

Seeing the Shape

A Geometric Introduction to Multivariate Quantitative Genetics

Daniel Ortiz-Barrientos

School of the Environment
The University of Queensland

ARC Centre of Excellence for Plant Success
in Nature and Agriculture

December 6, 2025

The shape of the ellipsoid and the direction of the arrow—these two things, together, determine what will happen. The G matrix is potential; selection is actuality. Their interaction is evolution.

Seeing the Shape: A Geometric Introduction to Multivariate Quantitative Genetics

© 2025 Daniel Ortiz-Barrientos

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0). You are free to share and adapt this material for non-commercial purposes, provided you give appropriate credit and distribute any derivative works under the same license.

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

School of the Environment
The University of Queensland
Brisbane, Queensland, Australia

<https://www.ortizbarrientoslabs.org>

First edition: December 2025

Typeset in Palatino using L^AT_EX.

Preface

These notes began as a companion to my reading of Mark Blows' paper "*A tale of two matrices*." They grew while teaching Biostatistics at The University of Queensland, where I found that students comfortable with regression, *t*-tests, and ANOVA would lose their footing as soon as covariance matrices, eigenvalues, and diagonalisation appeared. The notes have expanded through conversations with Nicholas O'Brien, Mark Blows, Mark Cooper, Jan Engesltaedter, Pamela Burrage, and Kevin Burrage.

The aim here is to build a different path into that material. We start from distance, which is intuitive, and we show how the standard tools of multivariate evolutionary biology—phenotypic and genetic covariance matrices, Mahalanobis distance, PCA, canonical analyses, and the matrices underlying multivariate selection—are all expressions of the same geometric story.

By the end of these notes you should be able to

- interpret eigenvectors and eigenvalues geometrically and biologically;
- perform diagonalisation in simple cases and understand each step;
- decide when diagonalisation is the right tool for a given biological question.

The guiding principle is simple: *symmetric matrices describe shapes*. The algebra is a precise language for those shapes. Whenever the symbols become opaque, the right move is to go back to the picture and draw the ellipse.

These are living lecture notes. They will change. Feedback is welcome.

Intended audience and prerequisites

This book is written for biologists who are already comfortable with basic statistics and are willing to learn some linear algebra along the way. A good starting point is the level of a typical second- or third-year biostatistics course: means and variances, covariance and correlation, simple linear regression, *t*-tests, and ANOVA.

You do *not* need a full course in linear algebra before starting, but you will get more from the later chapters if you have met vectors, matrices, and the idea of an eigenvalue at least once. The appendix *Mathematical and Statistical Background* collects the minimum machinery assumed in the main text: vectors and matrices as things you can multiply, eigenvalues and eigenvectors for symmetric matrices, basic probability language (variance, covariance, multivariate normal), and the definition of selection gradients and Lande's equation. Readers who

feel rusty on any of these topics are encouraged to skim that appendix early and return to it as needed.

The book is designed to be readable in two passes. On a first pass, you can focus on the geometric story: pictures of trait space, ellipses for covariance, the idea of whitening, and the shapes defined by \mathbf{G} , \mathbf{P} , and γ . On a second pass, you can pay closer attention to the algebra, work through the derivations in detail, and use the worked examples and code in the later chapters to practise complete analyses from data to interpretation.

How this book is organised

This book is written for biologists who want to understand the geometry behind quantitative genetics. It is organised in four parts that build on each other: we start with pictures of trait space, move to distance and covariance, then to natural axes and whitening, and finally to genetic and fitness objects such as the \mathbf{G} matrix and curved fitness surfaces.:contentReference[oaicite:1]index=1

Part I: Geometry of trait space

The first part introduces the basic geometric language.

In **Chapter 1: Points and Trait Space** we learn to see individual phenotypes as points in a trait space, samples as clouds of points, and trait differences as arrows between points.

In **Chapter 2: Vectors, Coordinates, and Angles** we put coordinates on this space, introduce vectors as directed differences, and use the dot product to talk about lengths and angles between trait combinations.

In **Chapter 3: Matrices as Machines That Move Vectors** we treat matrices as machines that move vectors, and we see how simple matrices can stretch, rotate, and shear trait space in ways that we will later use to describe variance, covariance, and selection.

Part II: Distance, variance, and covariance

The second part explains why distance matters for variation and why simple Euclidean distance is sometimes misleading.

In **Chapter 4: Distance and Why We Square It** we link straight-line distance to Pythagoras' theorem, show how squaring distance leads naturally to variance as "average squared distance from the mean", and connect this idea to the familiar one-dimensional formulas used in statistics.

In **Chapter 5: When Euclidean Distance Fails** we see concrete examples where Euclidean distance ignores scale differences between traits, misses correlations between traits, and does not match the probability structure of our data, motivating the need for a more flexible metric.

In **Chapter 6: Covariance and Mahalanobis Distance** we introduce the covariance matrix as a shape that summarises how traits vary together and define Mahalanobis distance by placing this matrix between two vectors, leading to ellipses (and ellipsoids) that match the spread and correlation of the data.

Part III: Natural axes, diagonalisation, and whitening

The third part shows how to find natural axes of variation and how to rescale trait space so

that phenotypic variance looks spherical.

In **Chapter 7: Diagonalisation and Natural Axes** we introduce eigenvalues and eigenvectors as directions in which a matrix only stretches and does not rotate, and we relate these natural axes to principal components and to the shape of covariance ellipses.

In **Chapter 8: Whitening and the P-sphere** we transform trait space using the phenotypic covariance matrix \mathbf{P} so that phenotypic variation becomes a sphere (the P-sphere), and in this whitened space we see how a transformed genetic matrix \mathbf{G}^* encodes directional heritability along different directions of trait change.

Part IV: Genetic and fitness geometry

The final part applies these geometric tools to genetic and fitness objects.

In **Chapter 9: The G Matrix and the Genetic Ellipsoid** we describe the additive genetic covariance matrix \mathbf{G} as an ellipsoid that channels evolutionary change, introduce key quantities such as the leading eigenvector (often called g_{\max}), and connect this geometric view to evolvability, constraint, and effective dimensionality.

In **Chapter 10: The Fitness Surface and γ** we represent local fitness surfaces as curved (paraboloid) shapes over trait space, introduce the curvature matrix γ , and explain how curvature interacts with genetic variation to influence the paths and limits of evolutionary change.

In **Chapter 11: PCA, MANOVA, and Projections** we connect the geometric picture to standard multivariate methods, showing how eigenstructure underlies PCA, MANOVA, discriminant analysis, and related projection techniques.

Part V: Practice and extensions

In **Chapter 12: Worked Examples: Complete Analyses** we walk through complete analyses from raw data to biological interpretation, combining \mathbf{G} , \mathbf{P} , β , and γ .

In **Chapter 13: Directional Heritability and the Geometry of Constraint** we develop the distributional view of directional heritability, link it to the eigenstructure of \mathbf{G}^* , and discuss constraint heterogeneity and its implications for evolution and breeding.

A final epilogue reflects on the geometric perspective, and the back matter collects references, the mathematical appendix, and hints for selected exercises.

Code companion and reproducibility

All code used to generate figures and worked examples is available in a companion repository:

<https://github.com/dortizbarrientos/seetheshape>

The repository contains parallel python/ and R/ directories with annotated scripts organised by chapter, together with figure-generation scripts in figures/. The naming convention matches the chapter numbering in this book:

- Chapter 1 (*Points and Trait Space*) corresponds to files such as python/ch01_points_trait_space.py and R/ch01_points_trait_space.R.

- Chapter 2 (*Vectors, Coordinates, and Angles*) to `ch02_vectors_coordinates.*`.
- ...
- Chapter 13 (*Directional Heritability and the Geometry of Constraint*) to `ch13_directional_heritability.*`

Each script mirrors the structure of the corresponding chapter: it implements the algebra, reproduces the main figures, and includes additional comments and small exercises. The top-level `README.md` in the repository summarises the directory layout and lists the main functions provided for each chapter.

Readers who prefer to learn by doing are encouraged to keep the book and the code side by side: read a section, run the matching code, and adjust parameters or trait combinations to see how the geometry changes.

Notation used in this book

This book uses a small, consistent set of symbols. The aim is to reduce the need to hunt through previous chapters when you forget what a symbol means. Roughly speaking, plain italics such as a or x denote single numbers (scalars), bold symbols such as \mathbf{x} denote vectors, and bold capitals such as \mathbf{A} denote matrices. The commands `\vect{}` and `\mat{}` in the source simply typeset these in bold.

This table does not try to list every symbol used in a specific example. Its role is to give you a quick reminder of the main objects that appear throughout the book: vectors and matrices for traits, the key covariance matrices \mathbf{P} and \mathbf{G} , and the quantities used to describe selection, response, and fitness.

Symbol	Meaning
a, b, c	Single numbers (scalars)
x, y, z	Scalar trait values or coordinates
i, j, k	Indices for individuals or traits
n	Number of individuals in a sample
p	Number of traits
\mathbf{x}	Column vector of trait values for one individual
\mathbf{x}_i	Phenotype vector of individual i
$\bar{\mathbf{x}}$	Mean phenotype vector in a sample or population
$\mathbf{0}$	Zero vector (all components equal to 0)
\mathbf{e}_i	Unit vector along trait i (1 in position i , 0 elsewhere)
\mathbf{A}, \mathbf{B}	General matrices (linear transformations)
\mathbf{I}	Identity matrix (leaves every vector unchanged)
Σ	Generic covariance matrix
\mathbf{P}	Phenotypic covariance matrix
\mathbf{G}	Additive genetic covariance matrix
\mathbf{G}^*	Whitened genetic matrix: $\mathbf{P}^{-1/2}\mathbf{G}\mathbf{P}^{-1/2}$
Λ	Diagonal matrix of eigenvalues
\mathbf{Q}	Matrix whose columns are eigenvectors
\mathbf{x}^\top	Transpose of \mathbf{x} (row vector)
$\langle \mathbf{x}, \mathbf{y} \rangle$	Inner (dot) product of two vectors
$\ \mathbf{x}\ $	Length (Euclidean norm) of vector \mathbf{x}
$\mathbf{x}^\top \mathbf{A} \mathbf{x}$	Quadratic form defined by \mathbf{A}
λ_i	i th eigenvalue of a matrix (stretch along \mathbf{v}_i)
\mathbf{v}_i	i th eigenvector (direction associated with λ_i)
X, Y	Scalar random variables
$\mathbb{E}[X]$	Expectation (mean) of X
$\text{Var}(X)$	Variance of X
$\text{Cov}(X, Y)$	Covariance between X and Y
$\mathcal{N}(\boldsymbol{\mu}, \Sigma)$	Multivariate normal with mean $\boldsymbol{\mu}$ and covariance Σ
\mathbf{f}	Vector of directional selection gradients
\mathbf{s}	Vector of selection differentials
$\Delta \bar{\mathbf{z}}$	Change in mean phenotype under selection
h^2	Heritability of a single trait
$h^2(\mathbf{u})$	Directional heritability along direction \mathbf{u}
$e(\mathbf{u})$	Evolvability along direction \mathbf{u} (additive variance in that direction)
w	Fitness of an individual
\bar{w}	Mean fitness in the population
γ	Matrix of quadratic selection gradients (curvature of the fitness surface)

Contents

I Arrows, Directions, and Tables of Numbers	1
1 Points and Trait Space	2
1.1 Traits as axes, individuals as points	2
1.2 Differences between individuals as arrows	3
1.3 The mean as natural reference	4
1.4 Adding and stretching arrows	4
1.5 From arrows to vectors	5
1.6 Distances and lengths of vectors	7
1.7 Summary	7
2 Vectors, Coordinates, and Angles	9
2.1 Column vectors and coordinates	9
2.2 Unit vectors and decomposing changes	10
2.3 Lengths written in vector notation	10
2.4 The dot product and angles	11
2.5 Projections onto a direction	11
2.6 Distances between phenotypes	11
2.7 Summary	12
3 Matrices as Machines That Move Vectors	15
3.1 A motivating example: scaling traits differently	15
3.2 Matrix–vector multiplication	16
3.3 What happens to the unit vectors?	16
3.4 Geometric vocabulary: stretch, rotate, shear	17
3.5 Linearity: the defining property	17
3.6 Composing transformations: matrix multiplication	19
3.7 The identity and inverse	20

3.8 Symmetric matrices: a special and important case	20
3.9 Preview: the quadratic form	21
3.10 A biological example: the G matrix as a transformation	21
3.11 Summary	22
II Distance and Shape	25
4 Distance and Why We Square It	26
4.1 Why distance matters	26
4.2 One trait: distance on a line	27
4.3 Two traits: the Pythagorean formula	27
4.4 Why do we square?	28
4.5 From individual distances to population spread	29
4.6 Variance in two traits: the covariance matrix appears	31
4.7 A worked example	31
4.8 The covariance matrix as a shape	32
4.9 What squared distance assumes	32
4.10 Summary	33
5 When Euclidean distance Distance Fails	36
5.1 Example 1: The problem of scale	36
5.2 Example 2: The problem of correlation	38
5.3 Example 3: The probability perspective	39
5.4 A biological interlude: why this matters for selection	41
5.5 What we need from a better metric	41
5.6 A geometric preview	42
5.7 A worked comparison	42
5.8 The connection to standardisation	43
5.9 Summary	44
6 Covariance and Mahalanobis Distance	47
6.1 The key insight: a matrix between the vectors	47
6.2 The covariance matrix and its inverse	48
6.3 A one-dimensional sanity check	48
6.4 Geometry: how the inverse reshapes space	49

6.5	The formula in components	49
6.6	Mahalanobis distance distance and probability	50
6.7	A worked example	51
6.8	The Mahalanobis distance as a transformation	52
6.9	Mahalanobis distance distance between two points	53
6.10	Connection to discriminant analysis	54
6.11	Biological interpretation	54
6.12	What Mahalanobis distance requires	55
6.13	Summary	55
III	Diagonalisation and Natural Axes	58
7	Diagonalisation and Natural Axes	59
7.1	The question that leads to eigenvalues	59
7.2	A concrete example	60
7.3	The spectral theorem: why symmetric matrices are special	61
7.4	diagonalisation: the matrix factorisation	61
7.5	Geometric interpretation: the ellipse revealed	62
7.6	Why diagonalisation simplifies everything	63
7.7	The trace and determinant as summaries	64
7.8	Variance in any direction: the quadratic form revisited	65
7.9	Principal Component Analysis (PCA) in one paragraph	65
7.10	Computing eigenvalues and eigenvectors in practice	66
7.11	A biological example: the G matrix	66
7.12	positive definiteness and what eigenvalues tell us	67
7.13	The condition number: how “ill-behaved” is the matrix?	68
7.14	Summary	68
8	Whitening and the P-sphere	71
8.1	The problem: comparing G and P	71
8.2	The naive approach and its problem	72
8.3	The P-sphere: uniform with respect to phenotype	72
8.4	The whitening transformation	73
8.5	A remarkable fact: eigenvalues of G* are directional heritabilities	73

8.6 The distribution of directional heritability	74
8.7 Constraint traps	75
8.8 Visualising G inside P	75
8.9 Connection to the breeder's equation equation	76
8.10 Computing G^* in practice	76
8.11 Why whitening matters	78
8.12 Summary	78
IV Evolutionary Applications	79
9 The G Matrix and the Genetic Ellipsoid	80
9.1 What the G matrix represents	80
9.2 The genetic ellipsoid	80
9.3 g_{\max} : the line of least evolutionary resistance	81
9.4 The multivariate breeder's equation equation revisited	82
9.5 Evolvability: genetic variance in the direction of selection	82
9.6 Respondability and the comparison with P	85
9.7 Constraint: when G limits evolution	85
9.8 Effective dimensionality	86
9.9 Empirical G matrices: what do they look like?	86
9.10 Stability and estimation of G	86
9.11 G in the context of P	87
9.12 Summary	87
10 The Fitness Surface and Gamma	89
10.1 Fitness as a surface over trait space	89
10.2 The selection gradient beta	89
10.3 Beyond slopes: the curvature matrix gamma	91
91section.10.4	
10.5 Correlational selection: the off-diagonal elements	92
10.6 Quadratic fitness functions	92
10.7 Estimating gamma from data	93
10.8 The geometry of gamma	94
10.9 Comparing gamma and G: alignment matters	95

10.10	Gamma and the maintenance of genetic variation	95
10.11	A worked example	95
10.12	Canonical analysis of the fitness surface	96
10.13	Summary	97
11	PCA, MANOVA, and Projections	98
11.1	The bridge from geometry to statistics	98
11.2	Principal Component Analysis (PCA)	99
11.3	MANOVA: comparing groups	100
11.4	discriminant analysis	101
11.5	Canonical correlationAnalysis (CCA)	102
11.6	Projection pursuit: the general principle	103
11.7	Comparing G matrices	104
11.8	Estimation issues	105
11.9	Visualising high-dimensional patterns	106
11.10	Worked example: PCA of a phenotypic dataset	106
11.11	Summary	107
V	Practice and Extensions	109
12	Worked Examples: Complete Analyses	110
12.1	Example 1: Two-trait G matrix analysis	110
12.2	Example 2: Four-trait G-P comparison with whitening	114
12.3	Example 3: Selection analysis with G and gamma	116
12.4	Computational tools	119
12.5	Summary	120
13	directional heritability and the Geometry of Constraint	122
13.1	From eigenvalues to distributions	122
13.2	The coefficient of variation of directional heritability	123
124	section.13.3	
13.4	Constraint traps revisited	125
13.5	Selection strength and constraint risk	125
13.6	Empirical estimation	126
13.7	Case study: empirical G-P analysis	127

13.8 Implications for breeding and evolution	128
13.9 Extensions and open questions	129
13.10 Summary	129
Epilogue: The Shape of Things	131
References	134
Appendix: Mathematical and Statistical Background	135
13.11 Vectors and matrices	135
13.12 eigenvalues and eigenvectors	137
13.13 Basic probability and covariance	139
13.14 Selection gradients and Lande's equation	140
Hints and Selected Solutions	143

Part I

Arrows, Directions, and Tables of Numbers

Chapter 1

Points and Trait Space

These notes will use pictures of points, arrows, and clouds in space throughout. Before we mention vectors or matrices, we need a clear picture of what the space is and what it means to move around in it.

