf{w}]$$

in order to express \mathbf{w} as a linear combination of the basis vectors (equivalently, to find the co-ordinates of \mathbf{w} with respect to the basis). However if \mathcal{B} is an orthogonal basis, all we need to do is to compute the dot products to find the coordinate vector of \mathbf{w} with respect to the basis.

Let

$$D = \text{diag} \left(\left[\frac{1}{\mathbf{v}_1^T \mathbf{v}_1}, \frac{1}{\mathbf{v}_2^T \mathbf{v}_2}, \dots, \frac{1}{\mathbf{v}_k^T \mathbf{v}_k} \right] \right).$$

Then

$$[\mathbf{w}]_{\mathcal{B}} = D \begin{bmatrix} \mathbf{v}_1^T \mathbf{w} \\ \mathbf{v}_2^T \mathbf{w} \\ \vdots \\ \mathbf{v}_k^T \mathbf{w} \end{bmatrix} = D \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_k^T \end{bmatrix} \mathbf{w} = D [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_k]^T \mathbf{w}.$$

If \mathcal{B} is an orthonormal basis then $D = I$ which further simplifies the formula:

$$[\mathbf{w}]_{\mathcal{B}} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_k]^T \mathbf{w}.$$

The idea of choosing a “nice” basis (one adapted to the problem at hand) is a recurring theme in linear algebra and its applications.

Example 6.5. Find the coordinates of $\mathbf{w} = \begin{bmatrix} 3 \\ 2 \\ 5 \end{bmatrix}$ with respect to the orthogonal basis

$$\mathcal{B} = \left(\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} \right).$$

In order to convert an orthogonal basis to an orthonormal basis, we need to divide each vector in the basis by its length. Thus, in the above example,

$$\mathbf{q}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{q}_2 = \frac{1}{\sqrt{6}} \begin{bmatrix} -1 \\ 2 \\ 1 \end{bmatrix}, \quad \mathbf{q}_3 = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix},$$

will form an orthonormal basis.

Theorem 6.6. *Let*

$$Q = [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_k].$$

$\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}$ is an orthonormal set if and only if $Q^T Q = I$.

Proof.

$$Q^T Q = \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_k^T \end{bmatrix} [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_k] = \begin{bmatrix} \mathbf{q}_1^T \mathbf{q}_1 & \mathbf{q}_1^T \mathbf{q}_2 & \cdots & \mathbf{q}_1^T \mathbf{q}_k \\ \mathbf{q}_2^T \mathbf{q}_1 & \mathbf{q}_2^T \mathbf{q}_2 & \cdots & \mathbf{q}_2^T \mathbf{q}_k \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}_k^T \mathbf{q}_1 & \mathbf{q}_k^T \mathbf{q}_2 & \cdots & \mathbf{q}_k^T \mathbf{q}_k \end{bmatrix}$$

Thus $Q^T Q = I$ if and only if

$$\mathbf{q}_i^T \mathbf{q}_j = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j. \end{cases}$$

In other words, $Q^T Q = I$ if and only if $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}$ is an orthonormal set. \square

Definition 6.7 (Orthogonal Matrix). A square $n \times n$ matrix Q whose columns form an orthonormal set is an *orthogonal matrix*.

Note the columns must form an *orthonormal set* and not just an orthogonal set. The terminology is historical; it would have been better to call Q an orthonormal matrix! The columns of an orthogonal matrix form an orthonormal basis for \mathbf{R}^n . By the above theorem, we have that Q is orthogonal if and only if $Q^T Q = I$ and so, since Q is square,

$$Q^{-1} = Q^T.$$

Thus

Theorem 6.8. *A square matrix Q is orthogonal if and only if*

$$Q^{-1} = Q^T.$$

Orthogonal matrices are about the only case where the computation of the inverse is justified!

Example 6.9. Show that

$$B = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

is an orthogonal matrix.

This example is a rotation in \mathbf{R}^3 (rotating about the z -axis). Rotations preserve both length of vectors and angles between vectors; that is, they preserve the dot product of two vectors. Orthogonal transformations do the same and so can be considered as a generalization of rotations to \mathbf{R}^n . We have

Theorem 6.10. *Let Q be an $n \times n$ matrix. The following statements are equivalent.*

(a) Q is orthogonal.

(b) $\|Q\mathbf{x}\| = \|\mathbf{x}\|$ for all $\mathbf{x} \in \mathbf{R}^n$ (preserves lengths).

(c) $Q\mathbf{x} \cdot Q\mathbf{y} = \mathbf{x} \cdot \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in \mathbf{R}^n$ (preserves angles).

Proof. (a) \Rightarrow (c).

An orthogonal matrix does not distort geometry

$$Q\mathbf{x} \cdot Q\mathbf{y} = (Q\mathbf{x})^T Q\mathbf{y} = \mathbf{x}^T Q^T Q\mathbf{y} = \mathbf{x}^T \mathbf{y} = \mathbf{x} \cdot \mathbf{y}$$

since Q is orthogonal.

(c) \Rightarrow (b).

$$\|Q\mathbf{x}\|^2 = Q\mathbf{x} \cdot Q\mathbf{x} = \mathbf{x} \cdot \mathbf{x} = \|\mathbf{x}\|^2.$$

(b) \Rightarrow (a). First note that

$$\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2 = (\mathbf{x} + \mathbf{y}) \cdot (\mathbf{x} + \mathbf{y}) - (\mathbf{x} - \mathbf{y}) \cdot (\mathbf{x} - \mathbf{y}) = 4\mathbf{x} \cdot \mathbf{y}$$

for any $\mathbf{x}, \mathbf{y} \in \mathbf{R}^n$. In particular

$$\begin{aligned} Q\mathbf{x} \cdot Q\mathbf{y} &= \frac{1}{4} (\|Q(\mathbf{x} + \mathbf{y})\|^2 - \|Q(\mathbf{x} - \mathbf{y})\|^2) \\ &= \frac{1}{4} (\|\mathbf{x} + \mathbf{y}\|^2 - \|\mathbf{x} - \mathbf{y}\|^2) = \mathbf{x} \cdot \mathbf{y}. \end{aligned}$$

This shows that (b) \Rightarrow (c). Let \mathbf{e}_i be the i^{th} standard basis vector. Then $\mathbf{q}_i = Q\mathbf{e}_i$ and so

$$\mathbf{q}_i \cdot \mathbf{q}_j = Q\mathbf{e}_i \cdot Q\mathbf{e}_j = \mathbf{e}_i \cdot \mathbf{e}_j = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j. \end{cases}$$

Thus Q is orthogonal. □

This result shows that any linear transformation that preserves length, automatically will preserve angles and thus be an orthogonal transformation. Finally we have

Theorem 6.11. *Let Q be an orthogonal matrix.*

(a) $Q^{-1} = Q^T$ is orthogonal.

(b) The rows of Q form an orthonormal set.

(c) $\det Q = \pm 1$.

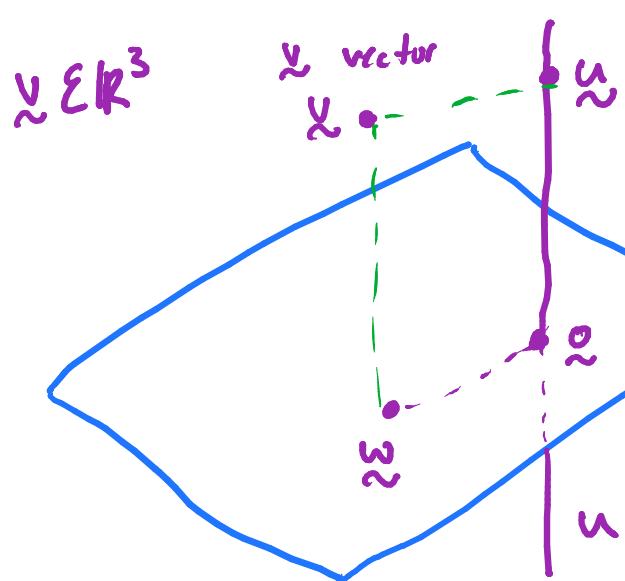
(d) If Q_1 and Q_2 are orthogonal matrices then $Q_1 Q_2$ is also an orthogonal matrix.

Let's look at c) $(Q_x) \cdot (Q_y) = x \cdot y$

$$\begin{aligned}
 (Q_x) \cdot (Q_y) &= (Q_x)^T (Q_y) \\
 &= x^T Q^T Q y \\
 &= x^T I y \\
 &= x^T y \\
 &= x \cdot y
 \end{aligned}$$

Orthogonal complements

We can determine a plane in \mathbb{R}^3 by point f & direction perpendicular to the plane



If the plane goes through \mathbb{Q} , then we have a subspace.

w (the plane)

& u the normal line.

Such that any pair of vectors $w \in W$ & $u \in U$ are orthogonal to one another

Namely, we can "decompose" space into orthogonal components

Definition

Let W be a subspace of \mathbb{R}^n

The orthogonal complement of W , W^\perp

is $W^\perp = \{v \in \mathbb{R}^n : v \text{ is orthogonal to every vector in } W\}$

7 Projections

Poole Ch. 5.2

We have previously seen the case of a projection of a vector onto a single vector (or, equivalently, a one dimensional subspace). We will generalize this idea to projection onto an r -dimensional subspace. For example, when one wants to represent a 3-dimensional object on, say, a computer screen, we must project the object onto a 2-dimensional plane (namely the screen).

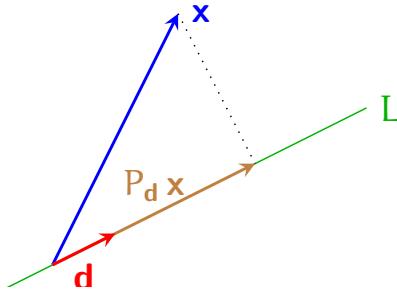
For the projection onto a line, L , in \mathbf{R}^2 passing through the origin, we have seen that the standard matrix is given by

$$P_d = \frac{1}{d_1^2 + d_2^2} \begin{bmatrix} d_1^2 & d_1 d_2 \\ d_1 d_2 & d_2^2 \end{bmatrix}$$

where

$$\mathbf{d} = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

is any vector parallel to (the direction of) L .



Let

$$\mathbf{u} = \frac{1}{\sqrt{d_1^2 + d_2^2}} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

be a *unit* vector in the direction of \mathbf{d} . Now

$$\mathbf{u}\mathbf{u}^T = \frac{1}{d_1^2 + d_2^2} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \begin{bmatrix} d_1 & d_2 \end{bmatrix} = \frac{1}{d_1^2 + d_2^2} \begin{bmatrix} d_1^2 & d_1 d_2 \\ d_1 d_2 & d_2^2 \end{bmatrix}$$

and so

$$P_d = \mathbf{u}\mathbf{u}^T.$$

This is a very convenient form for the standard matrix. Note that

$$P_d^T = (\mathbf{u}\mathbf{u}^T)^T = \mathbf{u}\mathbf{u}^T = P_d$$

and so P_d is a *symmetric* matrix. Since \mathbf{u} is a unit vector

$$\mathbf{u}^T \mathbf{u} = 1$$

and so

$$P_d^2 = \mathbf{u}(\mathbf{u}^T \mathbf{u}) \mathbf{u}^T = \mathbf{u}\mathbf{u}^T = P_d.$$

A matrix A such that $A^2 = A$ is called *idempotent*. This reflects the fact that if we project a vector that lies in the subspace then it will be unchanged (that is; applying a projection twice has the same effect as applying it once). Furthermore,

$$P_d = \mathbf{u}\mathbf{u}^T = \frac{1}{\sqrt{d_1^2 + d_2^2}} \begin{bmatrix} d_1\mathbf{u} & d_2\mathbf{u} \end{bmatrix}$$

and so

$$\text{col}(P_d) = \text{span}(\mathbf{u}).$$

Thus the one dimensional subspace which P_d projects onto is its column space.