1.1 Traits as axes, individuals as points

Start with a single quantitative trait, such as body length. We can draw a line and mark each individual on that line according to its measured value. This is the usual number line from school (Fig. 1.1).

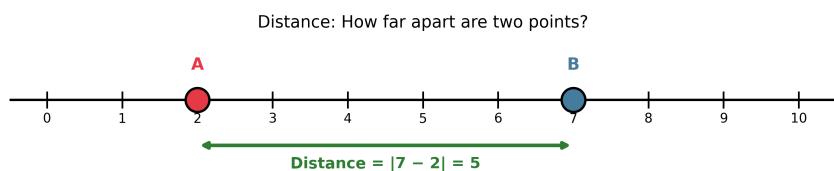


Figure 1.1: Distance begins with two points. On a single trait axis, the distance between individuals A and B is the absolute difference of their values.

Now take two traits, for example body length and wing span. Instead of a single line, we draw a horizontal axis for body length and a vertical axis for wing span. Every individual is now a point in the resulting plane. If individual i has body length x_i and wing span y_i , we draw it at the point (x_i, y_i) .

With three traits, we would have three axes and each individual would be a point in three-dimensional space. We cannot draw that easily on paper, but the idea is the same. In general, if we measure p traits on each individual, then each individual is a point in a p -dimensional *trait space*.

Key Idea

A multivariate phenotype is a point in trait space. The number of dimensions equals the number of traits.

This is our basic mental model: a sample is a cloud of points in trait space. Much of quanti-

tative genetics is about describing the shape of this cloud and how it moves under selection, drift, and other processes.

1.2 Differences between individuals as arrows

We often care about how two individuals differ. On a single trait, the difference between individuals i and j is just

$$x_j - x_i,$$

a signed distance along the line. When we have a reference point, every individual can be described by its displacement from that reference (Fig. 1.2).

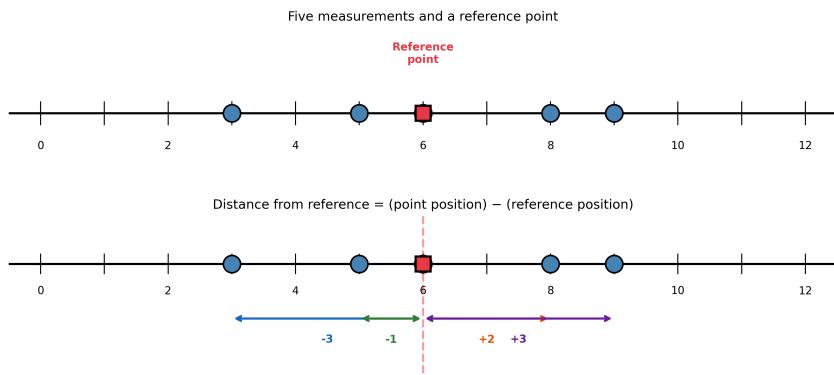


Figure 1.2: Choosing a reference point lets us describe each individual by its displacement. Here, individual A sits 2 units from the reference, B sits 7 units away. The distance between A and B is still $|7 - 2| = 5$.

In two traits, the situation is similar but now differences have two components. If individual i is at (x_i, y_i) and individual j is at (x_j, y_j) , then the difference between them can be drawn as an arrow from i to j .

To construct this arrow, we subtract coordinates:

$$\text{change in body length} = x_j - x_i,$$

$$\text{change in wing span} = y_j - y_i.$$

The arrow itself records two pieces of information:

- a *direction* in the plane (where you would walk if you were trying to go from individual i to individual j);
- a *length* (how far you would need to walk).

We can describe the same arrow in words:

“Starting at individual i , add $(x_j - x_i)$ units of body length and $(y_j - y_i)$ units of wing span.”

In coordinates, we might write this as

$$\begin{pmatrix} x_j - x_i \\ y_j - y_i \end{pmatrix}.$$

At this stage we do not need any new jargon. The important point is that these arrows behave in a regular way.

1.3 The mean as natural reference

In statistics, we typically use the sample mean as our reference point (Fig. 1.3). This choice is not arbitrary: measuring deviations from the mean is the foundation of variance.

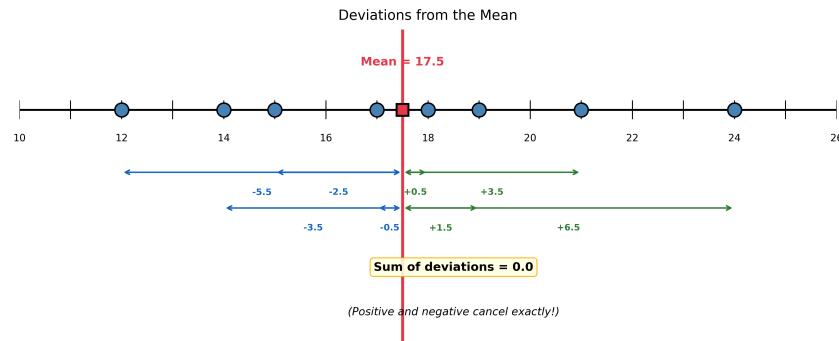


Figure 1.3: Using the mean as reference point. Each individual's position is now a deviation from the mean. The sum of all deviations (with signs) is zero—this is a defining property of the mean.

When we summarise a sample, we face a problem: how do we reduce many individual deviations to a single number that captures the “spread” of the cloud? (Fig. 1.4)

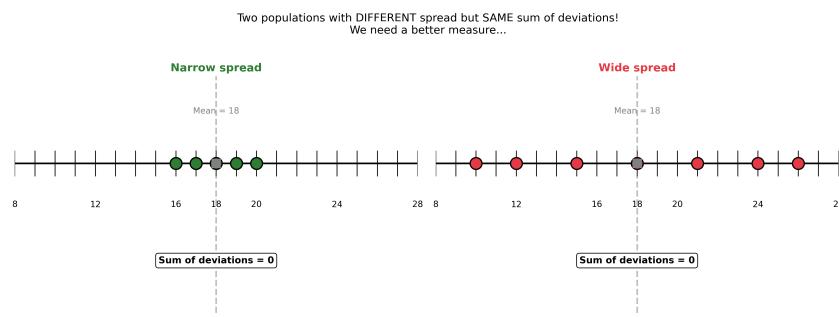


Figure 1.4: Many individuals, many deviations. We need a single number to describe how spread out the cloud is. Simply averaging the deviations will not work—positive and negative values cancel.

1.4 Adding and stretching arrows

If arrows represent changes in phenotype, then we can combine changes.

Imagine one change that goes from phenotype A to phenotype B, and another change that goes from phenotype B to phenotype C. If we draw both arrows head-to-tail, the overall change from A to C is the arrow from the start of the first to the end of the second.

Algebraically, if the first change has components $(\Delta x_1, \Delta y_1)$ and the second has components $(\Delta x_2, \Delta y_2)$, then the combined change has components

$$(\Delta x_1 + \Delta x_2, \Delta y_1 + \Delta y_2).$$

Similarly, we can stretch or shrink an arrow. If a change in phenotype is described by $(\Delta x, \Delta y)$, then half that change is $(\frac{1}{2}\Delta x, \frac{1}{2}\Delta y)$, and twice that change is $(2\Delta x, 2\Delta y)$.

Key Idea

Changes in phenotype can be added and scaled. Geometrically, this means we can join arrows head-to-tail and stretch or shrink them. Algebraically, this corresponds to adding and scaling their coordinate pairs.

This simple behaviour is what makes these objects so useful. It is the reason we will eventually give them a special name.

1.5 From arrows to vectors

We are now ready to introduce the word “vector”.

Definition 1.1. A *vector* in trait space is a quantity that has both direction and length and that can be added and scaled in the way just described.

The difference between two phenotypes is a vector. A selection gradient is also a vector: it points in the direction of steepest increase in fitness in trait space. A response to selection is a vector: it describes how the mean phenotype moves.

Each vector can be viewed in three equivalent ways:

- as an arrow in trait space;
- as a verbal instruction for how to change each trait;
- as a list of numbers, one for each trait.

For example, in two traits we might write

$$\mathbf{v} = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$$

to represent the change “add 2 units of trait 1 and subtract 1 unit of trait 2”. In three traits, a vector would have three components, and so on.

When you see a bold symbol like \mathbf{z} or β in later chapters, it will always stand for such an arrow in trait space.

Vector Addition: Head-to-Tail Composition of Phenotypic Changes
The combined change equals the sum of individual changes.

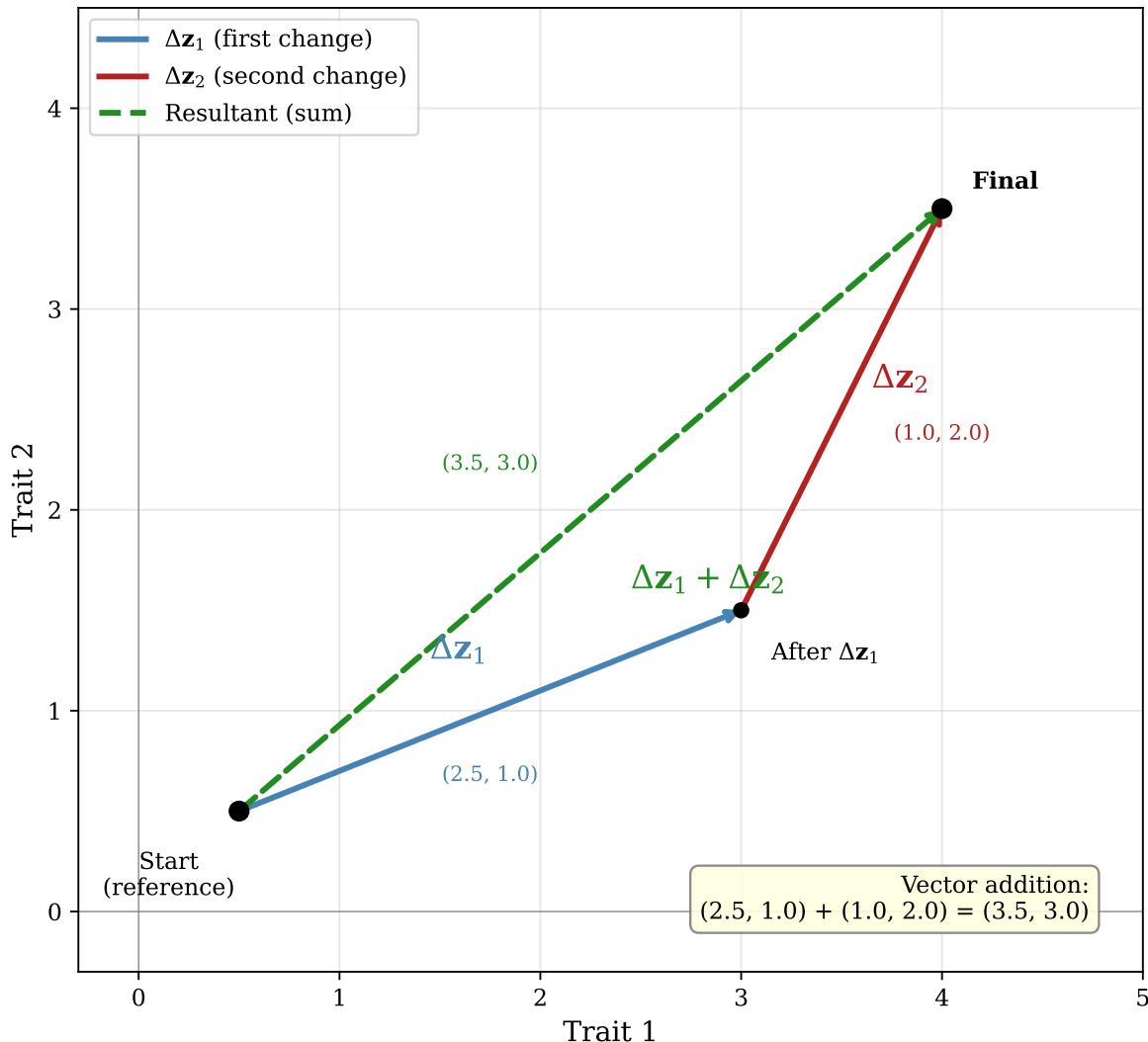


Figure 1.5: Vector addition by the head-to-tail method. (a) Two vectors \mathbf{u} and \mathbf{v} originating from the origin, representing independent changes in phenotype. (b) Head-to-tail construction: the tail of \mathbf{v} is placed at the head of \mathbf{u} . (c) The resulting sum vector $\mathbf{u} + \mathbf{v}$ connects the origin to the endpoint. This construction underlies how selection differentials accumulate across episodes of selection and how evolutionary responses combine across generations.

1.6 Distances and lengths of vectors

Once we have arrows, it is natural to ask how long they are. For a vector with components $(\Delta x, \Delta y)$, the length is given by the Pythagorean theorem:

$$\|\mathbf{v}\| = \sqrt{(\Delta x)^2 + (\Delta y)^2}.$$

We can also use this to define the distance between two phenotypes: draw the arrow from one to the other and take its length. This recovers the usual Euclidean distance distance in the plane.

In higher dimensions, the same idea applies. If a vector has components $(\Delta x_1, \Delta x_2, \dots, \Delta x_p)$, then its length is

$$\|\mathbf{v}\| = \sqrt{(\Delta x_1)^2 + (\Delta x_2)^2 + \dots + (\Delta x_p)^2}.$$

The square of this length,

$$\|\mathbf{v}\|^2 = (\Delta x_1)^2 + (\Delta x_2)^2 + \dots + (\Delta x_p)^2,$$

is a sum of squared components. This will connect directly to variance and, later, to matrix notation.

1.7 Summary

In this chapter we have:

- represented multivariate phenotypes as points in trait space;
- represented differences as arrows, and named them vectors;
- linked vector length to sums of squared components;
- shown how to generate a simple trait space plot with code.

In the next chapters we will move from these arrows to distances from a mean, and from there to variance and covariance.