Definition 7.1 (Projection Matrix). An $n \times n$ matrix P is called a *projection matrix* if it is symmetric ($P^T = P$) and idempotent ($P^2 = P$).

The following theorem shows that this definition captures the idea of a projection at least in the 1-dimensional case.

Theorem 7.2. *Let P be a rank 1 projection matrix. Then*

$$P = \mathbf{u}\mathbf{u}^T$$

where $\mathbf{u} \in \text{col}(P)$ is a unit vector. Furthermore

$$P\mathbf{x} = \text{proj}_{\mathbf{u}}(\mathbf{x})$$

and so P is the standard matrix for the linear transformation $\text{proj}_{\mathbf{u}}(\mathbf{x})$.

Proof. Since P has rank 1, we have

$$P = \mathbf{u}\mathbf{v}^T = [v_1\mathbf{u} \ v_2\mathbf{u} \ \cdots \ v_n\mathbf{u}]$$

where we can assume that, without loss of generality, \mathbf{u} is a unit vector (why?). We must show $\mathbf{v} = \mathbf{u}$. Since P is idempotent, we have

$$O = P^2 - P = \mathbf{u}(\mathbf{v}^T\mathbf{u})\mathbf{v}^T - \mathbf{u}\mathbf{v}^T.$$

Now $\mathbf{v}^T\mathbf{u}$ is a scalar (it is a dot product). Therefore we have

$$\mathbf{u}(\mathbf{v}^T\mathbf{u})\mathbf{v}^T = (\mathbf{v}^T\mathbf{u})\mathbf{u}\mathbf{v}^T.$$

Thus

$$O = (\mathbf{v}^T\mathbf{u} - 1)\mathbf{u}\mathbf{v}^T$$

and so

$$\mathbf{v}^T\mathbf{u} = 1.$$

Furthermore P is symmetric and so

$$P^T = \mathbf{v}\mathbf{u}^T = \mathbf{u}\mathbf{v}^T = P.$$

Now \mathbf{u} is a unit vector, $\mathbf{u}^T\mathbf{u} = 1$ and so

$$\begin{aligned} \mathbf{v} = \mathbf{v}(\mathbf{u}^T\mathbf{u}) &= (\mathbf{v}\mathbf{u}^T)\mathbf{u} = \mathbf{u}\mathbf{v}^T\mathbf{u} && \text{since } P \text{ is symmetric} \\ &= \mathbf{u}(\mathbf{v}^T\mathbf{u}) = \mathbf{u} && \text{since } P \text{ is idempotent} \end{aligned}$$

as required. Clearly $\mathbf{u} \in \text{col}(P)$. Finally, since \mathbf{u} is a unit vector, we have

$$P\mathbf{x} = \mathbf{u}\mathbf{u}^T\mathbf{x} = (\mathbf{u} \cdot \mathbf{x})\mathbf{u} = \text{proj}_{\mathbf{u}}(\mathbf{x}).$$

□

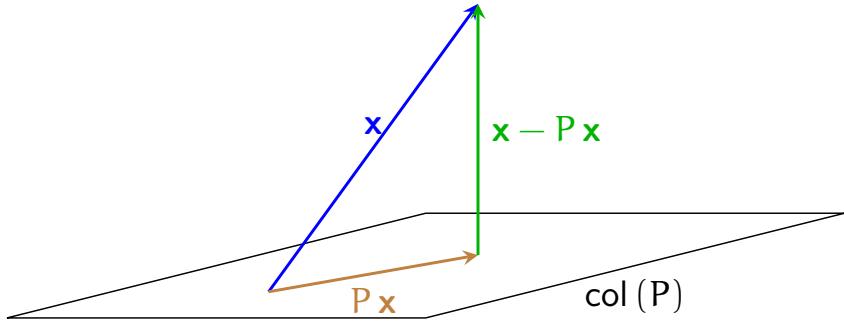
This result generalizes our earlier observations in \mathbf{R}^2 to \mathbf{R}^n . The projection of a vector $x \in \mathbf{R}^n$ onto a direction given by the unit vector $u \in \mathbf{R}^n$ is given by $uu^T x$. Note that there are two choices for the unit vector ($\pm u$) but that the projection formula is independent of that choice.

Example 7.3. Find the projection of $x = \begin{bmatrix} 1 \\ -1 \\ 2 \end{bmatrix}$ onto the direction $d = \begin{bmatrix} 4 \\ 0 \\ -3 \end{bmatrix}$.

What is the effect of projection matrices of higher rank? We know that Px is a linear combination of the columns of P and so P might act as a *projection onto its column space*. At this stage we have two options to address this question. First we could consider more examples. For example, we could examine the projection of a vector onto a 2-dimensional subspace in \mathbf{R}^3 (that is, onto a plane passing through the origin). The second alternative is to try to generalize the properties we have found in the rank 1 case.

Exercise 7.4. Do Problems 6–12 on pages 378–379 (pages 364–365 in the second edition) in Poole. This computes the projection of a vector in \mathbf{R}^3 onto a plane passing through the origin.

The idea of projection is to split a vector x into a component which lies in a given subspace (Px) and a component which is perpendicular to that subspace ($x - Px$).



The two crucial *geometric* properties for a projection are

- Any vector $v \in \text{col}(P)$ remains unchanged; that is $Pv = v$.
- For any x , the “orthogonal” component $x - Px$ is orthogonal to *every* vector in $\text{col}(P)$.

The question that we must address is whether we have captured these two geometric properties in our (algebraic) definition of a projection matrix. As a first step, we can check that these two properties hold in the 1-dimensional case by using the explicit form of $P = uu^T$.

We see that if $v \in \text{col}(P)$ then $v = cu$ for some scalar c and so

$$Pv = uu^T cu = cu = v$$

since \mathbf{u} is a unit vector. Thus, in the rank 1 case, P leaves unchanged anything in its column space. In addition, for any \mathbf{x} , we have

$$\mathbf{u}^T(\mathbf{x} - P\mathbf{x}) = \mathbf{u}^T\mathbf{x} - (\mathbf{u}^T\mathbf{u})\mathbf{u}^T\mathbf{x} = 0$$

since \mathbf{u} is a unit vector. Therefore $\mathbf{x} - P\mathbf{x}$ is orthogonal to \mathbf{u} and so is orthogonal to any vector in $\text{col}(P)$.

Let us consider the general case. Suppose P is a projection matrix. Does the associated linear transformation satisfy the two requirements for a projection? For any \mathbf{x} , we have

$$P(\mathbf{x} - P\mathbf{x}) = P\mathbf{x} - P^2\mathbf{x} = 0$$

since $P^2 = P$ and so $\mathbf{x} - P\mathbf{x} \in \text{null}(P)$. Thus we want the null space of P to be “orthogonal” to the column space of P . Let $\mathbf{y} \in \text{null}(P)$. Since $P^T = P$, the rows of P are the same as the columns of P and so

$$P = \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_n^T \end{bmatrix}$$

where \mathbf{p}_i are the columns of P . Thus

$$P\mathbf{y} \begin{bmatrix} \mathbf{p}_1^T \mathbf{y} \\ \mathbf{p}_2^T \mathbf{y} \\ \vdots \\ \mathbf{p}_n^T \mathbf{y} \end{bmatrix} = \mathbf{0}.$$

So \mathbf{y} is orthogonal to any vector in $\text{col}(P) = \text{span}(\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n)$.

The fact that P is idempotent guarantees that $\mathbf{x} - P\mathbf{x} \in \text{null}(P)$. The fact that P is symmetric guarantees that anything in the null space of P is orthogonal to any vector in the column space of P . In particular, if \mathbf{y} is in both $\text{null}(P)$ and $\text{col}(P)$ then $\mathbf{y}^T \mathbf{y} = 0$ and so $\mathbf{y} = \mathbf{0}$. Finally, for $\mathbf{v} \in \text{col}(P)$ we have $\mathbf{v} - P\mathbf{v} \in \text{col}(P)$ (why?) and $\mathbf{v} - P\mathbf{v} \in \text{null}(P)$.

Thus $\mathbf{v} - P\mathbf{v} = \mathbf{0}$ and so

$$P\mathbf{v} = \mathbf{v}$$

as required for a projection. Thus the linear transformation associated with a projection matrix is a projection in the geometric sense.

Theorem 7.5. *Let P be an $n \times n$ projection matrix of rank r . Let $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r\}$ be a basis for $\text{col}(P)$ and $\{\mathbf{v}_{r+1}, \mathbf{v}_{r+2}, \dots, \mathbf{v}_n\}$ be a basis for $\text{null}(P)$. Then $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ is a basis for \mathbf{R}^n . Furthermore, if $\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \dots + c_n\mathbf{v}_n$ then*

$$P\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \dots + c_r\mathbf{v}_r.$$

Geometrically, we are splitting a vector \mathbf{v} into a component that lies in the subspace, the *parallel component*

$$\text{proj}_{\text{col}(P)}(\mathbf{v}) = \mathbf{v}_{||} = P\mathbf{v} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \dots + c_r\mathbf{v}_r,$$

and a component that is orthogonal to the subspace, the *orthogonal* (or *perpendicular*) *component*

$$\text{perp}_{\text{col}(P)}(\mathbf{v}) = \mathbf{v}_\perp = \mathbf{v} - P\mathbf{v} = c_{r+1}\mathbf{v}_{r+1} + c_{r+2}\mathbf{v}_{r+2} + \cdots + c_n\mathbf{v}_n.$$

We have yet to find an efficient way (or for that matter any way) to compute P . We need a symmetric and idempotent matrix with a prescribed column space.

Note that, for $i = 1, 2, \dots, r$,

$$\mathbf{v}_i \cdot \mathbf{v} = \mathbf{v}_i^T \mathbf{v} = c_1 \mathbf{v}_i^T \mathbf{v}_1 + c_2 \mathbf{v}_i^T \mathbf{v}_2 + \cdots + c_r \mathbf{v}_i^T \mathbf{v}_r$$

since \mathbf{v}_i is orthogonal to everything in $\text{null}(P)$. This gives a system of r equations for the r unknowns c_1, c_2, \dots, c_r

$$\begin{bmatrix} \mathbf{v}_1^T \mathbf{v}_1 & \mathbf{v}_1^T \mathbf{v}_2 & \cdots & \mathbf{v}_1^T \mathbf{v}_r \\ \mathbf{v}_2^T \mathbf{v}_1 & \mathbf{v}_2^T \mathbf{v}_2 & \cdots & \mathbf{v}_2^T \mathbf{v}_r \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{v}_r^T \mathbf{v}_1 & \mathbf{v}_r^T \mathbf{v}_2 & \cdots & \mathbf{v}_r^T \mathbf{v}_r \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_r \end{bmatrix} = \begin{bmatrix} \mathbf{v}_1^T \mathbf{v} \\ \mathbf{v}_2^T \mathbf{v} \\ \vdots \\ \mathbf{v}_r^T \mathbf{v} \end{bmatrix}$$

This system can be rewritten

$$\begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_r^T \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T & \mathbf{v}_2^T & \cdots & \mathbf{v}_r^T \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_r \end{bmatrix} = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_r^T \end{bmatrix} \mathbf{v}.$$

Since $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r\}$ are linearly independent (they form a basis), the rank of the coefficient matrix is r and so this system has a unique solution for all \mathbf{v} . Let

$$A = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_r];$$

a matrix whose column space is given by $\text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r)$. Then this system takes the form

$$A^T A \mathbf{c} = A^T \mathbf{v}.$$

Example 7.6. Find the projection of $\mathbf{v} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$ onto the subspace spanned by

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix} \text{ and } \mathbf{v}_2 = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}.$$

This is a cumbersome way to compute the projection (particularly if one wants to project many vectors onto the same subspace). With the rank 1 case, we found a very nice form for P by choosing a special basis for the subspace; namely a unit vector to span the 1-dimensional subspace. Can we do a similar thing with the higher rank cases?

The coefficient matrix is

$$\begin{bmatrix} \mathbf{v}_1^T \mathbf{v}_1 & \mathbf{v}_1^T \mathbf{v}_2 & \cdots & \mathbf{v}_1^T \mathbf{v}_r \\ \mathbf{v}_2^T \mathbf{v}_1 & \mathbf{v}_2^T \mathbf{v}_2 & \cdots & \mathbf{v}_2^T \mathbf{v}_r \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{v}_r^T \mathbf{v}_1 & \mathbf{v}_r^T \mathbf{v}_2 & \cdots & \mathbf{v}_r^T \mathbf{v}_r \end{bmatrix}$$

If we choose an orthogonal basis for the column space then the coefficient matrix would become a *diagonal* matrix. In this case, the solution of the system is immediate and is given by

$$c_j = \frac{\mathbf{v}_j^T \mathbf{v}}{\mathbf{v}_j^T \mathbf{v}_j}.$$

Note that

$$c_j \mathbf{v}_j = \frac{\mathbf{v}_j^T \mathbf{v}}{\mathbf{v}_j^T \mathbf{v}_j} \mathbf{v}_j = \frac{\mathbf{v}_j^T \mathbf{v} \mathbf{v}_j}{\mathbf{v}_j^T \mathbf{v}_j} = \frac{\mathbf{v}_j \mathbf{v}_j^T}{\mathbf{v}_j^T \mathbf{v}_j} \mathbf{v}$$

since $\mathbf{v}_j^T \mathbf{v}$ is a scalar and so

$$P\mathbf{v} = \left(\frac{\mathbf{v}_1 \mathbf{v}_1^T}{\mathbf{v}_1^T \mathbf{v}_1} + \frac{\mathbf{v}_2 \mathbf{v}_2^T}{\mathbf{v}_2^T \mathbf{v}_2} + \cdots + \frac{\mathbf{v}_r \mathbf{v}_r^T}{\mathbf{v}_r^T \mathbf{v}_r} \right) \mathbf{v}$$

Furthermore if we chose an orthonormal basis then the coefficient matrix will be an identity matrix. In this case the solution simplifies to

$$P\mathbf{v} = (\mathbf{v}_1 \mathbf{v}_1^T + \mathbf{v}_2 \mathbf{v}_2^T + \cdots + \mathbf{v}_r \mathbf{v}_r^T) \mathbf{v}$$

and we see

$$P = \mathbf{v}_1 \mathbf{v}_1^T + \mathbf{v}_2 \mathbf{v}_2^T + \cdots + \mathbf{v}_r \mathbf{v}_r^T.$$

8 Gram Schmidt process

Poole Ch. 5.2–5.3

We have seen that if $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_k\}$ is an orthonormal basis for a subspace W then

$$\text{proj}_W(\mathbf{x}) = (\mathbf{q}_1\mathbf{q}_1^T + \mathbf{q}_2\mathbf{q}_2^T + \cdots + \mathbf{q}_k\mathbf{q}_k^T)\mathbf{x} = QQ^T\mathbf{x}$$

where

$$Q = [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_k].$$

Thus we can decompose the projection onto W

$$\text{proj}_W(\mathbf{x}) = \text{proj}_{\mathbf{q}_1}(\mathbf{x}) + \text{proj}_{\mathbf{q}_2}(\mathbf{x}) + \cdots + \text{proj}_{\mathbf{q}_k}(\mathbf{x})$$

to the sum of projections onto each of the vectors in the basis. Thus, if we *choose* an orthonormal basis, then the projection onto W can be computed by computing the projection onto each of the 1-dimensional subspaces given by $\text{span}(\mathbf{q}_i)$. This suggests that if we begin with an arbitrary basis $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ for W then we might be able to “orthogonalize” one vector at a time.

Let us start with a simple example.

Example 8.1. Find an orthogonal basis for $W = \text{span}(\mathbf{x}_1, \mathbf{x}_2)$ where

$$\mathbf{x}_1 = \begin{bmatrix} -2 \\ 0 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

Note that the output from this method depends on the *order* of the original basis. If we had taken

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \quad \text{and} \quad \mathbf{x}_2 = \begin{bmatrix} -2 \\ 0 \\ 1 \end{bmatrix}$$

then we would have obtained a different orthogonal basis (check!).

Now let us suppose W is a three dimensional subspace with $W = \text{span}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$. We proceed as above by setting

$$\begin{aligned} \mathbf{v}_1 &= \mathbf{x}_1 \\ \mathbf{v}_2 &= \mathbf{x}_2 - \frac{\mathbf{x}_2 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 \end{aligned}$$

We now have $\text{span}(\mathbf{v}_1, \mathbf{v}_2) = \text{span}(\mathbf{x}_1, \mathbf{x}_2) = W_2$, say, and \mathbf{v}_2 orthogonal to \mathbf{v}_1 . We need a third vector \mathbf{v}_3 which lies in W which is orthogonal to *both* \mathbf{v}_1 and \mathbf{v}_2 (that is, to W_2). Thus

$$\mathbf{v}_3 = \text{perp}_{W_2}(\mathbf{x}_3) = \mathbf{x}_3 - \text{proj}_{W_2}(\mathbf{x}_3).$$

But $\{\mathbf{v}_1, \mathbf{v}_2\}$ form an orthogonal basis for W_2 and so

$$\text{proj}_{W_2}(\mathbf{x}_3) = \text{proj}_{\mathbf{v}_1}(\mathbf{x}_3) + \text{proj}_{\mathbf{v}_2}(\mathbf{x}_3) = \frac{\mathbf{x}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 + \frac{\mathbf{x}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2$$

(since we have not required $\{\mathbf{v}_1, \mathbf{v}_2\}$ to form an *orthonormal* set). Thus

$$\mathbf{v}_3 = \mathbf{x}_3 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2$$

will be orthogonal to \mathbf{v}_1 and \mathbf{v}_2 . Explicitly, we have

$$\begin{aligned}\mathbf{v}_3 \cdot \mathbf{v}_1 &= \mathbf{x}_3 \cdot \mathbf{v}_1 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 \cdot \mathbf{v}_1 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 \cdot \mathbf{v}_1 \\ &= \mathbf{x}_3 \cdot \mathbf{v}_1 - \mathbf{x}_3 \cdot \mathbf{v}_1 = 0 \\ \mathbf{v}_3 \cdot \mathbf{v}_2 &= \mathbf{x}_3 \cdot \mathbf{v}_2 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 \cdot \mathbf{v}_2 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 \cdot \mathbf{v}_2 \\ &= \mathbf{x}_3 \cdot \mathbf{v}_2 - \mathbf{x}_3 \cdot \mathbf{v}_2 = 0.\end{aligned}$$

This construction generalises to give:

Theorem 8.2 (Gram-Schmidt Process). *Let $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ be a basis for a subspace W of \mathbf{R}^n . Define*

$$\begin{aligned}\mathbf{v}_1 &= \mathbf{x}_1 \\ \mathbf{v}_2 &= \mathbf{x}_2 - \frac{\mathbf{x}_2 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 \\ \mathbf{v}_3 &= \mathbf{x}_3 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 - \frac{\mathbf{x}_3 \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 \\ &\vdots \\ \mathbf{v}_k &= \mathbf{x}_k - \frac{\mathbf{x}_k \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 - \frac{\mathbf{x}_k \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 - \cdots - \frac{\mathbf{x}_k \cdot \mathbf{v}_{k-1}}{\mathbf{v}_{k-1} \cdot \mathbf{v}_{k-1}} \mathbf{v}_{k-1}\end{aligned}$$

and

$$W_i = \text{span}(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_i).$$

Then, for each $i = 1, 2, \dots, k$, $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_i\}$ is an orthogonal basis for W_i . In particular $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k\}$ is an orthogonal basis for $W_k = W$.

Proof. Induction based upon the argument above. \square

Note that it is the previously computed \mathbf{v}_i that are on the right hand side of these expressions (and *not* the \mathbf{x}_i).

Example 8.3. Find an orthogonal basis for $W = \text{span}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ where

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{x}_3 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}.$$

The Gram-Schmidt process is frequently messy to complete. For hand calculations, one can scale to vectors \mathbf{v}_i to avoid fractions. Thus, in the above example, we could have chosen

$$\mathbf{v}'_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -3 \end{bmatrix}$$

and then (recalculate?) \mathbf{v}_3 and scale to get

$$\mathbf{v}'_3 = \begin{bmatrix} 1 \\ 1 \\ -2 \\ 0 \end{bmatrix}.$$

The set $\{\mathbf{v}_1, \mathbf{v}'_2, \mathbf{v}'_3\}$ also forms an orthogonal basis for W . To obtain an orthonormal basis, we normalize each vector in any orthogonal basis. Thus

$$\begin{aligned} \mathbf{q}_1 &= \frac{1}{\|\mathbf{v}_1\|} \mathbf{v}_1 = \frac{1}{2} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ \mathbf{q}_2 &= \frac{1}{\|\mathbf{v}'_2\|} \mathbf{v}'_2 = \frac{\sqrt{3}}{6} \begin{bmatrix} 1 \\ 1 \\ 1 \\ -3 \end{bmatrix} \\ \mathbf{q}_3 &= \frac{1}{\|\mathbf{v}'_3\|} \mathbf{v}'_3 = \frac{\sqrt{6}}{6} \begin{bmatrix} 1 \\ 1 \\ -2 \\ 0 \end{bmatrix} \end{aligned}$$

is an orthonormal basis for W .