Exercises

Exercise 0.1 (Plotting a phenotype cloud). Five plants are measured for leaf length (cm) and leaf width (cm):

Plant	Length	Width
A	4.2	2.1
B	5.1	2.8
C	3.8	1.9
D	4.7	2.4
E	4.2	2.3

1. Plot these five individuals as points in a two-dimensional trait space.
2. Estimate the centroid (mean phenotype) by eye from your plot.
3. Calculate the centroid exactly. How close was your estimate?
4. A sixth plant F has measurements (6.0, 3.5). Add it to your plot. How does the centroid shift?

Exercise 0.2 (Trait space dimensions). A bird ecologist measures wing length, tarsus length, bill depth, and body mass on each individual.

1. How many dimensions does this trait space have?
2. Can you visualise this space directly? If not, what strategies might help you understand the distribution of individuals?
3. If you added bill width as a fifth trait, how would the dimensionality change?

Exercise 0.3 (Phenotype as position). Consider two fish: Fish 1 has length 15 cm and mass 50 g; Fish 2 has length 20 cm and mass 80 g.

1. Represent each fish as a point in (length, mass) space.
2. Draw the arrow from Fish 1 to Fish 2. What does this arrow represent biologically?
3. If a third fish lies exactly halfway along this arrow, what are its length and mass?

Exercise 0.4 (The meaning of “distance” in trait space). Two flowers differ in petal length by 2 mm and in petal width by 3 mm.

1. What is the straight-line (Euclidean) distance between them in trait space?
2. Does this number have a direct biological interpretation?
3. What might make two flowers “far apart” in trait space but similar in fitness?

Chapter 2

Vectors, Coordinates, and Angles

In the previous chapter we met vectors as arrows in trait space: they have direction and length and can be added and scaled. In this chapter we make their algebra a bit more precise. We introduce

- coordinates for vectors;
- the dot product (inner product);
- the idea of an angle between two vectors;
- distance between phenotypes written in vector notation.

2.1 Column vectors and coordinates

Suppose we measure p traits on each individual. A phenotype is then a point in a p -dimensional trait space. When we choose axes (one per trait), we can record the position of that point as a list of p numbers.

It is convenient to write this list as a column vector:

$$\mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix},$$

where z_1 is trait 1, z_2 is trait 2, and so on.

A change in phenotype is also a vector. If an evolutionary process moves the mean phenotype by Δz_1 units in trait 1, Δz_2 in trait 2, and so on, we can record this as

$$\Delta \mathbf{z} = \begin{pmatrix} \Delta z_1 \\ \Delta z_2 \\ \vdots \\ \Delta z_p \end{pmatrix}.$$

Key Idea

Writing vectors as columns of numbers does not change what they are. It is a compact way to store “change in each trait” and to perform calculations.

2.2 Unit vectors and decomposing changes

In two traits, define

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Geometrically, \mathbf{e}_1 is a step of length 1 along trait 1, and \mathbf{e}_2 is a step of length 1 along trait 2. Any change in phenotype can be written as a combination of these unit steps. For example,

$$\begin{pmatrix} 2 \\ -1 \end{pmatrix} = 2\mathbf{e}_1 - 1\mathbf{e}_2.$$

In p traits, we have p unit vectors, and any vector can be written as

$$\mathbf{v} = v_1\mathbf{e}_1 + v_2\mathbf{e}_2 + \cdots + v_p\mathbf{e}_p.$$

This just says that the components v_1, \dots, v_p are the coordinates of the vector along the trait axes.

2.3 Lengths written in vector notation

Previously we defined the length of a vector in p traits as

$$\|\mathbf{v}\| = \sqrt{v_1^2 + v_2^2 + \cdots + v_p^2}.$$

If we form the transpose \mathbf{v}^\top ,

$$\mathbf{v}^\top = (v_1 \ v_2 \ \dots \ v_p),$$

then

$$\mathbf{v}^\top \mathbf{v} = (v_1 \ v_2 \ \dots \ v_p) \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_p \end{pmatrix} = v_1^2 + v_2^2 + \cdots + v_p^2.$$

Key Idea

The squared length of a vector can be written compactly as

$$\|\mathbf{v}\|^2 = \mathbf{v}^\top \mathbf{v}.$$

This is our first piece of matrix notation. For now, it is just a compact way to write a sum of squares.

2.4 The dot product and angles

For two vectors \mathbf{v} and \mathbf{w} , the dot product is

$$\mathbf{v}^\top \mathbf{w} = v_1 w_1 + v_2 w_2 + \cdots + v_p w_p.$$

When $\mathbf{v} = \mathbf{w}$ we recover the squared length. More generally, if θ is the angle between \mathbf{v} and \mathbf{w} , then

$$\mathbf{v}^\top \mathbf{w} = \|\mathbf{v}\| \|\mathbf{w}\| \cos \theta.$$

So:

- if \mathbf{v} and \mathbf{w} point in the same direction, $\theta \approx 0$ and the dot product is large and positive;
- if they are at right angles, $\theta = \pi/2$ and the dot product is zero;
- if they point in opposite directions, the dot product is negative.

Key Idea

The dot product measures how much two vectors point in the same direction.

This will let us talk about selection along a particular trait combination, or response along a given direction.

2.5 Projections onto a direction

Take a non-zero vector \mathbf{u} and make the unit vector in its direction:

$$\hat{\mathbf{u}} = \frac{\mathbf{u}}{\|\mathbf{u}\|}.$$

For any vector \mathbf{v} , the quantity $\hat{\mathbf{u}}^\top \mathbf{v}$ measures the component of \mathbf{v} along $\hat{\mathbf{u}}$. It is the length of the “shadow” of \mathbf{v} when projected onto that direction.

In evolutionary terms, if $\hat{\mathbf{u}}$ represents a particular trait combination (say, an eigenvector of G), then $\hat{\mathbf{u}}^\top \beta$ measures how strong selection is along that combination.

2.6 Distances between phenotypes

Let \mathbf{z}_i and \mathbf{z}_j be the phenotypes of individuals i and j . The difference between them is

$$\mathbf{z}_j - \mathbf{z}_i,$$

and the squared Euclidean distance distance is

$$d_{\text{Euc}}^2(i, j) = \|\mathbf{z}_j - \mathbf{z}_i\|^2 = (\mathbf{z}_j - \mathbf{z}_i)^\top (\mathbf{z}_j - \mathbf{z}_i).$$

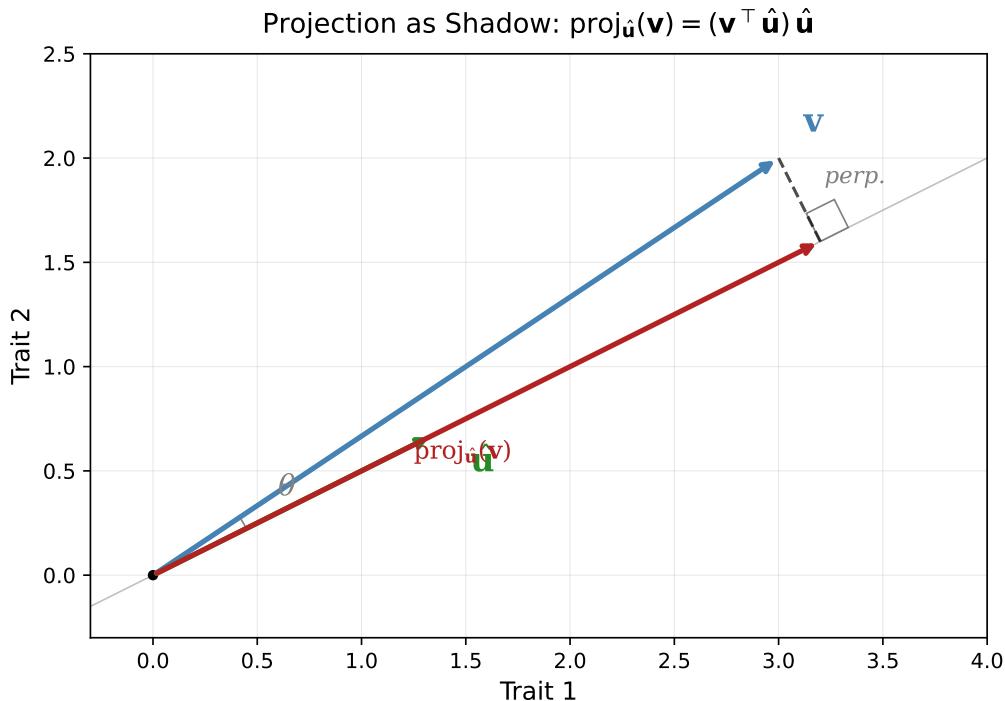


Figure 2.1: **Projection as shadow.** The projection of \mathbf{v} onto direction $\hat{\mathbf{u}}$ is the component of \mathbf{v} lying along $\hat{\mathbf{u}}$. The dashed line shows the perpendicular (residual) component. The length of the projection is $\mathbf{v}^\top \hat{\mathbf{u}} = \|\mathbf{v}\| \cos \theta$. In evolutionary terms, if $\hat{\mathbf{u}}$ is an eigenvector of \mathbf{G} , then $\beta^\top \hat{\mathbf{u}}$ measures how strongly selection aligns with that genetic axis.

Key Idea

Euclidean distance squared distance between two phenotypes can be written as

$$d_{\text{Euc}}^2(i, j) = (\mathbf{z}_j - \mathbf{z}_i)^\top (\mathbf{z}_j - \mathbf{z}_i).$$

Later, when we introduce covariance matrices, a matrix will appear between the transpose and the vector. The pattern “row \times matrix \times column” will keep returning.

2.7 Summary

In this chapter we have:

- written vectors as columns of trait values or changes;
- introduced unit vectors and decomposed vectors into components;
- defined the dot product and linked it to angles between directions;
- expressed squared lengths and distances using $\mathbf{v}^\top \mathbf{v}$ and $(\mathbf{z}_j - \mathbf{z}_i)^\top (\mathbf{z}_j - \mathbf{z}_i)$;
- introduced projections onto chosen directions.

We now have enough language to connect variance to squared distances from a mean, and to see what changes when multiple traits interact. That is the next step.

Exercises

Exercise 1.1 (Vector length). Compute the length (magnitude) of each vector:

1. $\mathbf{a} = (3, 4)$
2. $\mathbf{b} = (1, 1, 1)$
3. $\mathbf{c} = (2, -2, 1)$
4. $\mathbf{d} = (1, 0, 0, 0, 1)$

Exercise 1.2 (Normalising vectors). A unit vector has length 1. For each vector below, find the unit vector pointing in the same direction:

1. $(3, 4)$
2. $(1, 1)$
3. $(5, 0)$
4. $(1, 2, 2)$

Exercise 1.3 (Dot product and angles). The dot product of \mathbf{a} and \mathbf{b} is $\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \cos \theta$, where θ is the angle between them.

1. Compute the dot product of $(1, 0)$ and $(1, 1)$.
2. Find the angle between these vectors.
3. Compute the dot product of $(1, 2)$ and $(-2, 1)$. What does this tell you about the angle between them?
4. Two vectors are orthogonal if their dot product is zero. Find a vector orthogonal to $(3, 4)$.

Exercise 1.4 (Projection). The projection of \mathbf{a} onto \mathbf{b} is the “shadow” of \mathbf{a} in the direction of \mathbf{b} :

$$\text{proj}_{\mathbf{b}}(\mathbf{a}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\mathbf{b} \cdot \mathbf{b}} \mathbf{b}.$$

1. Project $(3, 4)$ onto $(1, 0)$. Interpret geometrically.
2. Project $(3, 4)$ onto $(1, 1)$.
3. Project $(3, 4)$ onto $(0, 1)$.
4. What is the projection of any vector onto itself?

Exercise 1.5 (Biological interpretation). In a two-trait system, the vector $(1, 1)/\sqrt{2}$ represents the direction where both traits increase equally.

1. What direction does $(1, -1)/\sqrt{2}$ represent?
2. If selection acts in the direction $(0.8, 0.6)$, is it favouring both traits equally? Which trait is favoured more?
3. A population's mean breeding value shifts by $\Delta \bar{z} = (0.5, 0.3)$. What is the magnitude of this response? What is its direction?

Chapter 3

Matrices as Machines That Move Vectors

In the previous chapters we represented phenotypes as points in trait space and changes in phenotype as vectors. We learned to measure lengths and angles using dot products. Now we need a language for transformations—rules that take one vector and return another.

This chapter introduces matrices as such rules. The key insight is geometric: a matrix does not merely store numbers in a rectangular array; it describes a transformation of space. Once you see this, covariance matrices, genetic variance matrices, and selection gradients all become visualisable.

3.1 A motivating example: scaling traits differently

Suppose we measure body size in centimetres and wing length in millimetres. A fly with body size 0.3 cm and wing length 2.5 mm sits at the point (0.3, 2.5) in our trait space.

Now imagine we want to convert both measurements to the same units—say, millimetres. Body size must be multiplied by 10; wing length stays as it is. The new coordinates are (3.0, 2.5).

We can write this conversion as a rule:

$$\begin{pmatrix} z'_1 \\ z'_2 \end{pmatrix} = \begin{pmatrix} 10 \cdot z_1 + 0 \cdot z_2 \\ 0 \cdot z_1 + 1 \cdot z_2 \end{pmatrix} = \begin{pmatrix} 10 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}.$$

The array of numbers in the middle is a *matrix*. It encodes the rule: “multiply trait 1 by 10, leave trait 2 alone.” When we apply this rule to every point in trait space, the entire cloud of phenotypes stretches horizontally by a factor of 10 while remaining unchanged vertically.

This is the central idea: a matrix is a machine that moves vectors.

3.2 Matrix–vector multiplication

Let us make the rule precise. A 2×2 matrix \mathbf{A} has four entries arranged in two rows and two columns:

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}.$$

When we multiply \mathbf{A} by a column vector \mathbf{v} , we obtain a new vector $\mathbf{w} = \mathbf{Av}$:

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} a_{11}v_1 + a_{12}v_2 \\ a_{21}v_1 + a_{22}v_2 \end{pmatrix}.$$

Each entry of the output is a dot product: row i of the matrix dotted with the input vector gives entry i of the output.

In words: the matrix takes each input component, weights it according to its entries, and combines those weighted contributions to produce each output component. This is a *linear combination*—no squares, no products of different components, just weighted sums.

Key Idea

Matrix–vector multiplication produces a new vector whose components are linear combinations of the original components. The matrix entries are the weights.

3.3 What happens to the unit vectors?

A powerful way to understand any matrix is to ask: what does it do to the standard unit vectors?

Recall from Chapter 1 that in two dimensions the unit vectors are

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Apply the matrix \mathbf{A} to \mathbf{e}_1 :

$$\mathbf{A}\mathbf{e}_1 = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} a_{11} \\ a_{21} \end{pmatrix}.$$

This is simply the first column of \mathbf{A} . Similarly,

$$\mathbf{A}\mathbf{e}_2 = \begin{pmatrix} a_{12} \\ a_{22} \end{pmatrix},$$

the second column.

Key Idea

The columns of a matrix are the images of the unit vectors. Column j tells you where \mathbf{e}_j lands after the transformation.

This observation is the key to visualising what any matrix does. If you know where the coordinate axes go, you know everything—because every other vector is a combination of those axes.

3.4 Geometric vocabulary: stretch, rotate, shear

Different matrices produce different geometric effects. Here are the main types, illustrated in two dimensions.

Scaling (stretching or compressing). A diagonal matrix stretches each axis independently:

$$\begin{pmatrix} 3 & 0 \\ 0 & 2 \end{pmatrix}$$

stretches trait 1 by a factor of 3 and trait 2 by a factor of 2. A circle of points becomes an ellipse aligned with the axes. If one diagonal entry is less than 1, that axis is compressed rather than stretched.

Rotation. The matrix

$$\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

rotates every vector by angle θ anticlockwise. A circle remains a circle; only its orientation changes. Lengths and angles between vectors are preserved.

Shear. The matrix

$$\begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

slides points horizontally in proportion to their vertical coordinate. A square becomes a parallelogram. Trait 1 gains a contribution from trait 2, but not vice versa.

Reflection. The matrix

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

flips points across the vertical axis. Reflections reverse orientation: a clockwise path around a triangle becomes anticlockwise after reflection.

Most matrices combine several of these effects. The covariance matrices we will meet shortly turn out to be pure stretches along rotated axes—no shear, no reflection. That special structure is what makes them diagonalisable.