Remark 8.4. When the Gram-Schmidt Process is implemented on a computer, there is almost always round off error. This round off error leads to a loss of orthogonality of the vectors \mathbf{q}_i . To minimize this, the vectors \mathbf{v}_i are normalized to give \mathbf{q}_i as soon as they are computed rather than at the end of the computation. The remaining vectors \mathbf{x}_j are modified to be orthogonal to \mathbf{q}_i . This procedure is known as the *Modified Gram-Schmidt Process*. For hand computations, where we are using exact arithmetic, the issue of round off error does not arise. In this case we wish to avoid dealing with fractions and square roots as long as possible and so we delay scaling until the end.

One important use of the Gram-Schmidt Process is to construct a basis containing a given vector (or set of vectors).

Example 8.5. Find an orthogonal basis for \mathbf{R}^3 that contains

$$\mathbf{v}_1 = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}.$$

8.1 QR factorization

Let A be a $m \times n$ matrix with linearly independent columns (thus $m \geq n$ and $\text{rank}(A) = n$). If we apply the Gram-Schmidt process to A we obtain a very

useful factorization of A into a product of a matrix Q whose columns form an orthonormal set and an upper triangular matrix R . This factorization is extremely useful in many numerical problems from the computation of eigenvalues to least squares approximation.

Definition 8.6. The factorization above is called the *QR factorization of A* .

Let

$$A = [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_n]$$

and $\{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n\}$ be the set of orthonormal vectors obtained from $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\}$ by the Gram-Schmidt Process with normalization. From the Gram-Schmidt Theorem, $W_1 = \text{span}(\mathbf{a}_1) = \text{span}(\mathbf{q}_1)$ and so

$$\mathbf{a}_1 = r_{11}\mathbf{q}_1$$

for some scalar r_{11} . Similarly $W_2 = \text{span}(\mathbf{a}_1, \mathbf{a}_2) = \text{span}(\mathbf{q}_1, \mathbf{q}_2)$ and so

$$\mathbf{a}_2 = r_{12}\mathbf{q}_1 + r_{22}\mathbf{q}_2$$

for some scalars r_{12} and r_{22} . Continuing, we obtain

$$\mathbf{a}_i = r_{1i}\mathbf{q}_1 + r_{2i}\mathbf{q}_2 + \cdots + r_{ii}\mathbf{q}_i$$

for $i = 1, 2, \dots, n$.

Thus

$$\begin{aligned} A &= [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_n] \\ &= [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_n] \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix} = QR \end{aligned}$$

Q has orthonormal columns (if A is square then Q is an orthogonal matrix) and $\text{col}(A) = \text{col}(Q)$. If $r_{ii} = 0$ then $\mathbf{a}_i \in W_{i-1}$ and so $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_i\}$ would not be linearly independent. Therefore $r_{ii} \neq 0$ and so R is upper triangular with non-zero diagonal elements. Thus R is invertible. The entries in R are computed during the Gram-Schmidt Process. At stage i of the Gram-Schmidt Process, we have

$$\mathbf{v}_i = \mathbf{a}_i - \frac{\mathbf{a}_i \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 - \frac{\mathbf{a}_i \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 - \cdots - \frac{\mathbf{a}_i \cdot \mathbf{v}_{i-1}}{\mathbf{v}_{i-1} \cdot \mathbf{v}_{i-1}} \mathbf{v}_{i-1}$$

and so

$$\mathbf{a}_i = \mathbf{v}_i + \frac{\mathbf{a}_i \cdot \mathbf{v}_1}{\mathbf{v}_1 \cdot \mathbf{v}_1} \mathbf{v}_1 + \frac{\mathbf{a}_i \cdot \mathbf{v}_2}{\mathbf{v}_2 \cdot \mathbf{v}_2} \mathbf{v}_2 + \cdots + \frac{\mathbf{a}_i \cdot \mathbf{v}_{i-1}}{\mathbf{v}_{i-1} \cdot \mathbf{v}_{i-1}} \mathbf{v}_{i-1}.$$

Now

$$\mathbf{q}_j = \frac{1}{\|\mathbf{v}_j\|} \mathbf{v}_j$$

and so

$$\frac{\mathbf{a}_i \cdot \mathbf{v}_j}{\mathbf{v}_j \cdot \mathbf{v}_j} \mathbf{v}_j = \frac{\|\mathbf{v}_j\| (\mathbf{a}_i \cdot \mathbf{q}_j)}{\mathbf{v}_j \cdot \mathbf{v}_j} \|\mathbf{v}_j\| \mathbf{q}_j = (\mathbf{a}_i \cdot \mathbf{q}_j) \mathbf{q}_j.$$

Therefore

$$\mathbf{a}_i = \|\mathbf{v}_i\| \mathbf{q}_i + (\mathbf{a}_i \cdot \mathbf{q}_1) \mathbf{q}_1 + (\mathbf{a}_i \cdot \mathbf{q}_2) \mathbf{q}_2 + \cdots + (\mathbf{a}_i \cdot \mathbf{q}_{i-1}) \mathbf{q}_{i-1}$$

and thus

$$r_{ji} = \begin{cases} \mathbf{a}_i \cdot \mathbf{q}_j & \text{for } j = 1, 2, \dots, i-1 \\ \|\mathbf{v}_i\| & \text{for } j = i. \end{cases}$$

Since \mathbf{q}_j for $j = 1, 2, \dots, i-1$ have been computed in previous steps, we can compute all these coefficients at step i of the Gram-Schmidt Process. This method also produces an upper triangular matrix R which has strictly positive entries on the diagonal (not just non-zero).

Note that, since Q has orthonormal columns, $Q^T Q = I$ and so

$$R = Q^T QR = Q^T A.$$

Moreover, since $\text{col}(A) = \text{col}(Q)$,

$$\text{proj}_{\text{col}(A)}(\mathbf{x}) = \text{proj}_{\text{col}(Q)}(\mathbf{x}) = QQ^T \mathbf{x}.$$

Example 8.7. Find a QR factorization for

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

The QR decomposition can be generalized for any matrix (not just for matrices whose columns are linearly independent). However, if the columns of A are not linearly independent, then some of the diagonal elements of R are 0. MATLAB has its `qr` command to compute QR decompositions. Thus

```
>> A = [1 1 1; 1 1 1; 1 1 0; 1 0 0];
>> [Q, R] = qr(A, 0)

Q =
    -0.5000   -0.2887    0.4082
    -0.5000   -0.2887    0.4082
    -0.5000   -0.2887   -0.8165
    -0.5000    0.8660      0

R =
    -2.0000   -1.5000   -1.0000
        0   -0.8660   -0.5774
        0       0    0.8165
```

MATLAB has chosen its normalizations differently from what we did above. This explains the differences in signs. Note that

```

>> [Q, R] = qr(A)
Q =
   -0.5000   -0.2887    0.4082   -0.7071
   -0.5000   -0.2887    0.4082    0.7071
   -0.5000   -0.2887   -0.8165   -0.0000
   -0.5000    0.8660        0    0.0000

R =
   -2.0000   -1.5000   -1.0000
      0   -0.8660   -0.5774
      0        0    0.8165
      0        0        0

```

In this case, MATLAB has computed a fourth vector

$$\mathbf{q}_4 = \frac{\sqrt{2}}{2} \begin{bmatrix} -1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

so that Q is orthogonal (that is, $\{\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{q}_4\}$ will form an orthonormal basis for \mathbf{R}^4). This necessitates adding a row of zeros to R , since the columns of A do not depend on \mathbf{q}_4 , to achieve $A = QR$.

```

>> Q * R
ans =
   1.0000   1.0000   1.0000
   1.0000   1.0000   1.0000
   1.0000   1.0000     0
   1.0000   0.0000   0.0000

```

If we have the QR decomposition of A then we can quickly compute the solution of $A\mathbf{x} = \mathbf{b}$ since $A\mathbf{x} = QR\mathbf{x}$ and so

$$R\mathbf{x} = Q^T QR\mathbf{x} = Q^T A\mathbf{x} = Q^T \mathbf{b}.$$

Since R is upper triangular, the system $R\mathbf{x} = Q^T \mathbf{b}$ can be solved by back substitution. This approach avoids row reduction. Round off error is a major concern in row reduction since we have ensure that off diagonal elements are zero. Any round off error can dramatic decrease the accuracy of the solution. The QR approach is *more stable numerically*. Of course, with exact arithmetic, this problem does not occur.

8.2 Orthogonal Complements

From our discussion on projections, we have seen that, for a symmetric matrix A , $\text{col}(A)$ is orthogonal to $\text{null}(A)$. What happens more generally? Can we get an insight into the geometry of the subspaces associated with a matrix A ?

Definition 8.8. Let W be a subspace of \mathbf{R}^n . A vector $v \in \mathbf{R}^n$ is *orthogonal to W* if v is orthogonal to every vector in W . The set of all vectors orthogonal to W is called the *orthogonal complement of W* , denoted by W^\perp . Thus

$$W^\perp = \{\mathbf{v} \in \mathbf{R}^n : \mathbf{v} \cdot \mathbf{w} = 0 \text{ for all } \mathbf{w} \in W\}.$$

For a symmetric matrix A , $\text{col}(A)$ is the orthogonal complement of $\text{null}(A)$.

Theorem 8.9. Let W be a subspace of \mathbf{R}^n .

- (a) W^\perp is a subspace of \mathbf{R}^n .
- (b) $(W^\perp)^\perp = W$.
- (c) $W \cap W^\perp = \{\mathbf{0}\}$.
- (d) If $W = \text{span}(\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_k)$ then $\mathbf{v} \in W^\perp$ if and only if $\mathbf{v} \cdot \mathbf{w}_i = 0$ for each $i = 1, 2, \dots, k$.

Proof. Exercise. □

Note that

$$\text{perp}_W(\mathbf{v}) \in W^\perp$$

since the perpendicular component is orthogonal to any vector in W .

Example 8.10. Find the orthogonal complement of the plane

$$W = \left\{ \begin{bmatrix} x \\ y \\ z \end{bmatrix} : x + y - z = 0 \right\}.$$

Theorem 8.11 (Orthogonal Decomposition). Let W be a subspace of \mathbf{R}^n and $\mathbf{v} \in \mathbf{R}^n$. Then there exists unique vectors $\mathbf{w} \in W$ and $\mathbf{w}^\perp \in W^\perp$ such that

$$\mathbf{v} = \mathbf{w} + \mathbf{w}^\perp.$$

Furthermore,

$$\mathbf{w} = \text{proj}_W(\mathbf{v}) \quad \text{and} \quad \mathbf{w}^\perp = \text{perp}_W(\mathbf{v})$$

Proof. Clearly

$$\mathbf{w} = \text{proj}_W(\mathbf{v}) \in W, \quad \mathbf{w}^\perp = \text{perp}_W(\mathbf{v}) \in W^\perp$$

and

$$\mathbf{v} = \mathbf{w} + \mathbf{w}^\perp.$$

It only remains to show that this choice is unique. Suppose $\mathbf{v} = \mathbf{w}_1 + \mathbf{w}_1^\perp$ with $\mathbf{w}_1 \in W$ and $\mathbf{w}_1^\perp \in W^\perp$. Then $\mathbf{w} + \mathbf{w}^\perp = \mathbf{w}_1 + \mathbf{w}_1^\perp$ and so

$$\mathbf{w} - \mathbf{w}_1 = \mathbf{w}^\perp - \mathbf{w}_1^\perp.$$

However $\mathbf{w} - \mathbf{w}_1 \in W$ and $\mathbf{w}^\perp - \mathbf{w}_1^\perp \in W^\perp$. By part (c) of the above theorem, the only vector in common to both W and W^\perp is $\mathbf{0}$ and so

$$\mathbf{w} - \mathbf{w}_1 = \mathbf{w}^\perp - \mathbf{w}_1^\perp = \mathbf{0}.$$

Therefore $\mathbf{w} = \mathbf{w}_1$ and $\mathbf{w}^\perp = \mathbf{w}_1^\perp$ and so we have uniqueness. \square

An immediate consequence of this result is the following.