3.5 Linearity: the defining property

Matrix transformations have a crucial property: they are *linear*. This means two things hold for any matrix \mathbf{A} and any vectors \mathbf{u}, \mathbf{v} :

1. **Additivity.** $\mathbf{A}(\mathbf{u} + \mathbf{v}) = \mathbf{Au} + \mathbf{Av}$.
2. **Scaling.** $\mathbf{A}(c\mathbf{v}) = c(\mathbf{Av})$ for any scalar c .

The Four Basic Linear Transformations

Unit square (dashed gray) → transformed shape (solid). Red/blue arrows show where e_1 and e_2 land.

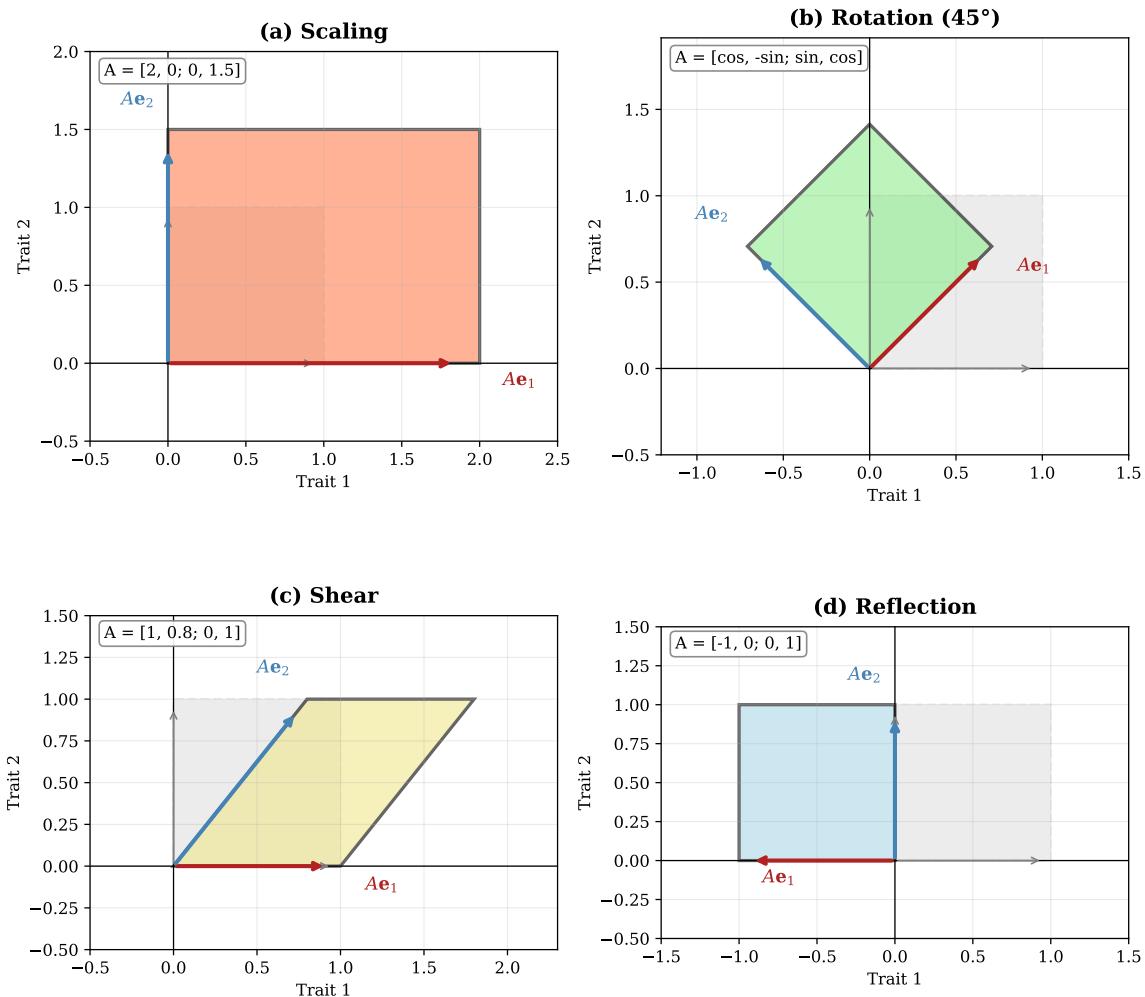


Figure 3.1: **The four basic linear transformations.** Each panel shows the unit square (dashed gray) and its image (solid colour) under a different transformation. (a) **Scaling** stretches each axis independently. (b) **Rotation** preserves lengths and angles. (c) **Shear** slides points parallel to one axis—note the parallelogram. (d) **Reflection** reverses orientation. Covariance matrices, being symmetric and positive definite, produce only scaling along rotated axes—no shear or reflection.

In plain language: if you transform two vectors and then add them, you get the same result as adding first and then transforming. And scaling a vector before or after the transformation gives the same answer.

Why does this matter biologically? Selection gradients, breeding values, and responses to selection are all defined as linear combinations of underlying quantities. The machinery of quantitative genetics is built on linearity. When nonlinear effects enter—epistasis, dominance, genotype-by-environment interaction—the linear framework becomes an approximation, and we must ask how good that approximation is. That question will occupy us in later chapters.

3.6 Composing transformations: matrix multiplication

Suppose we apply one transformation \mathbf{A} and then another \mathbf{B} . What is the combined effect?

Start with a vector \mathbf{v} . After \mathbf{A} , we have \mathbf{Av} . After \mathbf{B} , we have $\mathbf{B}(\mathbf{Av})$.

It turns out this combined transformation is itself a matrix, written \mathbf{BA} (note the order: \mathbf{A} acts first, then \mathbf{B}). The entries of the product matrix are computed by the row-times-column rule:

$$(\mathbf{BA})_{ij} = \sum_k B_{ik} A_{kj}.$$

We will not dwell on the mechanics of matrix multiplication here. The conceptual point is that composing linear transformations yields another linear transformation. This is why matrix algebra is so powerful: complex sequences of operations can be encoded as single matrices.

Key Idea

The product \mathbf{BA} represents “first \mathbf{A} , then \mathbf{B} .” Order matters: in general, $\mathbf{AB} \neq \mathbf{BA}$.

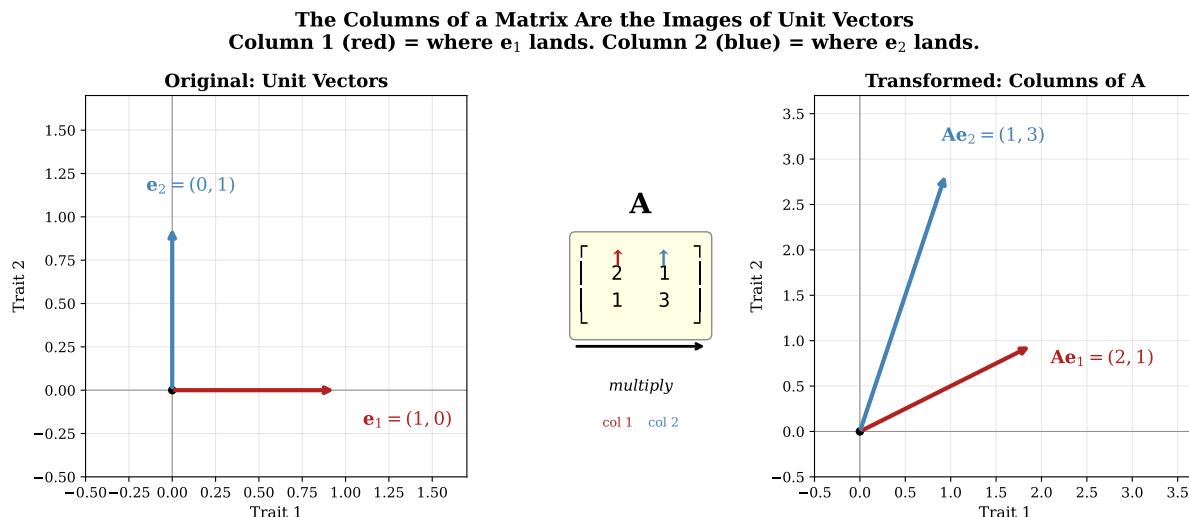


Figure 3.2: **The columns of a matrix are the images of the unit vectors.** Left: The standard unit vectors \mathbf{e}_1 and \mathbf{e}_2 . Right: Their images under the matrix \mathbf{A} . Notice that $\mathbf{Ae}_1 = (2, 1)^\top$ is exactly the first column of \mathbf{A} (red), and $\mathbf{Ae}_2 = (1, 3)^\top$ is the second column (blue). This is always true: column j tells you where \mathbf{e}_j lands.

3.7 The identity and inverse

Some matrices do nothing at all. The *identity matrix*

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

sends every vector to itself: $\mathbf{I}\mathbf{v} = \mathbf{v}$. It stretches each axis by a factor of 1—that is, it leaves everything unchanged.

If a matrix \mathbf{A} has an *inverse* \mathbf{A}^{-1} , then applying \mathbf{A} followed by \mathbf{A}^{-1} returns every vector to its starting point:

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{I}.$$

Geometrically, \mathbf{A}^{-1} undoes whatever \mathbf{A} did. If \mathbf{A} stretches trait 1 by 3, then \mathbf{A}^{-1} compresses it by 1/3. If \mathbf{A} rotates by 30, then \mathbf{A}^{-1} rotates by -30.

Not every matrix has an inverse. A matrix that collapses the plane onto a line, for instance, loses information and cannot be undone. Such matrices are called *singular*. In evolutionary applications, singular covariance matrices indicate that some trait combinations have zero variance—the population has no variation in those directions.

3.8 Symmetric matrices: a special and important case

A matrix is *symmetric* if it equals its own transpose:

$$\mathbf{A} = \mathbf{A}^\top,$$

which means $a_{ij} = a_{ji}$ for all i and j . The matrix is unchanged when you reflect it across its main diagonal.

Symmetric matrices have remarkable properties that we will exploit throughout these notes:

1. Their eigenvalues (stretching factors along special directions) are always real numbers, never complex.
2. Their eigenvectors (those special directions) are always perpendicular to one another.
3. They can always be diagonalised by a rotation—no shear is needed.

These properties mean that a symmetric matrix describes a pure stretch along a set of perpendicular axes. In two dimensions, this is an ellipse aligned with those axes. In three dimensions, an ellipsoid.

Covariance matrices are symmetric by construction: the covariance of trait 1 with trait 2 equals the covariance of trait 2 with trait 1. This is why ellipses appear everywhere in multivariate statistics and why diagonalisation is the natural tool for understanding them.

Key Idea

Symmetric matrices describe shapes—ellipses in 2D, ellipsoids in higher dimensions. The eigenvalues give the lengths of the principal axes; the eigenvectors give their directions.

We are not yet ready to define eigenvalues and eigenvectors formally. For now, hold onto the geometric picture: a symmetric matrix stretches space along perpendicular axes, and the amount of stretch along each axis is what we will call an eigenvalue.

3.9 Preview: the quadratic form

There is one more construction we need before connecting matrices to variance. Given a symmetric matrix \mathbf{A} and a vector \mathbf{v} , the quantity

$$\mathbf{v}^\top \mathbf{A} \mathbf{v}$$

is called a *quadratic form*. It takes a vector and returns a single number.

Let us unpack this in two dimensions. If

$$\mathbf{A} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix},$$

then

$$\mathbf{v}^\top \mathbf{A} \mathbf{v} = a v_1^2 + 2b v_1 v_2 + c v_2^2.$$

This is a weighted sum of squared terms and cross-products. When \mathbf{A} is a covariance matrix, this quadratic form will give us the variance of a linear combination of traits. When \mathbf{A} is a selection matrix, it will give us the curvature of the fitness surface.

The pattern “row vector \times matrix \times column vector” will appear repeatedly:

- Variance of a trait combination: $\mathbf{a}^\top \Sigma \mathbf{a}$
- Mahalanobis distance distance: $(\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu})$
- Quadratic selection: $\mathbf{z}^\top \gamma \mathbf{z}$

Understanding the quadratic form geometrically—as measuring how much a vector aligns with the axes of stretch encoded by the matrix—is the key to reading these expressions fluently.

3.10 A biological example: the \mathbf{G} matrix as a transformation

To make these ideas concrete, consider the additive genetic covariance matrix \mathbf{G} . In two traits,

$$\mathbf{G} = \begin{pmatrix} V_{A1} & \text{Cov}_A(z_1, z_2) \\ \text{Cov}_A(z_1, z_2) & V_{A2} \end{pmatrix}.$$

What does \mathbf{G} do when we treat it as a transformation?

Take the selection gradient $\boldsymbol{\beta}$, which points in the direction of steepest fitness increase. The response to selection is

$$\Delta \bar{\mathbf{z}} = \mathbf{G} \boldsymbol{\beta}.$$

The Quadratic Form $\mathbf{v}^\top \mathbf{A}\mathbf{v}$: Eigenvalues Bound the Values on the Unit Circle
 Maximum occurs along first eigenvector (red □), minimum along second (blue □).

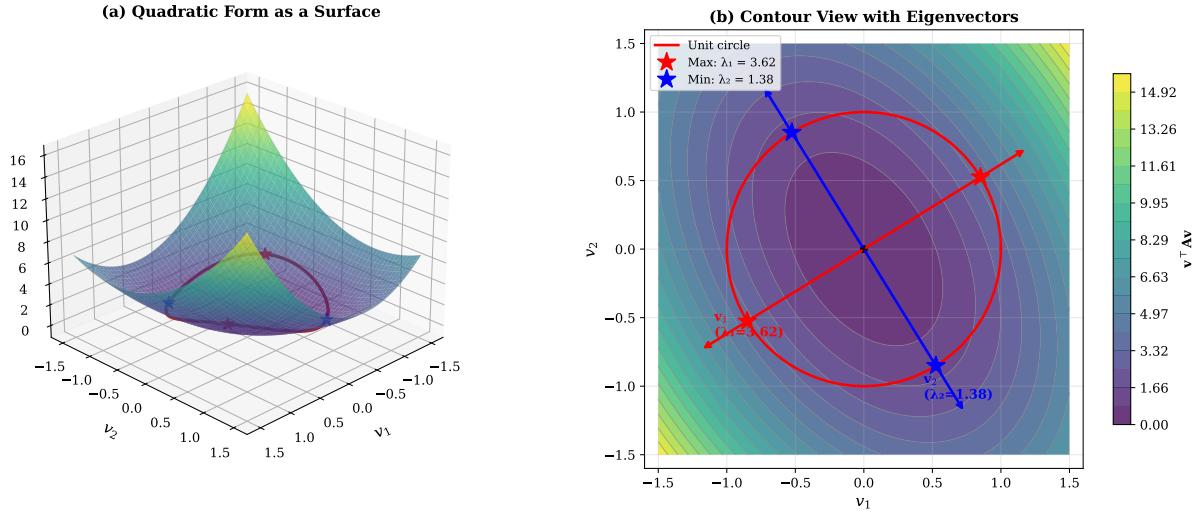


Figure 3.3: The quadratic form $\mathbf{x}^\top \mathbf{A}\mathbf{x}$ as a surface over trait space. (a) Three-dimensional view showing the paraboloid surface; the height at any point \mathbf{x} equals the quadratic form value. The red point illustrates how a specific phenotype maps to a scalar value. (b) Top-down view showing level curves, which are ellipses whose axes align with the eigenvectors of \mathbf{A} . The eigenvalues determine how steeply the surface rises along each principal direction. For covariance matrices, this surface represents variance; for selection matrices γ , it represents fitness curvature.

This is the multivariate breeder's equation equation. The matrix \mathbf{G} transforms the direction of selection into the direction of evolutionary response.

Geometrically: β tells you where selection wants to go; \mathbf{G} tells you where genetic variation allows you to go. The response $\mathbf{G}\beta$ is a compromise between these.

If \mathbf{G} is a diagonal matrix (no genetic correlations), the response is parallel to selection—you go where selection pushes. If \mathbf{G} has strong off-diagonal elements (genetic correlations), the response is deflected toward the direction of greatest genetic variance.

This deflection is not a bug; it is the central phenomenon of multivariate evolution. Understanding it requires understanding what \mathbf{G} does as a transformation, which in turn requires the diagonalisation tools we will develop in Part III.

3.11 Summary

In this chapter we have introduced matrices as rules that transform vectors. The main ideas are:

- A matrix acts on a vector to produce a new vector, with each output component being a linear combination of the input components.
- The columns of a matrix are the images of the unit vectors; they tell you where the coordinate axes go.

- Common transformations include scaling, rotation, shear, and reflection. Most matrices combine several of these.
- Symmetric matrices are special: they describe pure stretches along perpendicular axes, producing elliptical shapes.
- The quadratic form $\mathbf{v}^\top \mathbf{A} \mathbf{v}$ measures how a vector interacts with the shape encoded by a symmetric matrix.
- The \mathbf{G} matrix transforms selection gradients into evolutionary responses, making it a concrete biological example of matrix-as-machine.

We now have the vocabulary to ask precise questions about distance and shape. In Part II, we will see why the usual Euclidean distance fails when traits are correlated, and how the covariance matrix provides a remedy.

Exercises

Exercise 2.1 (Stretching). Consider the matrix

$$\mathbf{A} = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}.$$

1. Apply \mathbf{A} to the vectors $(1, 0)$, $(0, 1)$, and $(1, 1)$.
2. Describe in words what \mathbf{A} does to any vector.
3. Sketch the unit circle and its image under \mathbf{A} . What shape results?
4. Find a matrix that stretches by 3 in the x -direction and by 2 in the y -direction.

Exercise 2.2 (Rotation). The matrix

$$\mathbf{R} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

represents a rotation.

1. Apply \mathbf{R} to $(1, 0)$. Where does it go?
2. Apply \mathbf{R} to $(0, 1)$. Where does it go?
3. By what angle does \mathbf{R} rotate vectors?
4. Apply \mathbf{R} twice (compute $\mathbf{R}^2 = \mathbf{RR}$). What transformation is \mathbf{R}^2 ?

Exercise 2.3 (Shearing). Consider the shear matrix

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

1. Apply \mathbf{S} to $(1,0)$, $(0,1)$, and $(1,1)$.
2. Sketch the unit square with corners at $(0,0)$, $(1,0)$, $(0,1)$, $(1,1)$. Then sketch its image under \mathbf{S} .
3. Compute \mathbf{S}^2 . How does the shear accumulate?
4. Is \mathbf{S} symmetric? Does it change lengths?

Exercise 2.4 (Matrix multiplication). Let

$$\mathbf{A} = \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

1. Compute \mathbf{AB} and \mathbf{BA} . Are they equal?
2. \mathbf{B} swaps the two coordinates. Describe what \mathbf{AB} does (first \mathbf{B} , then \mathbf{A}).
3. Describe what \mathbf{BA} does (first \mathbf{A} , then \mathbf{B}).
4. Find the inverse of \mathbf{A} . Verify by computing \mathbf{AA}^{-1} .

Exercise 2.5 (Symmetric matrices). A symmetric matrix satisfies $\mathbf{M} = \mathbf{M}^\top$.

1. Which of the following are symmetric?

$$\begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}, \quad \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad \begin{pmatrix} 5 & 0 \\ 0 & 5 \end{pmatrix}$$

2. If \mathbf{M} is any matrix, show that $\mathbf{M}^\top \mathbf{M}$ is symmetric.
3. Covariance matrices are always symmetric. Why does this make biological sense?

Part II

Distance and Shape

Chapter 4

Distance and Why We Square It

In Part I we built a language for trait space: phenotypes are points, differences are vectors, and matrices are machines that transform vectors. Now we turn to a deceptively simple question: how do we measure how different two phenotypes are?

This chapter develops the idea of distance and explains why squaring plays such a central role in statistics. The answer is not arbitrary convention. Squaring has a geometric meaning that connects individual differences to population summaries in a way that no other operation does.

4.1 Why distance matters

Distance is fundamental to biology. When we ask whether two species have diverged, we are asking about distance in phenotype space. When we measure the strength of selection, we compare phenotypes that survive to those that do not—again, a question of distance. When we estimate heritability, we ask whether offspring are closer to their parents than to random individuals in the population.

In all these cases, we need a number that captures “how different” two phenotypes are. That number should have sensible properties:

- The distance from A to B should equal the distance from B to A.
- The distance from any phenotype to itself should be zero.
- If A, B, and C lie on a straight line with B between them, the distance from A to C should equal the sum of distances A to B and B to C.

These are the axioms of a *metric*. Different metrics give different answers to “how different,” and choosing the right metric is not a mathematical formality—it changes what we see in our data.

4.2 One trait: distance on a line

Start with a single trait. Individual i has value z_i , individual j has value z_j . The natural measure of difference is

$$z_j - z_i,$$

a signed quantity that tells us both the magnitude and direction of the difference.

But for many purposes we want an unsigned measure—just “how far apart,” regardless of which is larger. We could take the absolute value:

$$|z_j - z_i|.$$

Or we could square:

$$(z_j - z_i)^2.$$

Both give non-negative numbers that are zero only when $z_i = z_j$. Why might we prefer one over the other?

4.3 Two traits: the Pythagorean formula

With two traits, the situation becomes geometric (Fig. 4.1). Individual i sits at (x_i, y_i) , individual j at (x_j, y_j) . The difference in trait 1 is $\Delta x = x_j - x_i$; the difference in trait 2 is $\Delta y = y_j - y_i$.

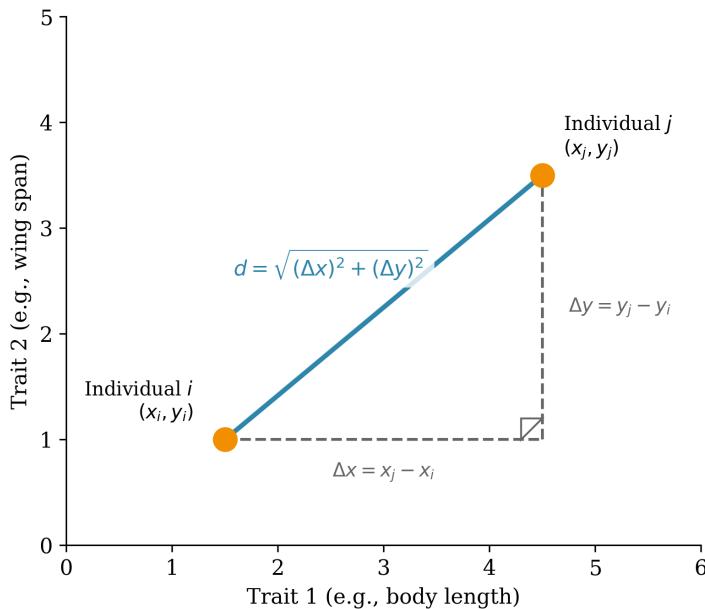


Figure 4.1: The distance between two phenotypes in trait space. The horizontal leg is Δx , the vertical leg is Δy , and the hypotenuse has length $\sqrt{(\Delta x)^2 + (\Delta y)^2}$.

The Pythagorean theorem gives the straight-line distance:

$$d = \sqrt{(\Delta x)^2 + (\Delta y)^2}.$$

This is the length of the arrow from i to j . In vector notation, if $\mathbf{v} = \mathbf{z}_j - \mathbf{z}_i$, then

$$d = \|\mathbf{v}\| = \sqrt{\mathbf{v}^\top \mathbf{v}}.$$

The squared distance is simpler:

$$d^2 = (\Delta x)^2 + (\Delta y)^2 = \mathbf{v}^\top \mathbf{v}.$$

No square root, just a sum of squared components. This pattern extends to any number of traits: if we have p traits, the squared Euclidean distance is

$$d^2 = \sum_{k=1}^p (\Delta z_k)^2.$$

4.4 Why do we square?

Here is the central question of this chapter. The Pythagorean formula involves squares. Variance involves squares. Why?

The answer has three parts, each revealing a different reason why squaring is not arbitrary.

Reason 1: Geometry demands it

The Pythagorean theorem is not a human invention; it is a property of flat space. If you want the distance along the hypotenuse of a right triangle, you must add the squares of the legs and take the square root. This is what “straight-line distance” means in Euclidean distance geometry.

Using absolute values instead— $|\Delta x| + |\Delta y|$ —gives a different metric, sometimes called “Manhattan distance” or “taxicab distance” because it measures how far you would walk along a grid of streets. This is a perfectly valid metric, but it does not measure straight-line distance. In trait space, we usually want the length of the arrow, not the length of a path that only moves parallel to the axes.

Reason 2: Algebra rewards it

Squared quantities have a remarkable property: they decompose additively when the underlying components are independent.

Consider two independent random variables X and Y . The variance of their sum satisfies

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y).$$

This additivity is the foundation of ANOVA, of partitioning variance into genetic and environmental components, of combining information across independent sources. It works because variance is defined using squares.

If we used absolute deviations instead, we would not get this clean decomposition. The mean absolute deviation of a sum is not, in general, the sum of the mean absolute deviations. The algebra becomes intractable.

Key Idea

Squaring is the price we pay for additivity. Variance decomposes into components precisely because it is built from squared deviations.

Reason 3: Calculus prefers it

Squared functions are smooth. The function $f(x) = x^2$ has a derivative everywhere; the function $f(x) = |x|$ has a kink at zero where the derivative is undefined.

This matters when we optimise. Least-squares fitting works because the sum of squared residuals is a smooth, bowl-shaped function with a unique minimum that calculus can find. Least-absolute-deviation fitting is harder: the objective function has corners, and the minimum is not always unique.

In evolutionary biology, we often model fitness surfaces, selection gradients, and breeding values using derivatives. Squared quantities fit naturally into this calculus-based framework.

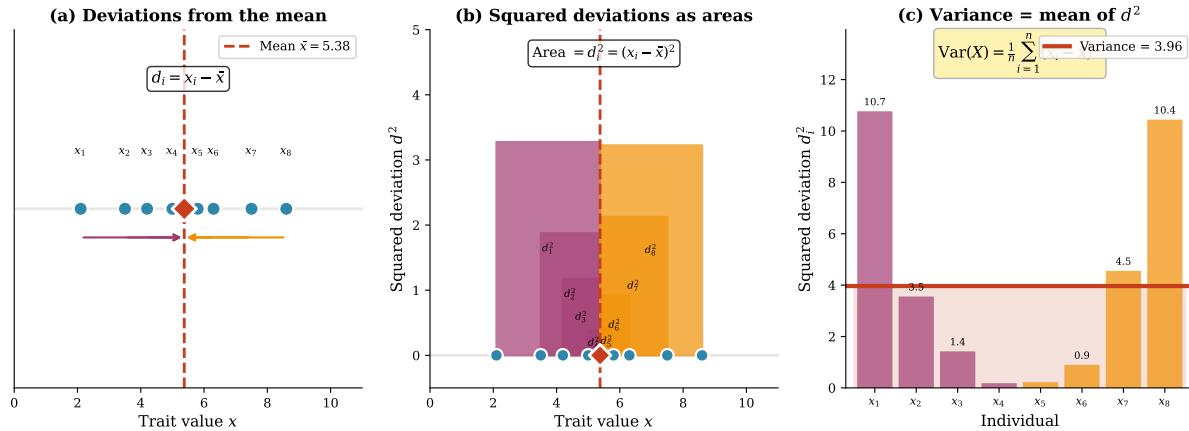


Figure 4.2: **Variance as the mean of squared distances from the mean.** (a) A sample of eight individuals (blue points) on a trait axis, with deviation arrows showing each point's displacement from the mean $\bar{x} = 5.38$ (red diamond). Orange arrows indicate positive deviations; magenta arrows indicate negative deviations. (b) The same deviations represented as squares, where the area of each square equals $d_i^2 = (x_i - \bar{x})^2$. Larger deviations contribute disproportionately more to the variance because of the squaring operation. (c) Bar chart of individual squared deviations. The horizontal red line marks the variance—the mean of these squared deviations. The shaded region emphasizes that variance is this average, not the sum. This figure illustrates why squaring is essential: it makes all contributions positive while maintaining the geometric connection to distance via Pythagoras.

4.5 From individual distances to population spread

Now we make the connection to variance.

Take a sample of n individuals with values z_1, z_2, \dots, z_n on a single trait. The sample mean is

$$\bar{z} = \frac{1}{n} \sum_{i=1}^n z_i.$$

Each individual's deviation from the mean is $z_i - \bar{z}$. The sample variance is the average of the squared deviations:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (z_i - \bar{z})^2.$$

Key Idea

Variance is the mean squared distance from the mean. It measures how spread out the sample is by averaging how far each individual sits from the centre.

The geometric picture is clear (Fig. 4.3): each individual has an arrow pointing from the mean to its location. Variance averages the squared lengths of these arrows.

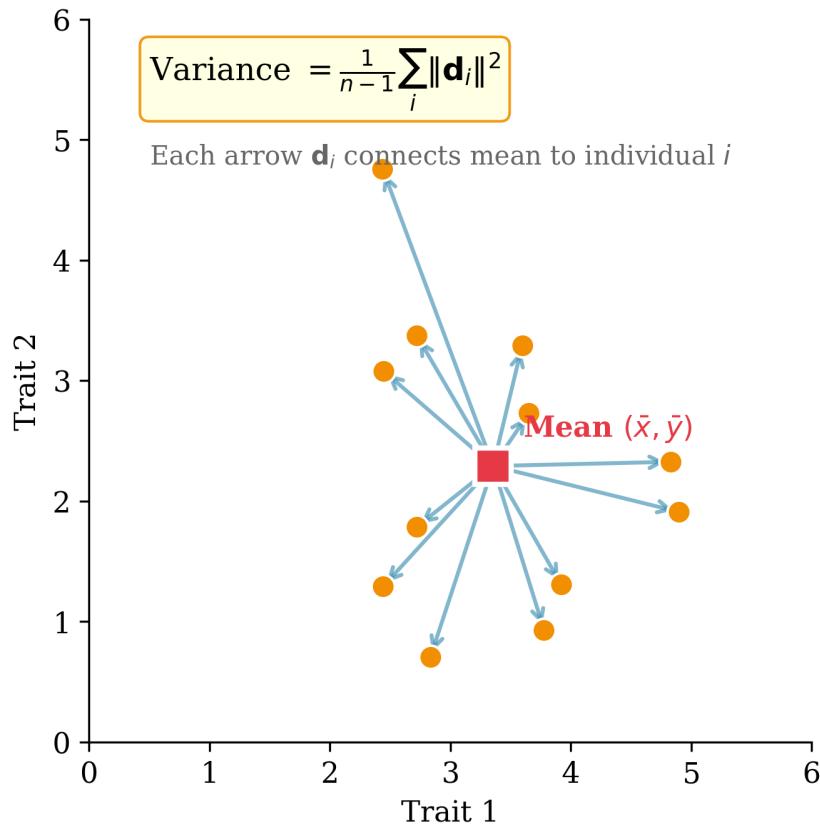


Figure 4.3: Variance as the mean squared distance from the mean. Each arrow connects an individual to the sample mean. Squaring and averaging these lengths gives the variance.

Why $n - 1$ in the denominator rather than n ? This is Bessel's correction, which makes the sample variance an unbiased estimator of the population variance. The geometric reason: the deviations $z_i - \bar{z}$ are constrained to sum to zero (by definition of the mean), so only $n - 1$ of them are free to vary. We are averaging over $n - 1$ independent pieces of information, not n .

4.6 Variance in two traits: the covariance matrix appears

With two traits, each individual is a point in the plane. The mean is now a point (\bar{x}, \bar{y}) , and each individual's deviation is a vector:

$$\mathbf{d}_i = \begin{pmatrix} x_i - \bar{x} \\ y_i - \bar{y} \end{pmatrix}.$$

The squared distance from individual i to the mean is

$$\|\mathbf{d}_i\|^2 = (x_i - \bar{x})^2 + (y_i - \bar{y})^2.$$

If we average these squared distances, we get a single number—the total variance, summed across traits. But this discards information. It does not tell us whether the cloud is elongated, or in which direction.