Theorem 8.12. *Let W be a subspace of \mathbf{R}^n . Then $W \oplus W^\perp = \mathbf{R}^n$ and*

$$\dim(W) + \dim(W^\perp) = n.$$

Furthermore

$$\text{perp}_W(\mathbf{v}) = \text{proj}_{W^\perp}(\mathbf{v})$$

for all $\mathbf{v} \in \mathbf{R}^n$.

Returning to the issue of the subspaces associated with a matrix A , we have

Theorem 8.13. *Let A be an $m \times n$ matrix. Then*

$$\begin{aligned} (\text{col}(A))^\perp &= \text{null}(A^T) \\ (\text{row}(A))^\perp &= \text{null}(A). \end{aligned}$$

Proof. If $\mathbf{x} \in (\text{col}(A))^\perp$ then \mathbf{x} is orthogonal to every column of $A = [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_n]$. This occurs if and only if $\mathbf{x}^T \mathbf{a}_i = 0$, that is

$$\mathbf{x}^T [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_n] = \mathbf{x}^T A = \mathbf{0}^T.$$

Therefore

$$A^T \mathbf{x} = \mathbf{0}$$

and so $\mathbf{x} \in \text{null}(A^T)$ as required. For the second identity, we replace A by A^T and note $\text{col}(A^T) = \text{row}(A)$. \square

For an arbitrary $m \times n$ matrix A , we see that $\text{row}(A)$ and $\text{null}(A)$ are orthogonal complements in \mathbf{R}^n whereas $\text{col}(A)$ and (A^T) are orthogonal complements in \mathbf{R}^m . These four subspaces are called the *fundamental subspaces* of A . The particular case of a symmetric $n \times n$ matrix, we have $\text{row}(A) = \text{col}(A)$ and $\text{null}(A)$ are orthogonal complements in \mathbf{R}^n .

Combining the above two theorems, we obtain

Theorem 8.14. *Let A be an $m \times n$ matrix. Then*

$$\begin{aligned} \text{rank}(A) + \text{nullity}(A) &= n \\ \text{rank}(A) + \text{nullity}(A^T) &= m. \end{aligned}$$

The first of these equations is the rank equation.

The linear transformation associated with A , $T_A(\mathbf{v}) = A\mathbf{v}$, is a mapping from \mathbf{R}^n into \mathbf{R}^m . For $\mathbf{v} \in \mathbf{R}^n$, we have, by the orthogonal decomposition theorem,

$$\mathbf{v} = \mathbf{v}_1 + \mathbf{v}_2$$

where

$$\mathbf{v}_1 = \text{proj}_{\text{row}(A)}(\mathbf{v}) \in \text{row}(A) \quad \text{and} \quad \mathbf{v}_2 = \text{proj}_{\text{null}(A)}(\mathbf{v}) \in \text{null}(A).$$

Thus

$$A\mathbf{v} = A(\mathbf{v}_1 + \mathbf{v}_2) = A\mathbf{v}_1.$$

Therefore the projection of \mathbf{v} onto $\text{null}(A)$ represents the information lost when applying the transformation T_A . If we restrict the domain of T_A to $\text{row}(A)$ then we will get an invertible map from $\text{row}(A)$ onto $\text{col}(A)$.

9 Least Squares

Poole Ch. 7.3

9.1 Overdetermined Systems

MATLAB will give a solution to an overdetermined system of equations, $A\mathbf{x} = \mathbf{b}$; that is, where A has more rows than columns (more equations than unknowns). For such a system, \mathbf{b} is unlikely to be in $\text{col}(A)$ and therefore the system is unlikely to have a solution. However if we project \mathbf{b} onto $\text{col}(A)$ then we get the system

$$A\mathbf{x} = \text{proj}_{\text{col}(A)}(\mathbf{b})$$

which will have a solution (note that if $\mathbf{b} \in \text{col}(A)$ then this system is identical to the original system). If we find the QR decomposition of A then this equation takes the form

$$QR\mathbf{x} = QQ^T\mathbf{b}$$

or

$$R\mathbf{x} = Q^T\mathbf{b}.$$

This is the so-called *least squares solution* for an overdetermined system and has the same form as a system which is consistent.

The residual (or “error”) in the least squares solution is

$$\mathbf{r} = \mathbf{b} - A\mathbf{x} = \mathbf{b} - \text{proj}_{\text{col}(A)}(\mathbf{b}) = \text{perp}_{\text{col}(A)}(\mathbf{b}).$$

The choice of the least squares solution \mathbf{x} minimizes the length of this vector. If we choose any vector \mathbf{y} to “be” the solution then the length of the residual would be

$$\|\mathbf{b} - Ay\|^2 = \|\mathbf{b} - Ax + Ax - Ay\|^2 = \|\mathbf{r}\|^2 + \|A(\mathbf{x} - \mathbf{y})\|^2 + 2\mathbf{r} \cdot A(\mathbf{x} - \mathbf{y}).$$

Now $A(\mathbf{x} - \mathbf{y}) \in \text{col}(A)$ and \mathbf{r} is orthogonal to anything in $\text{col}(A)$ and so

$$\|\mathbf{b} - Ay\|^2 = \|\mathbf{r}\|^2 + \|A(\mathbf{x} - \mathbf{y})\|^2 \geq \|\mathbf{r}\|^2.$$

Therefore the error in choosing \mathbf{y} is at least as bad as that for the least squares solution. Furthermore, \mathbf{y} minimizes the residual if and only if $\mathbf{x} - \mathbf{y} \in \text{null}(A)$. Thus $\text{null}(A)$ gives the ambiguity in the solution that minimizes the residual. If $\text{nullity}(A) = \dim(\text{null}(A)) = 0$ then the least squares solution is the *unique* solution that minimizes the residual.

Example 9.1. Find the (least squares) solution for $A\mathbf{x} = \mathbf{b}$ where

$$A = \begin{bmatrix} 1 & 1 \\ 2 & 3 \\ 2 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 3 \\ -1 \\ -1 \end{bmatrix}.$$

Problem 9.2. We have data from an experiment in the form of ordered pairs (x_i, y_i) . We want to find a relationship $y = f(x)$ that fits the data “well”.

Suppose we wish to “fit” a straight line to a set of data (x_i, y_i) ; that is we wish to find c_0 and c_1 such that

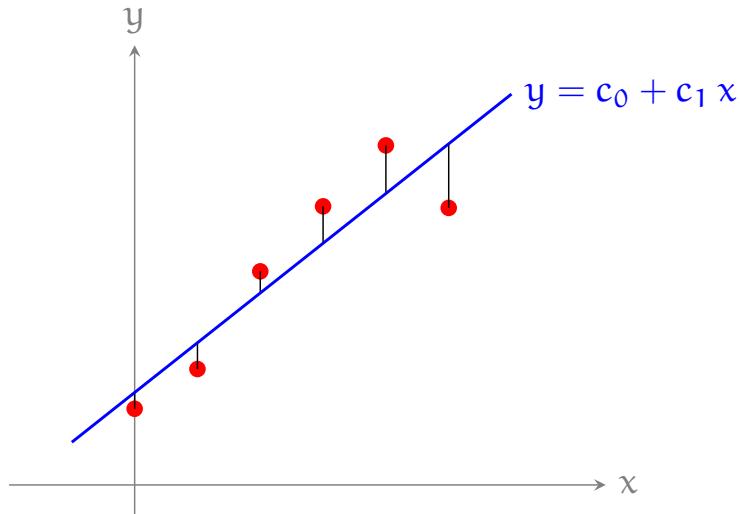
$$y_i = c_0 + c_1 x_i$$

If the relationship is exact and there is no error in the measurements then only two data points would be needed to determine c_0 and c_1 . In practice the linear relationship may only be an approximation and there will be error in the measurements. So more than two measurements are made and we try to find c_0 and c_1 that “best” fit the data

$$y_i \approx c_0 + c_1 x_i$$

That is, we try to minimize the length of the *residual*, \mathbf{r} , whose components are given by

$$r_i = y_i - (c_0 + c_1 x_i).$$



Let \mathbf{x} be the vector whose components are x_i and \mathbf{y} be the vector whose components are y_i . Furthermore let

$$A = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} = [\mathbf{1} \ \mathbf{x}]$$

where $\mathbf{1}$ is a vector of ones (that is, in MATLAB, `ones(n, 1)`). The equations

$$y_i = c_0 + c_1 x_i$$

are given by the matrix equation

$$\begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}.$$

That is,

$$A\mathbf{c} = \mathbf{y}$$

where

$$\mathbf{c} = \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}.$$

This is an overdetermined system (n equations in 2 variables).

Example 9.3. Find the least squares fit for the data $(0, 0.485)$, $(0.4, 0.738)$, $(0.8, 1.360)$, $(1.2, 1.774)$, $(1.6, 2.162)$ and $(2.0, 1.764)$ (this is the data used to generate the above diagram).

Solution. Using MATLAB we have

```
>> x = [0 : 0.4 : 2]'

x =
0
0.4000
0.8000
1.2000
1.6000
2.0000

>> A = [ones(6, 1) x]

A =
1.0000      0
1.0000  0.4000
1.0000  0.8000
1.0000  1.2000
1.0000  1.6000
1.0000  2.0000

>> y = [0.48460.73841.35951.77402.16201.7640]'

y =
0.4868
0.7384
1.3595
1.7740
2.1620
1.7640

>> c = A\y

c =
0.5888
0.7916
```

Thus the best fit is

$$y = 0.5888 + 0.7916x.$$

Note that MATLAB's backslash operator, \backslash , will *automatically* compute the least squares solution for an overdetermined system. The residual for this fit is

```
>> r = y - A * c
```

```
r =
```

```
-0.1042
-0.1671
0.1374
0.2353
0.3066
-0.4080
```

```
>> norm(r)
```

```
ans =
```

```
0.6111
```

If we were unhappy with this fit or we suspected that the relationship between the variables x and y was quadratic

$$y = c_0 + c_1x + c_2x^2$$

(for example, if the data came from projectile motion) then we would solve the matrix equation

$$\begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \mathbf{y}.$$

Let

$$\mathbf{x}^2 = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \vdots \\ x_n^2 \end{bmatrix}$$

be the *component-wise* square of \mathbf{x} (note that \mathbf{x}^2 is *not* defined as a matrix product).

In MATLAB this is given by

$$\mathbf{x}.*\mathbf{x} \quad \text{or} \quad \mathbf{x.^2}$$

Therefore the matrix system may be written as

$$[\mathbf{1} \ \mathbf{x} \ \mathbf{x}^2] \mathbf{c} = \mathbf{y}.$$

Using the above data, we have

```
>> A = [A x.*x]
```

```
A =
```

```

1.0000      0      0
1.0000  0.4000  0.1600
1.0000  0.8000  0.6400
1.0000  1.2000  1.4400
1.0000  1.6000  2.5600
1.0000  2.0000  4.0000

```

>> $c = A \setminus y$

$c =$

```

0.3393
1.7272
-0.4678

```

>> $r = y - A * c$

$r =$

```

0.1453
-0.2170
-0.0622
0.0357
0.2567
-0.1585

```

>> $\text{norm}(r)$

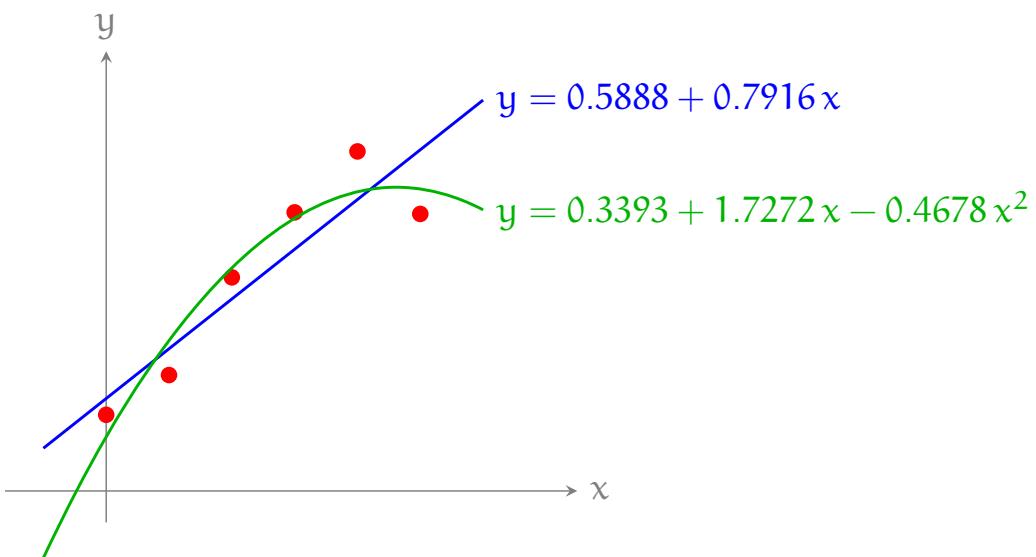
$\text{ans} =$

```
0.4054
```

The best quadratic fit is thus

$$y = 0.3393 + 1.7272x - 0.4678x^2$$

and the length of the residual has been reduced to 0.4054.



If we wish to fit a polynomial of degree k to a data set then we solve

$$[\mathbf{1} \quad \mathbf{x} \quad \mathbf{x}^2 \quad \cdots \quad \mathbf{x}^k] \mathbf{c} = \mathbf{y}.$$

Remember that the columns of the coefficient matrix are component-wise powers of the vector \mathbf{x} . We can extend this to fit any function to a data set. Suppose we wish to fit the data to a collection of functions $f_i(x)$; that is

$$y = c_1 f_1(x) + c_2 f_2(x) + \cdots + c_k f_k(x)$$

then we solve the system

$$[f_1(\mathbf{x}) \quad f_2(\mathbf{x}) \quad \cdots \quad f_k(\mathbf{x})] \mathbf{c} = \mathbf{y}.$$

Again these columns are the function $f_i(x)$ applied to each component of \mathbf{x} . This is the default behaviour of MATLAB.

In particular, if we wish to fit an exponential to a data set

$$y = ae^{bx}$$

we first take logarithms

$$\log y = \log a + bx$$

and then solve

$$[\mathbf{1} \quad \mathbf{x}] \begin{bmatrix} \log a \\ b \end{bmatrix} = \log \mathbf{y}.$$

Similarly to fit a power law to a data set

$$y = ax^k,$$

we take logarithms

$$\log y = \log a + k \log x$$

and solve

$$[\mathbf{1} \quad \log \mathbf{x}] \begin{bmatrix} \log a \\ k \end{bmatrix} = \log \mathbf{y}.$$

We have seen that the residual from a least squares fit $A\mathbf{c} = \mathbf{y}$ is

$$\mathbf{r} = \mathbf{y} - A\mathbf{c} = \text{perp}_{\text{col}(A)}(\mathbf{y}) \in (\text{col}(A))^\perp.$$

Therefore, for any column of A , \mathbf{a}_i , we have

$$0 = \mathbf{a}_i \cdot \mathbf{r} = \mathbf{a}_i^T \mathbf{r}.$$

Thus

$$\begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_n^T \end{bmatrix} \mathbf{r} = \mathbf{0},$$

or, in other words,

$$A^T \mathbf{r} = \mathbf{0}.$$

This is equivalent to

$$A^T A \mathbf{c} = A^T \mathbf{y},$$

the so-called *normal equations* of the overdetermined system $A\mathbf{c} = \mathbf{y}$. If $A = QR$ then

$$A^T A = R^T Q^T QR = R^T R$$

and so the normal equations become

$$R^T R \mathbf{c} = R^T Q^T \mathbf{y}.$$

If the columns of A are linearly independent then R is invertible and so we obtain

$$R \mathbf{c} = Q^T \mathbf{y}.$$

We have thus proven:

Theorem 9.4 (Least Squares Theorem). *Let A be an $n \times k$ matrix and $\mathbf{y} \in \mathbf{R}^n$. Then $A\mathbf{c} = \mathbf{y}$ always has a least squares solution $\bar{\mathbf{c}}$. Moreover $\bar{\mathbf{c}}$ is a least squares solution if and only if it is a solution of the normal equations*

$$A^T A \mathbf{c} = A^T \mathbf{y}.$$

In addition, A has linearly independent columns if and only if $A^T A$ is invertible. In this case the least squares solution is unique.

9.2 Pseudoinverse

For an overdetermined system $A\mathbf{x} = \mathbf{b}$ we have seen that the least squares solution is the solution of the normal equations

$$A^T A \mathbf{x} = A^T \mathbf{b}$$

and that if A has linearly independent columns then $A^T A$ is invertible. Thus, in this case, the solution is

$$\mathbf{x} = (A^T A)^{-1} A^T \mathbf{b}.$$

Note that A is, in general, not a square matrix and so we can *not* simplify this expression by writing

$$(A^T A)^{-1} = A^{-1} A^{-T}.$$

Normally would not want to compute an inverse. However if the rank of A is small (that is A has a small number of columns) then $A^T A$ will be a small square matrix. For example, in the least squares fit for a straight line, A has 2 columns and so $A^T A$ is a 2×2 matrix

$$A^T A = \begin{bmatrix} \mathbf{1}^T \\ \mathbf{x}^T \end{bmatrix} \begin{bmatrix} \mathbf{1} & \mathbf{x} \end{bmatrix}$$

which can be easily inverted. Motivated by this we have the following definition.

Definition 9.5 (Pseudoinverse). Let A be a matrix with linearly independent columns. Then the *pseudoinverse* of A is the matrix A^+ given by

$$A^+ = (A^T A)^{-1} A^T.$$

Note that if A is square then $A^+ = A^{-1}$. If Q has orthonormal columns then $Q^T Q = I$ and so

$$Q^+ = Q^T.$$

Example 9.6. Find the pseudoinverse of a (column) vector \mathbf{v} .

The solution to an overdetermined system $A\mathbf{x} = \mathbf{b}$ is

$$\mathbf{x} = A^+ \mathbf{b}.$$

We can also write the projection matrix in terms of the pseudoinverse. We have

Theorem 9.7. Let W be a subspace of \mathbf{R}^n and A be a $n \times m$ matrix whose columns are linearly independent and form a basis for W . For any $\mathbf{x} \in \mathbf{R}^n$

$$\text{proj}_W(\mathbf{x}) = AA^+ \mathbf{x}.$$

In particular, the projection matrix that projects \mathbf{R}^n onto W is given by

$$P = AA^+.$$

This theorem allows us to find a projection without computing an orthonormal basis. However the price to pay is that we must invert the matrix $A^T A$. For a projection onto a 1-dimensional subspace spanned by \mathbf{v} we have

$$P = \mathbf{v} \mathbf{v}^+ = \frac{1}{\|\mathbf{v}\|^2} \mathbf{v} \mathbf{v}^T = \mathbf{u} \mathbf{u}^T$$

where

$$\mathbf{u} = \frac{\mathbf{v}}{\|\mathbf{v}\|}$$

is a unit vector parallel to \mathbf{v} .

Theorem 9.8. Let A be a matrix with linearly independent columns. Then the pseudoinverse A^+ of A satisfies the Penrose conditions for A ; that is

- (a) $AA^+A = A$
- (b) $A^+AA^+ = A^+$
- (c) AA^+ and A^+A are symmetric.

Pseudoinverses can be generalised to matrices whose columns are not linearly independent.

10 Singular Values

Poole Ch. 7.3

10.1 Symmetric matrices

Poole Ch. 5.4

We have seen symmetric matrices (matrices that satisfy $A^T = A$) have featured in the discussion above. In fact symmetric matrices have some very nice properties. Consider the symmetric matrix

$$A = \begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix}.$$

It is easily seen that the eigenvalues are -3 and 2 with corresponding eigenvectors

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

Thus A is diagonalizable with $P = [\mathbf{x}_1 \ \mathbf{x}_2]$ and

$$P^{-1}AP = \begin{bmatrix} -3 & 0 \\ 0 & 2 \end{bmatrix} = D.$$

However, note that

$$\mathbf{x}_1 \cdot \mathbf{x}_2 = 0;$$

the eigenvectors are *orthogonal*. If we normalize the eigenvectors

$$\mathbf{u}_1 = \frac{\sqrt{5}}{5} \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \quad \mathbf{u}_2 = \frac{\sqrt{5}}{5} \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

we obtain an *orthogonal* matrix

$$Q = [\mathbf{u}_1 \ \mathbf{u}_2]$$

that diagonalizes A . Thus

$$A = QDQ^T$$

since $Q^{-1} = QT$. Thus we make the definition

Definition 10.1. A square matrix A is *orthogonally diagonalizable* if there exists an orthogonal matrix Q such that $Q^T A Q = D$, a diagonal matrix.

Theorem 10.2. *If A is orthogonally diagonalizable then A is symmetric.*

Proof. If A is orthogonally diagonalizable then there exists an orthogonal matrix Q such that $A = QDQ^T$. Now

$$AT = (QDQ^T)^T = QD^TQ^T = QDQ^T = A$$

since D is diagonal. Therefore A is symmetric. \square

This result does *not* show that all symmetric matrices are orthogonally diagonalizable (we are coming to that). Symmetric matrices have several nice properties.

Theorem 10.3. Let A be a real symmetric matrix. The eigenvalues of A are real.