To capture the shape of the cloud, we need to keep track of more than just the sum of squares. We need the individual squared deviations in each trait *and* the cross-products between traits. This leads to the covariance matrix:

$$\mathbf{S} = \frac{1}{n-1} \sum_{i=1}^n \mathbf{d}_i \mathbf{d}_i^\top = \frac{1}{n-1} \sum_{i=1}^n \begin{pmatrix} (x_i - \bar{x})^2 & (x_i - \bar{x})(y_i - \bar{y}) \\ (x_i - \bar{x})(y_i - \bar{y}) & (y_i - \bar{y})^2 \end{pmatrix}.$$

The diagonal entries are the variances of each trait. The off-diagonal entries are the covariances—they measure how the two traits vary together.

Key Idea

The covariance matrix packages the variances and covariances into a single object. It captures not just how spread out the data are, but the shape and orientation of the cloud.

Notice the construction: each deviation vector \mathbf{d}_i is multiplied by its own transpose to form a 2×2 matrix, and these matrices are averaged. This is the multivariate generalisation of “average of squared deviations.”

4.7 A worked example

Consider five individuals measured on two traits:

Individual	Trait 1 (x)	Trait 2 (y)
1	2	3
2	4	5
3	3	4
4	5	7
5	6	6

The means are $\bar{x} = 4$ and $\bar{y} = 5$. The deviations are:

Individual	$x_i - \bar{x}$	$y_i - \bar{y}$
1	-2	-2
2	0	0
3	-1	-1
4	1	2
5	2	1

The sums of squares and cross-products are:

$$\begin{aligned}\sum(x_i - \bar{x})^2 &= 4 + 0 + 1 + 1 + 4 = 10, \\ \sum(y_i - \bar{y})^2 &= 4 + 0 + 1 + 4 + 1 = 10, \\ \sum(x_i - \bar{x})(y_i - \bar{y}) &= 4 + 0 + 1 + 2 + 2 = 9.\end{aligned}$$

Dividing by $n - 1 = 4$:

$$\mathbf{S} = \begin{pmatrix} 2.5 & 2.25 \\ 2.25 & 2.5 \end{pmatrix}.$$

The variances are equal (2.5 each), and the covariance is positive (2.25), indicating that the traits tend to increase together. The cloud is elongated along a diagonal.

4.8 The covariance matrix as a shape

The covariance matrix \mathbf{S} is symmetric: the (1,2) entry equals the (2,1) entry. From Chapter 2, we know that symmetric matrices describe shapes—ellipses in two dimensions, ellipsoids in higher dimensions.

What shape does \mathbf{S} describe? Consider the set of all points \mathbf{z} satisfying

$$(\mathbf{z} - \bar{\mathbf{z}})^\top \mathbf{S}^{-1} (\mathbf{z} - \bar{\mathbf{z}}) = c$$

for some constant c . This is an ellipse centred at the mean (Fig. 4.4).

The eigenvalues of \mathbf{S} determine the lengths of the ellipse's axes. The eigenvectors determine the directions of those axes. If the eigenvalues are equal, the ellipse is a circle. If they are very different, the ellipse is elongated.

We will develop this connection fully in Chapter 12. For now, the key point is that squaring gives us access to this geometric structure. The covariance matrix is not just a table of numbers; it is a description of shape.

4.9 What squared distance assumes

Euclidean distance treats all traits equally. One unit of body length counts the same as one unit of wing span, regardless of how variable each trait is or how they correlate.

This assumption is often unreasonable:

- If body length varies over a range of 10 units and wing span varies over 0.1 units, a difference of 1 unit in body length is tiny (within normal variation), while a difference of 1 unit in wing span is enormous (ten times the typical range).

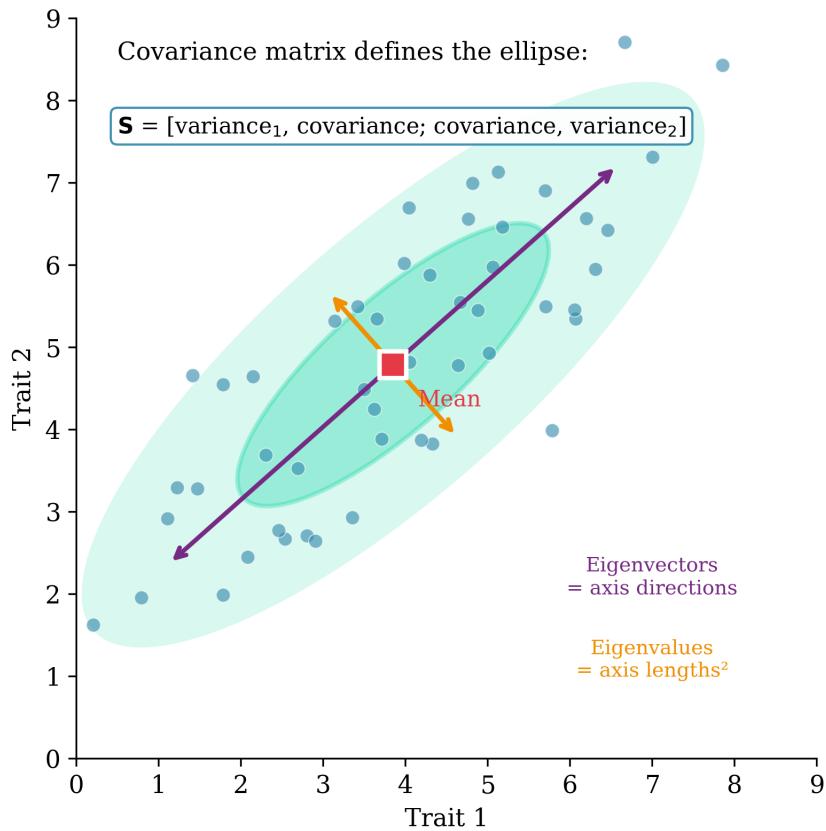


Figure 4.4: The covariance matrix defines an ellipse. Points on the ellipse are equidistant from the mean in a sense we will make precise. The orientation and elongation of the ellipse encode the correlations between traits.

- If body length and wing span are positively correlated, an individual with large body and small wings is unusual not because either trait is extreme, but because the *combination* is rare.

Euclidean distance ignores both issues. It can badly misrepresent how “different” two phenotypes really are.

Key Idea

Euclidean distance assumes all traits are measured on comparable scales and are uncorrelated. When these assumptions fail, we need a better metric.

This sets up the problem that the next chapter will solve.

4.10 Summary

In this chapter we have:

- Motivated distance as a fundamental biological quantity—the answer to “how different are these phenotypes?”

- Derived the Pythagorean formula and connected it to $\mathbf{v}^\top \mathbf{v}$.
- Explained why we square: geometry demands it for straight-line distance; algebra rewards it with additive decomposition; calculus prefers the smoothness.
- Reframed variance as mean squared distance from the mean.
- Introduced the covariance matrix as a summary of spread that captures shape, not just magnitude.
- Noted that Euclidean distance assumes comparable scales and no correlation—assumptions that often fail.

The stage is set. In the next chapter, we will see concrete examples where Euclidean distance gives misleading answers, and we will introduce the Mahalanobis distance distance as the remedy. The covariance matrix will move from being a summary statistic to being a tool for measuring distance itself.

Exercises

Exercise 10.1 (Euclidean distance). Compute the Euclidean distance between:

1. (0, 0) and (3, 4)
2. (1, 2) and (4, 6)
3. (0, 0, 0) and (1, 2, 2)
4. (1, 1, 1, 1) and (2, 2, 2, 2)

Exercise 10.2 (Variance as mean squared distance). Consider the data set $\{2, 4, 6, 8, 10\}$.

1. Compute the mean.
2. Compute each observation's deviation from the mean.
3. Square these deviations and compute their average. This is the variance.
4. Verify using the formula $\text{Var}(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$.

Exercise 10.3 (Why squaring?). Suppose we defined “variance” using absolute deviations instead of squared deviations: $\text{MAD} = \frac{1}{n} \sum_i |x_i - \bar{x}|$.

1. Compute MAD for the data $\{2, 4, 6, 8, 10\}$.
2. The function $f(x) = |x|$ has a corner at $x = 0$. Why does this make calculus difficult?
3. The function $g(x) = x^2$ is smooth everywhere. Why does this matter for optimisation?
4. Give one advantage and one disadvantage of using squared deviations.

Exercise 10.4 (The mean minimises squared distance). Consider three points on a line: $x_1 = 1, x_2 = 3, x_3 = 5$.

1. Compute the sum of squared distances from each point to $c = 2$: $\sum_i (x_i - 2)^2$.
2. Compute the sum of squared distances from each point to $c = 3$ (the mean).
3. Compute the sum of squared distances from each point to $c = 4$.
4. Which value of c minimises the sum of squared distances?
5. (Challenge) For general data x_1, \dots, x_n , use calculus to show that the sum of squared distances is minimised when $c = \bar{x}$.

Exercise 10.5 (Distance in trait space). Two birds are measured:

- Bird A: wing = 10 cm, tarsus = 2 cm
 - Bird B: wing = 12 cm, tarsus = 2.5 cm
1. Compute the Euclidean distance between them.
 2. Now express wing in mm instead of cm. Recompute the distance.
 3. Why did the distance change? What does this tell us about using raw Euclidean distance for traits measured on different scales?

Chapter 5

When Euclidean distance Distance Fails

In the previous chapter we derived the Euclidean distance distance formula and saw why squaring plays a central role in statistics. But we ended with a warning: Euclidean distance distance assumes that all traits are measured on comparable scales and that traits are uncorrelated. When these assumptions fail, Euclidean distance distance can give badly misleading answers.

This chapter presents three examples where Euclidean distance distance leads us astray. Each example reveals a different way the assumption can break. By the end, you will be convinced that we need a better metric—one that accounts for the structure of variation in the data.

5.1 Example 1: The problem of scale

Consider two morphological traits measured on a sample of beetles:

- Elytra length, measured in millimetres, with a sample mean of 12 mm and a standard deviation of 2 mm.
- Body mass, measured in milligrams, with a sample mean of 450 mg and a standard deviation of 80 mg.

Now compare two beetles to a reference individual at the population mean (12 mm, 450 mg):

Beetle	Elytra (mm)	Mass (mg)	Euclidean distance distance
Reference	12	450	0
A	14	450	2
B	12	530	80

Beetle A differs from the reference by 2 mm in elytra length. Beetle B differs by 80 mg in mass. The Euclidean distance distances are 2 and 80, respectively. By this measure, beetle B is 40 times more different from the reference than beetle A.

But wait. Beetle A is one standard deviation above the mean in elytra length ($2 \text{ mm}/2 \text{ mm} = 1$). Beetle B is also one standard deviation above the mean in mass ($80 \text{ mg}/80 \text{ mg} = 1$). In terms of how unusual each beetle is within the population, they are *equally extreme*.

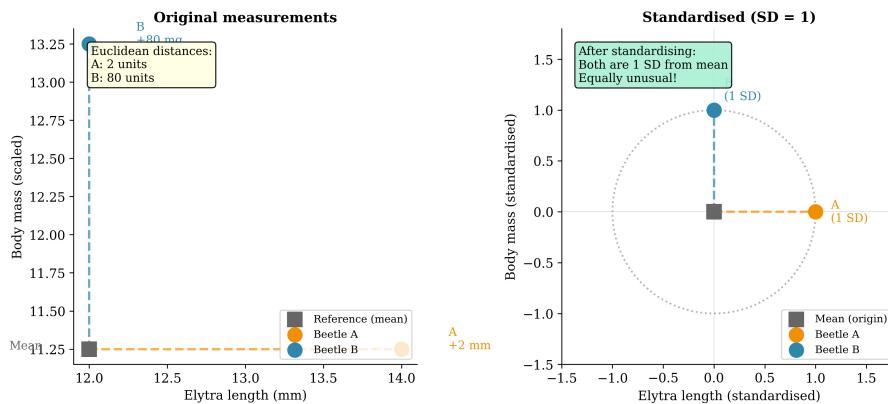


Figure 5.1: Euclidean distance distance depends on measurement units. Beetles A and B are each one standard deviation from the mean, but Euclidean distance distance makes B appear 40 times more extreme because mass is measured in larger numbers.

The problem is clear: Euclidean distance distance treats a difference of 1 mm the same as a difference of 1 mg, even though 1 mm is a large deviation for elytra length while 1 mg is tiny for body mass. The units of measurement contaminate our notion of “how different.”

Key Idea

Euclidean distance distance is not unit-free. If you change from millimetres to metres, or from milligrams to grams, the distances change. This is a problem because biological questions should not depend on arbitrary choices of measurement scale.

A partial fix: standardisation

One common remedy is to standardise each trait by its standard deviation before computing distances. Define

$$z_{\text{std}} = \frac{z - \bar{z}}{s},$$

where s is the sample standard deviation. After standardisation, each trait has mean zero and standard deviation one.

For our beetles:

$$\begin{aligned} \text{Beetle A (standardised): } & (1, 0) \\ \text{Beetle B (standardised): } & (0, 1) \end{aligned}$$

The Euclidean distance distances from the origin (the standardised mean) are now both equal to 1. Problem solved?

Not quite. Standardisation fixes the scale problem, but it ignores something else: the correlation between traits. The next example shows why this matters.

5.2 Example 2: The problem of correlation

Suppose elytra length and body mass are positively correlated—larger beetles tend to be both longer and heavier. This is biologically reasonable; body parts scale together.

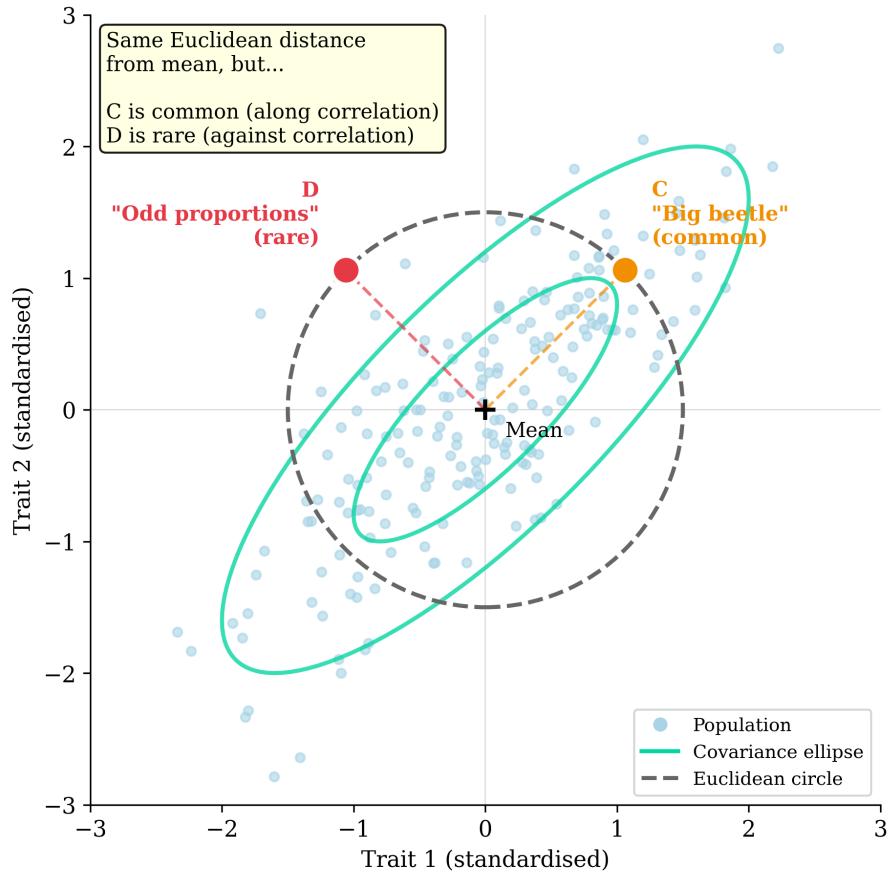


Figure 5.2: correlation

problem] When traits are correlated, the cloud of individuals is elongated. Points C and D are equally far from the mean in Euclidean distance terms, but C lies along the main axis of variation (common) while D lies perpendicular to it (rare).

Consider two more beetles, both one unit of Euclidean distance distance from the mean after standardisation (Fig. 5.2):

- Beetle C: large elytra and heavy body—both traits elevated together, in the direction of the correlation.
- Beetle D: large elytra but light body—one trait elevated, the other depressed, against the correlation.

Euclidean distance distance says C and D are equally different from the mean. But look at the data cloud. Beetles like C are common; they lie along the elongated axis of the ellipse. Beetles like D are rare; they lie in a direction where the population shows little variation.