Proof. Let λ be an eigenvalue of A with corresponding eigenvector \mathbf{x} . Therefore $A\mathbf{x} = \lambda\mathbf{x}$. Taking complex conjugates, we have

$$\overline{A}\bar{\mathbf{x}} = A\bar{\mathbf{x}} = \bar{\lambda}\bar{\mathbf{x}}$$

since A is real. Thus

$$\bar{\mathbf{x}}^T A^T = \bar{\mathbf{x}}^T A = \bar{\lambda} \bar{\mathbf{x}}^T$$

since A is symmetric. Now

$$\lambda (\bar{\mathbf{x}}^T \mathbf{x}) = \bar{\mathbf{x}}^T \lambda \mathbf{x} = \bar{\mathbf{x}}^T A \mathbf{x} = \bar{\lambda} (\bar{\mathbf{x}}^T \mathbf{x})$$

and so

$$(\lambda - \bar{\lambda}) \bar{\mathbf{x}}^T \mathbf{x} = 0.$$

Let $\mathbf{x} = \mathbf{a} + i\mathbf{b}$ with \mathbf{a} and \mathbf{b} real-valued. Now $\bar{\mathbf{x}} = \mathbf{a} - i\mathbf{b}$

$$\bar{\mathbf{x}}^T \mathbf{x} = (\mathbf{a}^T - i\mathbf{b}^T)(\mathbf{a} + i\mathbf{b}) = \mathbf{a}^T \mathbf{a} + \mathbf{b}^T \mathbf{b} + i(\mathbf{a}^T \mathbf{b} - \mathbf{b}^T \mathbf{a}).$$

However $\mathbf{b}^T \mathbf{a} = (\mathbf{a}^T \mathbf{b})^T = \mathbf{a}^T \mathbf{b}$ since $\mathbf{a}^T \mathbf{b}$ is a scalar and so

$$\bar{\mathbf{x}}^T \mathbf{x} = \mathbf{a}^T \mathbf{a} + \mathbf{b}^T \mathbf{b} \neq 0.$$

Therefore we conclude

$$\lambda - \bar{\lambda} = 0$$

and so λ is real. □

Theorem 10.4. Let A be a symmetric matrix. Then any two eigenvectors corresponding to distinct eigenvalues are orthogonal.

Proof. Suppose

$$A\mathbf{x}_1 = \lambda_1 \mathbf{x}_1, \quad A\mathbf{x}_2 = \lambda_2 \mathbf{x}_2$$

with $\lambda_1 \neq \lambda_2$. Now

$$\lambda_1 (\mathbf{x}_1 \cdot \mathbf{x}_2) = (A\mathbf{x}_1)^T \mathbf{x}_2 = \mathbf{x}_1^T A^T \mathbf{x}_2 = \mathbf{x}_1^T A \mathbf{x}_2$$

since A is symmetric. However

$$\mathbf{x}_1^T A \mathbf{x}_2 = \mathbf{x}_1^T (\lambda_2 \mathbf{x}_2) = \lambda_2 (\mathbf{x}_1 \cdot \mathbf{x}_2)$$

and so

$$(\lambda_1 - \lambda_2) \mathbf{x}_1 \cdot \mathbf{x}_2 = 0.$$

Since $\lambda_1 \neq \lambda_2$, we have

$$\mathbf{x}_1 \cdot \mathbf{x}_2 = 0$$

and so \mathbf{x}_1 and \mathbf{x}_2 are orthogonal. □

Theorem 10.5 (Spectral Theorem). Let A be a real $n \times n$ matrix. Then A is symmetric if and only if it is orthogonally diagonalizable.

Proof. See Poole Theorem 5.20. □

This theorem shows that every real symmetric matrix is not defective (that is the geometric multiplicity is always equal to the algebraic multiplicity for each of its eigenvalues).

Example 10.6. Orthogonally diagonalize the matrix

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}$$

Note that the spectral theorem implies

$$\begin{aligned} A &= QDQ^T = [\mathbf{q}_1 \ \mathbf{q}_2 \ \cdots \ \mathbf{q}_n] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_n^T \end{bmatrix} \\ &= [\lambda_1 \mathbf{q}_1 \ \lambda_2 \mathbf{q}_2 \ \cdots \ \lambda_n \mathbf{q}_n] \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_n^T \end{bmatrix} \\ &= \lambda_1 \mathbf{q}_1 \mathbf{q}_1^T + \lambda_2 \mathbf{q}_2 \mathbf{q}_2^T + \cdots + \lambda_n \mathbf{q}_n \mathbf{q}_n^T. \end{aligned}$$

This is called the *spectral decomposition* of A . Each of the terms $\lambda_i \mathbf{q}_i \mathbf{q}_i^T$ is a rank 1 matrix and $\mathbf{q}_i \mathbf{q}_i^T$ is the projection onto the subspace spanned by \mathbf{q}_i .

Example 10.7. Find the spectral decomposition for the matrix A from the previous example.

Observe that the spectral decomposition expresses a real symmetric matrix in terms of its eigenvalues and eigenvectors. We can use this to construct a symmetric matrix with given eigenvalues and (orthonormal) eigenvectors.

Example 10.8. Find a 2×2 matrix with eigenvalues $\lambda_1 = 13$ and $\lambda_2 = -13$ with corresponding eigenvectors

$$\mathbf{x}_1 = \begin{bmatrix} 12 \\ 5 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} -5 \\ 12 \end{bmatrix}.$$

10.2 Singular values

We have seen that real symmetric matrices, A , have a (very) nice representation via the spectral theorem

$$A = \lambda_1 \mathbf{q}_1 \mathbf{q}_1^T + \lambda_2 \mathbf{q}_2 \mathbf{q}_2^T + \cdots + \lambda_n \mathbf{q}_n \mathbf{q}_n^T.$$

Is there any similar form for a non-symmetric or non-square matrices? For *any* $k \times n$ matrix A , the $n \times n$ matrix $A^T A$ is symmetric. Suppose μ is an eigenvalue of $A^T A$ and \mathbf{v} is an associated *unit* eigenvector; that is

$$A^T A \mathbf{v} = \mu \mathbf{v}.$$

Now

$$0 \leq \|A\mathbf{v}\|^2 = (A\mathbf{v})^T A\mathbf{v} = \mathbf{v}^T A^T A \mathbf{v} = \mu \mathbf{v}^T \mathbf{v} = \mu \|\mathbf{v}\|^2 = \mu$$

since \mathbf{v} is a unit vector. Therefore all the eigenvalues of $A^T A$ are not only real but also *non-negative*. It therefore makes sense to take the positive square root of this eigenvalues.

Definition 10.9. Let A be a real $k \times n$ matrix. The *singular values* of A are the (positive) square roots of the eigenvalues of $A^T A$ and are denoted by $\sigma_1, \sigma_2, \dots, \sigma_n$. It is conventional to arrange the singular values so that $\sigma_1 > \sigma_2 > \dots > \sigma_n$.

Singular values are, by construction, non-negative. For a symmetric matrix A , $A^T A = A^2$ and so

$$\mu_i = \lambda_i^2$$

where λ_i are the eigenvalues of A . Therefore the singular values of a symmetric matrix are

$$\sigma_i = \sqrt{\lambda_i^2} = |\lambda_i|$$

(note that the eigenvalues of A are real but may be negative).

Example 10.10. Find the singular value(s) of

$$\mathbf{v} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{w} = [1 \ 1 \ 1]$$

Since $A^T A$ is symmetric, there is an *orthonormal* basis for \mathbf{R}^n consisting of eigenvectors of $A^T A$. Let this basis be given by $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ where we order the eigenvalues of $A^T A$ so that $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n$. We have observed earlier that

$$\mu_i = \|A\mathbf{v}_i\|^2$$

and so

$$\sigma_i = \sqrt{\mu_i} = \|A\mathbf{v}_i\|.$$

Now, for $i \neq j$,

$$(A\mathbf{v}_i) \cdot (A\mathbf{v}_j) = (A\mathbf{v}_i)^T A\mathbf{v}_j = \mathbf{v}_i^T A^T A \mathbf{v}_j = \mu_j (\mathbf{v}_i \cdot \mathbf{v}_j) = 0$$

since the eigenvectors \mathbf{v}_i are orthogonal. Suppose the *non-zero* singular values of A are

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$$

with $\sigma_{r+1} = \sigma_{r+2} = \dots = \sigma_n = 0$. Since $\sigma_i = \|A\mathbf{v}_i\|$,

$$\mathbf{u}_i = \frac{1}{\sigma_i} A\mathbf{v}_i$$

for $i = 1, 2, \dots, r$ will be *unit* vectors which are mutually orthogonal; that is, they form an orthonormal set. Furthermore $\|A\mathbf{v}_i\| = \sigma_i = 0$; that is

$$A\mathbf{v}_i = \mathbf{0}$$

for $i = r+1, r+2, \dots, n$. Therefore

$$A [\mathbf{v}_1 \ \cdots \ \mathbf{v}_r \ \mathbf{v}_{r+1} \ \cdots \ \mathbf{v}_n] = [\sigma_1 \mathbf{u}_1 \ \cdots \ \sigma_r \mathbf{u}_r \ 0 \ \cdots \ 0].$$

Since $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ is an orthonormal basis for \mathbf{R}^n ,

$$V = [\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_n]$$

is an *orthogonal* $n \times n$ matrix. Note that $\mathbf{u}_i \in \mathbf{R}^k$. If $r < k$ we use Gram Schmidt to construct an orthonormal basis for \mathbf{R}^k that includes \mathbf{u}_i ; that is, we add $\mathbf{u}_j, j = r+1, r+2, \dots, k$ such that

$$U = [\mathbf{u}_1 \ \mathbf{u}_2 \ \cdots \ \mathbf{u}_r \ \mathbf{u}_{r+1} \ \cdots \ \mathbf{u}_k]$$

is an *orthogonal* $k \times k$ matrix.

We now have

$$AV = U\Sigma$$

where Σ is the $k \times n$ “diagonal” matrix

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} D & O \\ O & O \end{bmatrix}$$

with

$$D = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r \end{bmatrix}.$$

Note that Σ is the same size as A . Since V is orthogonal, we have

$$A = U\Sigma V^T.$$

This is called the *singular value decomposition* (SVD) of A . The columns of U are called the *left singular vectors* and the columns of V are called the *right singular vectors*. Formally we have:

Theorem 10.11 (Singular Value Decomposition). *Let A be a real $k \times n$ matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0$ and $\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$. Let Σ be the $k \times n$ matrix*

$$\Sigma = \begin{bmatrix} D & O \\ O & O \end{bmatrix}$$

with

$$D = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r \end{bmatrix}.$$

Then there exists an orthogonal $n \times n$ matrix V and an orthogonal $k \times k$ matrix U such that

$$A = U\Sigma V^T.$$

The matrix Σ is fixed but the matrices U and V are *not* unique.