In biological terms, a beetle with large elytra and heavy body is just a “big beetle”—unusual in size but not in proportion. A beetle with large elytra but light body has an unusual *combination* of traits. It is a genuine outlier, not just an extreme of normal variation.

Key Idea

Euclidean distance ignores correlation structure. It treats directions of high variation the same as directions of low variation. Two points can have the same Euclidean distance from the mean but very different probabilities under the population distribution.

Why standardisation does not help

Standardising each trait individually does not fix this problem. Standardisation rescales the axes so that each trait has variance 1, but it does not rotate the axes to align with the correlation structure. The ellipse becomes a different ellipse (closer to circular along the original axes), but it is still tilted.

What we need is a distance measure that accounts for the full covariance structure—both the variances and the correlations. That measure is the Mahalanobis distance distance, which we will develop in the next chapter.

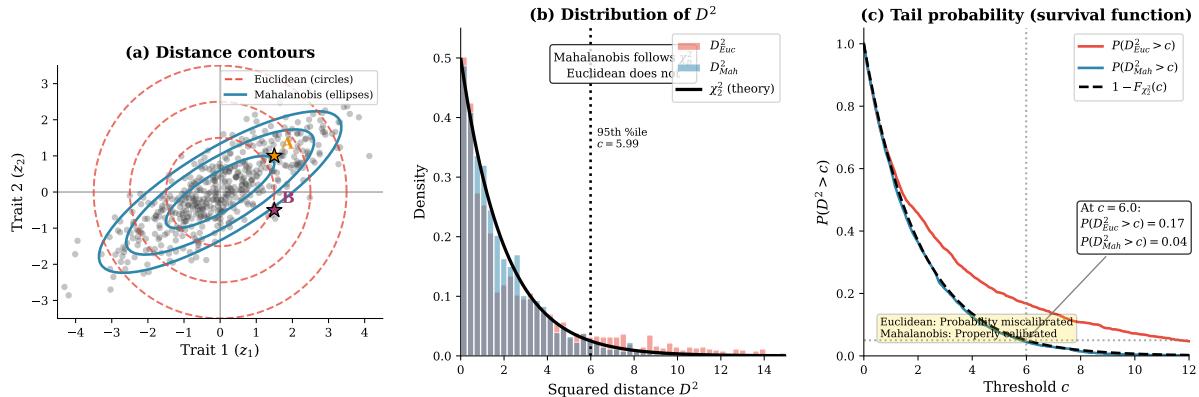


Figure 5.3: Probability calibration: Euclidean versus Mahalanobis distance. Data drawn from a bivariate normal with correlation $\rho = 0.8$. (a) Scatter plot with contours of constant distance. Red dashed circles show Euclidean distance contours; blue ellipses show Mahalanobis distance contours aligned with the covariance structure. Point A lies along the correlation direction (common phenotype); point B lies against it (rare phenotype). Despite similar Euclidean distances, their Mahalanobis distances differ dramatically: $D_{Mah}(A) = 1.05$ versus $D_{Mah}(B) = 2.39$. (b) Distribution of squared distances. Mahalanobis D^2 follows the theoretical χ^2_2 distribution (black curve); Euclidean D^2 does not. (c) Tail probabilities $P(D^2 > c)$. At the 95th percentile of χ^2_2 ($c = 5.99$), Mahalanobis correctly identifies 5% of points as extreme, while Euclidean identifies 17%—a threefold miscalibration. This is why Mahalanobis distance is essential for probability-based inference: it is the only metric properly calibrated to the data’s covariance structure.

5.3 Example 3: The probability perspective

Here is another way to see the problem. Suppose our two traits follow a bivariate normal distribution with means $\mu_1 = 0$, $\mu_2 = 0$, standard deviations $\sigma_1 = 1$, $\sigma_2 = 1$, and correlation $\rho = 0.8$.

The probability density at a point (z_1, z_2) depends on how “central” that point is. Points near the mean have high density; points far from the mean have low density. But “far from the mean” must be measured in a way that respects the correlation.

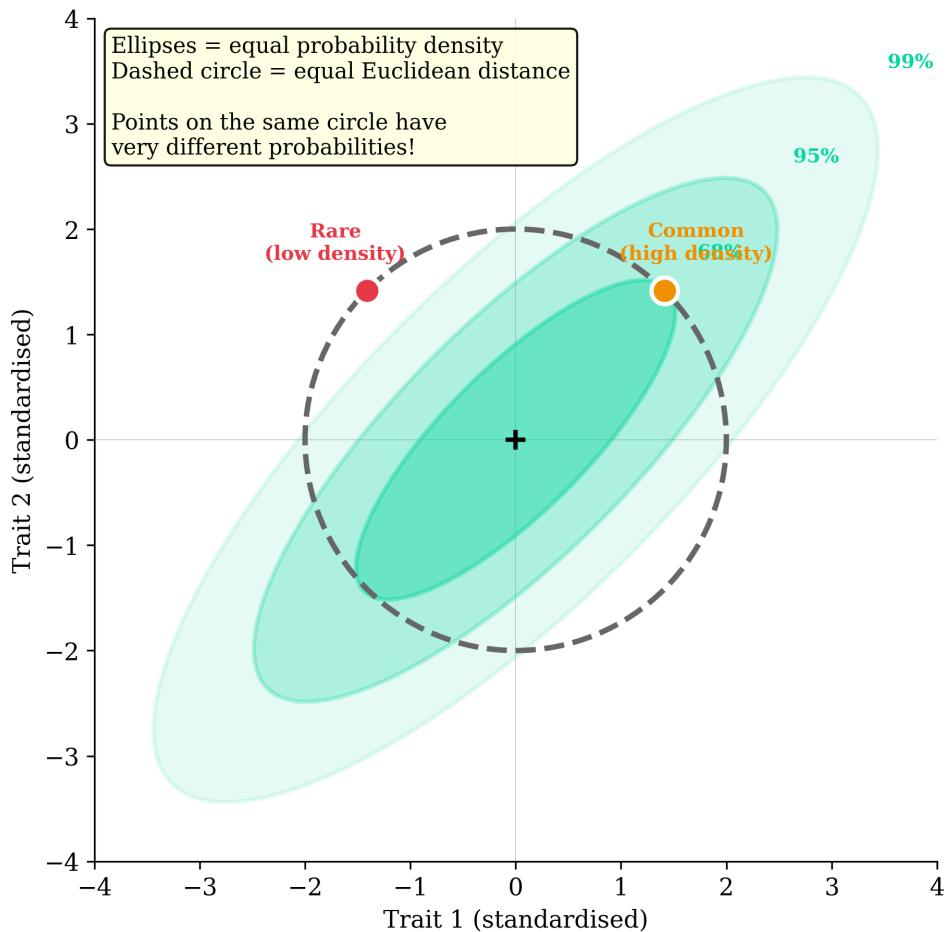


Figure 5.4: Contours of equal probability density for a correlated bivariate normal. The contours are ellipses, not circles. Points on the same Euclidean distance circle (dashed) can have very different probability densities.

Figure 5.4 shows contours of equal probability density. These contours are ellipses aligned with the correlation structure. The dashed circle shows points at constant Euclidean distance from the mean. Notice that:

- Some points on the circle lie well inside the 95% probability contour—they are not unusual at all.
- Other points on the same circle lie far outside—they are extremely rare.

Euclidean distance cannot distinguish these cases. It sees only the radius of the circle, not where on the circle the point lies.

Key Idea

Euclidean distance distance is blind to the shape of the distribution. Points at the same Euclidean distance distance can have wildly different probabilities. A proper distance metric should make “equally distant” mean “equally probable.”

5.4 A biological interlude: why this matters for selection

These are not just statistical curiosities. The failure of Euclidean distance distance has direct consequences for how we think about natural selection.

Consider stabilising selection on multiple traits. Fitness decreases as phenotypes deviate from an optimum. If we model this using Euclidean distance distance from the optimum,

$$w(\mathbf{z}) = \exp\left(-\frac{1}{2}\|\mathbf{z} - \boldsymbol{\theta}\|^2\right),$$

we are implicitly assuming that all directions of deviation are equally costly. A beetle that is too long receives the same fitness penalty as a beetle that is too heavy, unit for unit.

But if the population has abundant genetic variation in the “too long and too heavy” direction and little variation in the “too long but too light” direction, these deviations are not biologically equivalent. The first represents a common, easily produced phenotype; the second represents a rare, difficult-to-produce phenotype.

A more realistic fitness function would penalise deviations according to how unusual they are in the population—that is, according to Mahalanobis distance distance, not Euclidean distance distance. We will return to this point when we discuss selection surfaces in Part IV.

5.5 What we need from a better metric

Let us summarise what Euclidean distance distance gets wrong and what a better metric should provide.

Problem 1: Scale dependence. Euclidean distance distance changes when we change units. A better metric should be *scale-invariant*—the answer should not depend on whether we measure length in millimetres or metres.

Problem 2: Ignoring correlation. Euclidean distance distance treats all directions equally. A better metric should weight directions according to how variable the population is in those directions. Deviations in low-variance directions should count more than deviations in high-variance directions.

Problem 3: Disconnection from probability. Euclidean distance distance does not correspond to probability. A better metric should have the property that points at the same distance from the mean are equally probable under the population distribution.

All three problems have a common solution: use the covariance matrix to define distance. The covariance matrix knows about both variances (which address the scale problem) and correlations (which address the shape problem). Using it correctly will give us a distance that corresponds to probability.

5.6 A geometric preview

Before we derive the Mahalanobis distance distance formally, let us see geometrically what we are aiming for.

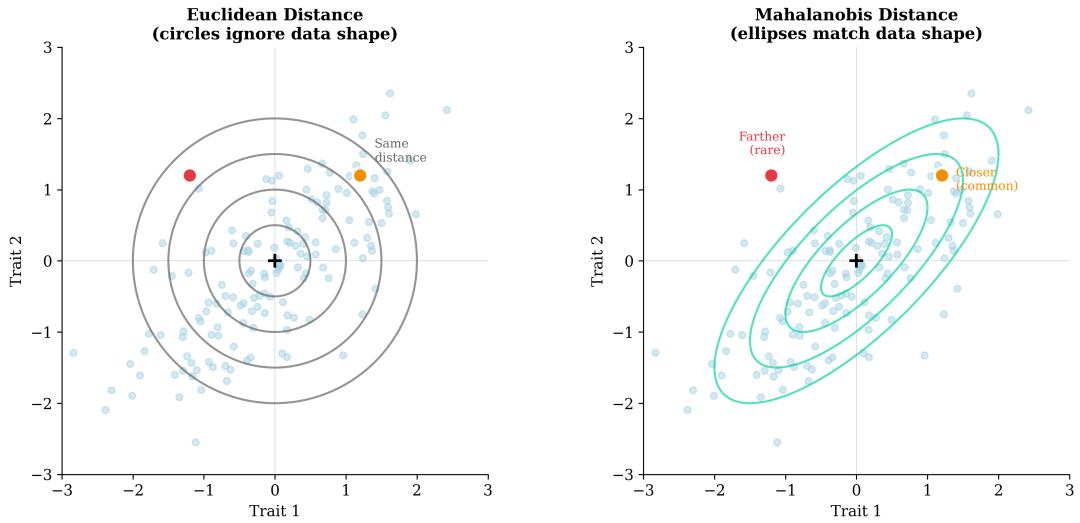


Figure 5.5: Left: Euclidean distance distance uses circles centred on the mean. Points on the same circle are “equally far” even if some are common and others rare. Right: Mahalanobis distance distance uses ellipses that match the shape of the data. Points on the same ellipse are equally probable—this is the right notion of “equally far” for a population.

The left panel of Fig. 5.5 shows Euclidean distance distance: circles centred on the mean. The right panel shows what we want: ellipses that match the shape of the covariance structure. Points on the same ellipse are equally probable; they represent the same degree of “unusualness.”

The transformation from circles to ellipses is exactly what the covariance matrix encodes. In the next chapter, we will see how to use the inverse of the covariance matrix to define a new distance—the Mahalanobis distance distance—that has all the properties we want.

5.7 A worked comparison

To make the contrast concrete, consider the following covariance matrix:

$$\mathbf{S} = \begin{pmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{pmatrix}.$$

Both traits have variance 1, and the correlation is 0.8. Consider three points, all at Euclidean distance distance 1 from the origin:

Point	z_1	z_2	Euclidean distance dist	Direction
E	$1/\sqrt{2}$	$1/\sqrt{2}$	1.0	Along correlation
F	$1/\sqrt{2}$	$-1/\sqrt{2}$	1.0	Against correlation
G	1	0	1.0	Along trait 1 only

Point E lies along the major axis of the ellipse (both traits elevated together). Point F lies along the minor axis (traits in opposition). Point G lies along the first trait axis.

All three have Euclidean distance distance 1. But their Mahalanobis distance distances (which we will compute properly in the next chapter) are:

Point	Euclidean distance dist	Mahalanobis distance dist
E	1.0	0.53
F	1.0	2.36
G	1.0	1.00

Point E, which lies in the direction of maximum variation, has a small Mahalanobis distance distance—it is not unusual. Point F, which lies in the direction of minimum variation, has a large Mahalanobis distance distance—it is very unusual. Point G is intermediate.

Key Idea

Mahalanobis distance distance rescales directions by how variable the population is in those directions. Large deviations in high-variance directions are less surprising than small deviations in low-variance directions.

5.8 The connection to standardisation

You might wonder: if we standardise each trait to have variance 1, and then rotate to align with the principal axes of the correlationmatrix, would that fix the problem?

Yes—and that is exactly what Mahalanobis distance distance does, algebraically. It can be understood as:

1. Standardise each trait by its standard deviation (fixing the scale problem).
2. Rotate to the principal axes of the covariance matrix (fixing the correlationproblem).
3. Measure ordinary Euclidean distance distance in this transformed space.

The matrix that performs this combined standardisation-and-rotation is related to the inverse (or inverse square root) of the covariance matrix. We will derive this in the next chapter.

This preview should help you see that Mahalanobis distance distance is not some arbitrary alternative to Euclidean distance distance. It is the natural distance in a space that has been “whitened”—transformed so that the covariance matrix becomes the identity. In whitened space, Euclidean distance distance works correctly because the assumptions behind it are satisfied.

5.9 Summary

In this chapter we have seen three ways that Euclidean distance distance fails:

- **Scale dependence:** Distances change with measurement units. Two individuals equally extreme in their traits can have very different Euclidean distance distances.
- **Ignoring correlation:** Directions of high and low variation are treated equally. Points along the major axis of variation appear just as extreme as points along the minor axis.
- **Disconnection from probability:** Points at the same Euclidean distance distance can have very different probabilities under the population distribution.

We have also previewed the solution:

- The covariance matrix encodes both scale (variances) and shape (correlations).
- Using the covariance matrix to define distance gives us contours that are ellipses matching the data, not circles ignoring it.
- This distance—the Mahalanobis distance distance—makes “equally far” mean “equally probable.”

In the next chapter, we derive the Mahalanobis distance distance formally and show how the inverse covariance matrix enters the formula. The key insight will be that inserting a matrix between the vectors in our distance formula changes the shape of the “unit ball” from a circle to an ellipse.

Copy

Exercises

Exercise 0.1 (Plotting a phenotype cloud). Five plants are measured for leaf length (cm) and leaf width (cm):

Plant	Length	Width
A	4.2	2.1
B	5.1	2.8
C	3.8	1.9
D	4.7	2.4
E	4.2	2.3

1. Plot these five individuals as points in a two-dimensional trait space.
2. Estimate the centroid (mean phenotype) by eye from your plot.
3. Calculate the centroid exactly. How close was your estimate?
4. A sixth plant F has measurements (6.0, 3.5). Add it to your plot. How does the centroid shift?

Exercise 0.2 (Trait space dimensions). A bird ecologist measures wing length, tarsus length, bill depth, and body mass on each individual.

1. How many dimensions does this trait space have?
2. Can you visualise this space directly? If not, what strategies might help you understand the distribution of individuals?
3. If you added bill width as a fifth trait, how would the dimensionality change?

Exercise 0.3 (Phenotype as position). Consider two fish: Fish 1 has length 15 cm and mass 50 g; Fish 2 has length 20 cm and mass 80 g.