For a symmetric matrix A , the SVD reduces to an orthogonal diagonalization. Recall that the singular values of A are

$$\sigma_i = |\lambda_i|$$

where λ_i are the eigenvalues of A . The eigenvectors \mathbf{v}_i of $A^T A = A^2$ are the eigenvectors \mathbf{q}_i of A and so

$$V = Q.$$

Furthermore

$$\mathbf{u}_i = \frac{1}{|\lambda_i|} A \mathbf{q}_i = \frac{1}{|\lambda_i|} \lambda_i \mathbf{q}_i = \text{sign}(\lambda_i) \mathbf{q}_i.$$

Therefore

$$\begin{aligned} A &= U\Sigma V^T = [\text{sign}(\lambda_1) \mathbf{q}_1 \quad \text{sign}(\lambda_2) \mathbf{q}_2 \quad \cdots \quad \text{sign}(\lambda_n) \mathbf{q}_n] \begin{bmatrix} |\lambda_1| & 0 & \cdots & 0 \\ 0 & |\lambda_2| & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & |\lambda_n| \end{bmatrix} Q^T \\ &= Q \begin{bmatrix} \text{sign}(\lambda_1) |\lambda_1| & 0 & \cdots & 0 \\ 0 & \text{sign}(\lambda_2) |\lambda_2| & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \text{sign}(\lambda_n) |\lambda_n| \end{bmatrix} Q^T \\ &= Q \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} Q^T. \end{aligned}$$

Example 10.12. Find a SVD for

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

Example 10.13. Find a SVD for

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

There is a form of the SVD which is analogous to the spectral decomposition of a symmetric matrix. We have

$$\begin{aligned} A &= U\Sigma V^T = [\mathbf{u}_1 \ \cdots \ \mathbf{u}_k] \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots & O \\ 0 & \cdots & \sigma_r & \\ O & & & O \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_n^T \end{bmatrix} \\ &= [\mathbf{u}_1 \ \cdots \ \mathbf{u}_k] \begin{bmatrix} \sigma_1 \mathbf{v}_1^T \\ \vdots \\ \sigma_r \mathbf{v}_r^T \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \cdots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T. \end{aligned}$$

Theorem 10.14 (Outer Product Form of the SVD). *Let A be a real $k \times n$ matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ and $\sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$. Let $\mathbf{u}_1, \dots, \mathbf{u}_r$ be the left singular vectors and $\mathbf{v}_1, \dots, \mathbf{v}_r$ be the right singular vectors corresponding to the non-zero singular values. Then*

$$A = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \cdots + \sigma_r \mathbf{u}_r \mathbf{v}_r^T.$$

Note that, in this form, there is no need to compute the singular vectors corresponding to the zero singular values. However the singular vectors span the fundamental subspaces of A .

Theorem 10.15. *Let $A = U\Sigma V^T$ be a singular value decomposition of a $k \times n$ matrix A . Let $\sigma_1, \dots, \sigma_r$ be the non-zero singular values of A . Then*

- (a) *The rank of A is r .*
- (b) *$\{\mathbf{u}_1, \dots, \mathbf{u}_r\}$ is an orthonormal basis for $\text{col}(A)$.*
- (c) *$\{\mathbf{u}_{r+1}, \dots, \mathbf{u}_k\}$ is an orthonormal basis for $\text{null}(A^T)$.*
- (d) *$\{\mathbf{v}_1, \dots, \mathbf{v}_r\}$ is an orthonormal basis for $\text{row}(A)$.*
- (e) *$\{\mathbf{v}_{r+1}, \dots, \mathbf{v}_n\}$ is an orthonormal basis for $\text{null}(A)$.*

Proof. Poole Theorem 7.15. □

In particular, this theorem implies that an $n \times n$ matrix is invertible if and only if it has no zero singular values—that is, $r = n$.

10.3 Compression

The other product form of an SVD implies that we only need to store r vectors $\mathbf{u}_i \in \mathbf{R}^k$, r vectors $\mathbf{v}_i \in \mathbf{R}^n$ and the r non-zero singular values σ_i in order to reconstruct the matrix A . In total, there are

$$rk + rn + r = r(k + n + 1)$$

numbers that we need to store compare to kn numbers that would be needed if we stored A directly. In fact we can do slightly better. If we store $\sigma_i \mathbf{u}_i = A \mathbf{v}_i$ then the storage is reduced to

$$r(k + n).$$

If r is small (that is, the rank is small) then

$$r(k + n) \ll kn$$

and we have achieved *loss-less compression*. In particular, for a square $n \times n$ matrix, we require

$$r < \frac{1}{2}n$$

in order to achieve loss-less compression.

In the case of loss-less compression, we can reconstruct A *exactly*. However if r is large, we might want to *approximate* A by a matrix of *smaller rank*. This will then achieve *lossy compression* since we will not be able to reconstruct A exactly.

Each term

$$\sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

in the SVD is a *rank 1* matrix. Since we have ordered the singular values in descending size, we call

$$A_1 = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T$$

the *rank 1 approximation to A* . Similarly, for $s < r$, we call

$$A_s = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \cdots + \sigma_s \mathbf{u}_s \mathbf{v}_s^T$$

the *rank s approximation to A* .

In typical applications (say image compression), there is a fair range of singular values. By ignoring the small non-zero singular values, we achieve a lower rank approximation to A that preserves most of the information contained in A (remember that the terms that we are ignoring will be small compared to the terms that we are keeping). The art (or engineering!) is to choose s as small as possible (to maximize the compression) but not too small so that we lose essential detail in A .



MATLAB

MATLAB has the function `svd` to compute the singular decomposition of a matrix A .

- `s = svd(A)` (or, simply, `svd(A)`) will return a *vector* of the singular values of A (in order of descending size).
- `[U, S, V] = svd(A)` will return the three factors of a SVD for A .

Consider the matrices

$$A = \begin{bmatrix} 8.1650 & -0.0041 & -0.0041 \\ 4.0825 & -3.9959 & 4.0042 \\ 4.0825 & 4.0042 & -3.9959 \end{bmatrix}, \quad B = \begin{bmatrix} 8.17 & 0 & 0 \\ 4.08 & -4.00 & 4.00 \\ 4.08 & 4.00 & -4.00 \end{bmatrix}$$

where B has been obtained from A by rounding to two decimal places. MATLAB gives the singular values

```
>> [svd(A), svd(B)]  
ans =  
10.0000 10.0000  
8.0000 8.0000  
0.0100 0.0000
```

We see that A has rank 3 and so is invertible but B has rank 2 and thus is not invertible. In applications, it is often assumed that small singular values are the result of round-off error and that the actual value should be zero. In this way “noise” can be filtered out. A SVD for A is

```
>> [U, S, V] = svd(A)  
U =  
-0.8165 -0.0000 -0.5774  
-0.4082 -0.7071 0.5774  
-0.4082 0.7071 0.5774  
S =  
10.0000 0 0  
0 8.0000 0  
0 0 0.0100  
V =  
-1.0000 0 0  
0 0.7071 0.7071  
0 -0.7071 0.7071
```

If we assume that the singular value 0.01 is the result of round-off error then

```
>> U * diag([10, 8, 0]) * V'  
ans =
```

$$\begin{matrix} 8.1650 & -0.0000 & 0.0000 \\ 4.0825 & -4.0000 & 4.0000 \\ 4.0825 & 4.0000 & -4.0000 \end{matrix}$$

that is,

$$U \begin{bmatrix} 10 & 0 & 0 \\ 0 & 8 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T = B.$$

Theorem 10.16. Let A be an invertible matrix with a singular value decomposition $A = U\Sigma V^T$. Then A^{-1} has a singular value decomposition

$$A^{-1} = V\Sigma^{-1}U^T.$$

Proof. Exercise. □

Note that this theorem implies that the singular values of A^{-1} are

$$\frac{1}{\sigma_n} \geq \frac{1}{\sigma_{n-1}} \geq \cdots \geq \frac{1}{\sigma_1}.$$

Condition Number Revisited

Recall that

$$\|A\|_2 = \max_{\|\mathbf{x}\|=1} \|A\mathbf{x}\|.$$

Let $A = U\Sigma V^T$ be a SVD for A . Since U is orthogonal, we have $\|U\mathbf{y}\| = \|\mathbf{y}\|$ for any $\mathbf{y} \in \mathbf{R}^k$ and so

$$\|A\mathbf{x}\| = \|U\Sigma V^T \mathbf{x}\| = \|\Sigma V^T \mathbf{x}\|$$

(set $\mathbf{y} = \Sigma V^T \mathbf{x}$). Since V (and therefore V^T) is orthogonal, for $\|\mathbf{x}\| = 1$ we have

$$\|V^T \mathbf{x}\| = \|\mathbf{x}\| = 1.$$

Therefore, setting $\mathbf{y} = V^T \mathbf{x}$ we have

$$\|A\|_2 = \max_{\|\mathbf{x}\|=1} \|A\mathbf{x}\| = \max_{\|\mathbf{y}\|=1} \|\Sigma \mathbf{y}\| = \sigma_1.$$

From the above theorem we also note that, if A is invertible,

$$\|A^{-1}\|_2 = \frac{1}{\sigma_n}.$$

Therefore the condition number (using the 2-norm) for a matrix is

$$\text{cond}_2(A) \equiv \|A^{-1}\|_2 \|A\|_2 = \frac{\sigma_1}{\sigma_n}.$$

For the matrix A in the above example, we have

$$\text{cond}_2(A) = \frac{10}{0.01} = 1000.$$

Pseudoinverse Revisited

We have defined the pseudoinverse for a matrix A that has linearly independent columns. We can now extend this definition to any matrix; in particular, to non-invertible square matrices.

Definition 10.17. Let $A = U\Sigma V^T$ be a SVD of a $k \times n$ matrix A with

$$\Sigma = \begin{bmatrix} D & O \\ O & O \end{bmatrix}$$

and D is the $r \times r$ diagonal matrix with the non-zero singular values $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$ of A on its diagonal. The *pseudoinverse* or *Moore-Penrose inverse* of A is the $n \times k$ matrix

$$A^+ = V\Sigma^+U^T$$

where Σ^+ is the $n \times k$ matrix

$$\Sigma^+ = \begin{bmatrix} D^{-1} & O^T \\ O^T & O^T \end{bmatrix}.$$

This new definition reduces to the old one in the case of a $k \times n$ matrix A with linearly independent columns. Suppose $A = U\Sigma V^T$ is a SVD for A . Then

$$A^T A = V\Sigma^T U^T U\Sigma V^T = V(\Sigma^T \Sigma) V^T$$

since U is orthogonal. Since A has linearly independent columns, it has no non-zero singular values ($r = n$) and so Σ is the $k \times n$ matrix

$$\Sigma = \begin{bmatrix} D \\ O \end{bmatrix}$$

and so

$$\Sigma^T \Sigma = [D \ O] \begin{bmatrix} D \\ O \end{bmatrix} = D^2.$$

Therefore

$$(A^T A)^{-1} = (V D^2 V^T)^{-1} = V^{-T} D^{-2} V^{-1}.$$

Since V is orthogonal,

$$\begin{aligned} (A^T A)^{-1} A^T &= V D^{-2} V^T V \Sigma^T U^T \\ &= V D^{-2} [D \ O] U^T \\ &= V [D^{-1} \ O] U^T \\ &= V \Sigma^+ U^T. \end{aligned}$$

Example 10.18. Find the pseudoinverse of

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

Example 10.19. Find the pseudoinverse of

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