1. Represent each fish as a point in (length, mass) space.
2. Draw the arrow from Fish 1 to Fish 2. What does this arrow represent biologically?
3. If a third fish lies exactly halfway along this arrow, what are its length and mass?

Exercise 0.4 (The meaning of “distance” in trait space). Two flowers differ in petal length by 2 mm and in petal width by 3 mm.

1. What is the straight-line (Euclidean) distance between them in trait space?
2. Does this number have a direct biological interpretation?
3. What might make two flowers “far apart” in trait space but similar in fitness?

Exercises I

Exercise 11.1 (Scale dependence). A data set contains height (in metres) and weight (in kg):

Individual	Height (m)	Weight (kg)
A	1.70	70
B	1.75	72
C	1.80	90

1. Compute the Euclidean distance from A to B and from A to C.
2. Convert height to centimetres. Recompute both distances.
3. Which distance changed more? Why?
4. Propose a way to make the distance independent of measurement units.

Exercise 11.2 (Ignoring correlation). Imagine a bivariate distribution where traits X and Y are strongly positively correlated ($r = 0.95$). Most individuals lie near the line $Y = X$.

1. Sketch this distribution as an elliptical cloud.
2. Mark two points that are equidistant (in Euclidean terms) from the centre: one along the major axis, one along the minor axis.
3. Which point is more “unusual” given the shape of the distribution?
4. Why does Euclidean distance fail to capture this?

Exercise 11.3 (A concrete failure). Consider a population where leaf length (L) and leaf width (W) have:

- Mean: $\bar{L} = 10 \text{ cm}$, $\bar{W} = 5 \text{ cm}$
- Standard deviation: $s_L = 2 \text{ cm}$, $s_W = 1 \text{ cm}$
- Correlation: $r = 0.8$

Three individuals are measured:

- Plant X: $(L, W) = (12, 6)$
- Plant Y: $(L, W) = (10, 7)$
- Plant Z: $(L, W) = (14, 5)$

1. Compute the Euclidean distance of each plant from the mean.
2. Standardise each trait (subtract mean, divide by SD) and recompute distances.
3. Which plant is most unusual given the correlation structure? (Hint: think about which point lies furthest from the major axis of the ellipse.)

Exercise 11.4 (When Euclidean distance works). Under what conditions would Euclidean distance be a reasonable measure of dissimilarity?

1. List two conditions on the traits.
2. Give a biological example where these conditions might hold.
3. Give a biological example where they clearly do not hold.

Chapter 6

Covariance and Mahalanobis Distance

The previous chapter showed three ways that Euclidean distance fails: it depends on measurement units, it ignores correlations, and it does not correspond to probability. We previewed the solution—ellipses that match the shape of the data—but did not derive it.

This chapter develops that solution. We will see how the covariance matrix enters the distance formula, why the *inverse* of the covariance matrix appears, and what it means geometrically. By the end, you will understand the Mahalanobis distance not as an arbitrary formula but as the natural way to measure “how unusual” a phenotype is.

6.1 The key insight: a matrix between the vectors

Recall from Chapter 1 that the squared Euclidean distance between two points \mathbf{z}_i and \mathbf{z}_j can be written as

$$d_{\text{Euc}}^2 = (\mathbf{z}_j - \mathbf{z}_i)^\top (\mathbf{z}_j - \mathbf{z}_i).$$

This is just the dot product of the difference vector with itself—the sum of squared components.

Now consider what happens if we insert a matrix \mathbf{M} between the transpose and the vector:

$$d_{\mathbf{M}}^2 = (\mathbf{z}_j - \mathbf{z}_i)^\top \mathbf{M} (\mathbf{z}_j - \mathbf{z}_i).$$

This is still a quadratic form. It still takes a vector and returns a non-negative number (provided \mathbf{M} is positive definite). But the matrix \mathbf{M} changes how different directions are weighted.

Key Idea

Inserting a matrix into the distance formula changes the shape of the “unit ball.” With the identity matrix, the unit ball is a sphere. With a general positive definite matrix, the unit ball becomes an ellipsoid.

The question is: which matrix \mathbf{M} should we use?

6.2 The covariance matrix and its inverse

If the problem with Euclidean distance is that it ignores the covariance structure of the data, then the solution should involve the covariance matrix Σ .

But should we use Σ itself, or its inverse Σ^{-1} ?

Consider what we want. We want deviations in high-variance directions to count *less* (because they are common) and deviations in low-variance directions to count *more* (because they are rare). This means we want to *downweight* directions of large variance and *upweight* directions of small variance.

The covariance matrix Σ has large eigenvalues in directions of large variance. Its inverse Σ^{-1} has *small* eigenvalues in those same directions (since the eigenvalues of the inverse are the reciprocals of the original eigenvalues).

Therefore, to downweight high-variance directions, we use the inverse:

$$d_{\text{Mah}}^2 = (\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu}).$$

This is the **Mahalanobis distance distance** (squared) from the point \mathbf{z} to the mean $\boldsymbol{\mu}$.

Key Idea

The Mahalanobis distance distance uses the *inverse* covariance matrix because we want to penalise deviations in low-variance directions more heavily than deviations in high-variance directions.

6.3 A one-dimensional sanity check

Before tackling multiple traits, let us verify that the formula makes sense in one dimension.

With a single trait, the covariance matrix is just the variance: $\Sigma = \sigma^2$. Its inverse is $1/\sigma^2$. The Mahalanobis distance distance from a value z to the mean μ is

$$d_{\text{Mah}}^2 = (z - \mu)^\top \cdot \frac{1}{\sigma^2} \cdot (z - \mu) = \frac{(z - \mu)^2}{\sigma^2}.$$

Taking the square root:

$$d_{\text{Mah}} = \frac{|z - \mu|}{\sigma}.$$

This is simply the number of standard deviations from the mean—the familiar z -score! The Mahalanobis distance distance generalises the z -score to multiple dimensions.

Key Idea

In one dimension, the Mahalanobis distance distance equals the absolute z -score. In multiple dimensions, it generalises this idea by accounting for both variances and covariances.

6.4 Geometry: how the inverse reshapes space

Let us see geometrically why the inverse covariance matrix produces ellipses that match the data.

Suppose the covariance matrix is

$$\Sigma = \begin{pmatrix} 4 & 0 \\ 0 & 1 \end{pmatrix}.$$

Trait 1 has variance 4 (standard deviation 2), and trait 2 has variance 1 (standard deviation 1). There is no correlation, so the ellipse is aligned with the axes (Fig. 6.1).

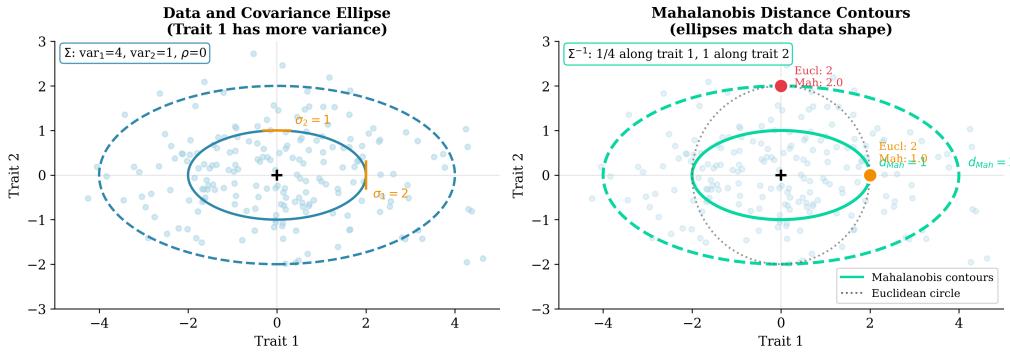


Figure 6.1: Left: The covariance ellipse shows the shape of variation. Trait 1 (horizontal) has more variance. Right: Using Σ^{-1} in the distance formula produces contours that are circles in the *standardised space*—ellipses matching the data in the original space.

The inverse covariance matrix is

$$\Sigma^{-1} = \begin{pmatrix} 1/4 & 0 \\ 0 & 1 \end{pmatrix}.$$

When we compute $\mathbf{v}^\top \Sigma^{-1} \mathbf{v}$, the component along trait 1 is divided by 4, while the component along trait 2 is left unchanged. This shrinks distances in the high-variance direction and leaves distances in the low-variance direction alone.

The result: a deviation of 2 units in trait 1 contributes the same to the Mahalanobis distance distance as a deviation of 1 unit in trait 2. Both represent one standard deviation from the mean.

6.5 The formula in components

For two traits with covariance matrix

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix},$$

the inverse is

$$\Sigma^{-1} = \frac{1}{\sigma_1^2\sigma_2^2(1-\rho^2)} \begin{pmatrix} \sigma_2^2 & -\rho\sigma_1\sigma_2 \\ -\rho\sigma_1\sigma_2 & \sigma_1^2 \end{pmatrix}.$$

The squared Mahalanobis distance distance from a point (z_1, z_2) to the mean (μ_1, μ_2) is

$$d_{\text{Mah}}^2 = \frac{1}{1-\rho^2} \left[\left(\frac{z_1 - \mu_1}{\sigma_1} \right)^2 - 2\rho \left(\frac{z_1 - \mu_1}{\sigma_1} \right) \left(\frac{z_2 - \mu_2}{\sigma_2} \right) + \left(\frac{z_2 - \mu_2}{\sigma_2} \right)^2 \right].$$

This formula has three parts:

1. The squared z -scores for each trait: $\left(\frac{z_1 - \mu_1}{\sigma_1} \right)^2$ and $\left(\frac{z_2 - \mu_2}{\sigma_2} \right)^2$.
2. A cross-term that subtracts when the correlation is positive and both deviations have the same sign (reducing distance for points along the correlation) or adds when they have opposite signs (increasing distance for points against the correlation).
3. A factor $\frac{1}{1-\rho^2}$ that inflates everything when correlation is high, reflecting the reduced “effective dimensionality” of the data.

When $\rho = 0$, the cross-term vanishes and we get

$$d_{\text{Mah}}^2 = \left(\frac{z_1 - \mu_1}{\sigma_1} \right)^2 + \left(\frac{z_2 - \mu_2}{\sigma_2} \right)^2,$$

which is just the sum of squared z -scores—Euclidean distance in standardised space.

6.6 Mahalanobis distance distance and probability

For multivariate normal data, the Mahalanobis distance distance has a direct connection to probability.

If \mathbf{z} follows a p -variate normal distribution with mean $\boldsymbol{\mu}$ and covariance Σ , then the squared Mahalanobis distance distance

$$d_{\text{Mah}}^2 = (\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu})$$

follows a chi-squared distribution with p degrees of freedom.

This means:

- Points with $d_{\text{Mah}}^2 < \chi_{p,0.95}^2$ lie within the 95% probability ellipse.
- The probability of observing a point at least as extreme as \mathbf{z} can be computed directly from the chi-squared distribution.
- Contours of equal Mahalanobis distance distance are contours of equal probability density.

Key Idea

For multivariate normal data, Mahalanobis distance distance is directly tied to probability. Equal Mahalanobis distance distance means equal probability density. This is exactly what we wanted from a “proper” distance metric.

6.7 A worked example

Let us compute Mahalanobis distance distances for the three points from Chapter 11. The covariance matrix was

$$\Sigma = \begin{pmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{pmatrix}, \quad \Sigma^{-1} = \begin{pmatrix} 2.778 & -2.222 \\ -2.222 & 2.778 \end{pmatrix}.$$

Consider three points, all at Euclidean distance 1 from the origin:

Point E: $(1/\sqrt{2}, 1/\sqrt{2}) \approx (0.707, 0.707)$. This point lies along the major axis of the ellipse (both traits elevated together).

$$\begin{aligned} d_{\text{Mah}}^2 &= (0.707 \ 0.707) \begin{pmatrix} 2.778 & -2.222 \\ -2.222 & 2.778 \end{pmatrix} \begin{pmatrix} 0.707 \\ 0.707 \end{pmatrix} \\ &= (0.707 \ 0.707) \begin{pmatrix} 0.393 \\ 0.393 \end{pmatrix} \\ &= 0.556. \end{aligned}$$

So $d_{\text{Mah}} = \sqrt{0.556} \approx 0.75$.

Point F: $(1/\sqrt{2}, -1/\sqrt{2}) \approx (0.707, -0.707)$. This point lies along the minor axis (traits in opposition).

$$\begin{aligned} d_{\text{Mah}}^2 &= (0.707 \ -0.707) \begin{pmatrix} 2.778 & -2.222 \\ -2.222 & 2.778 \end{pmatrix} \begin{pmatrix} 0.707 \\ -0.707 \end{pmatrix} \\ &= (0.707 \ -0.707) \begin{pmatrix} 3.536 \\ -3.536 \end{pmatrix} \\ &= 5.0. \end{aligned}$$

So $d_{\text{Mah}} = \sqrt{5.0} \approx 2.24$.

Point G: $(1, 0)$. This point lies along the first trait axis.

$$\begin{aligned} d_{\text{Mah}}^2 &= (1 \ 0) \begin{pmatrix} 2.778 & -2.222 \\ -2.222 & 2.778 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= 2.778. \end{aligned}$$

So $d_{\text{Mah}} = \sqrt{2.778} \approx 1.67$.

Point	Direction	Euclidean	Mahalanobis distance
E	Along correlation	1.0	0.75
F	Against correlation	1.0	2.24
G	Trait 1 only	1.0	1.67

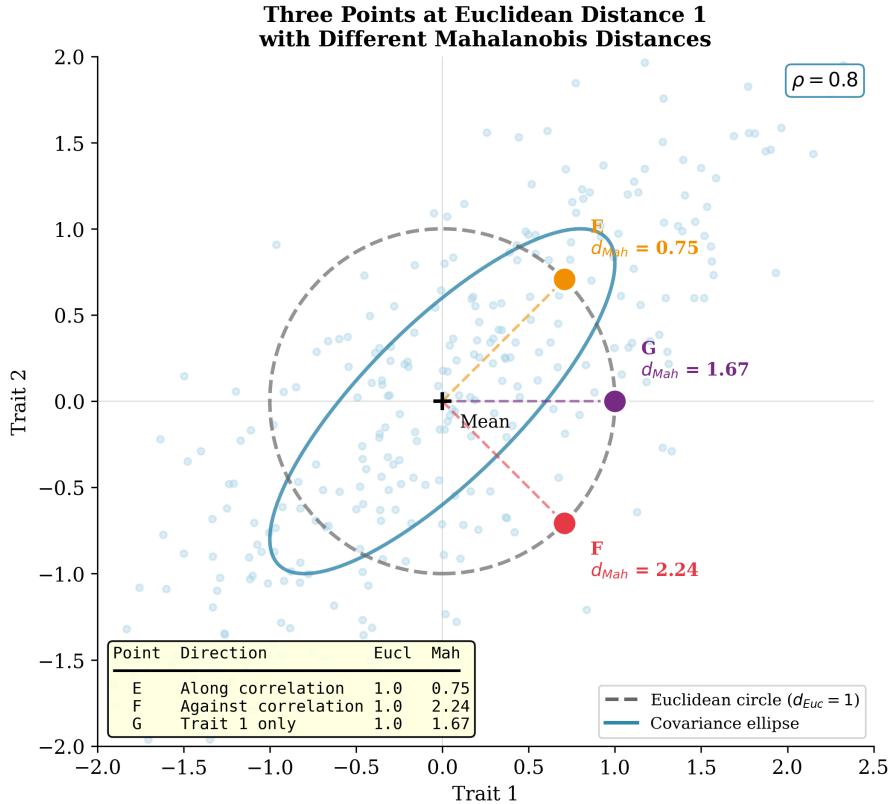


Figure 6.2: Three points at Euclidean distance 1 from the origin have very different Mahalanobis distance distances: E (along correlation) is closest, F (against correlation) is farthest, G is intermediate.

Point E, lying in the direction of maximum variance, is the *least* unusual—Mahalanobis distance distance is less than 1. Point F, lying in the direction of minimum variance, is the *most* unusual—Mahalanobis distance distance exceeds 2. The Mahalanobis distance distance correctly identifies which points are rare and which are common.

6.8 The Mahalanobis distance as a transformation

There is another way to understand Mahalanobis distance distance: as Euclidean distance after a particular transformation.

Any positive definite matrix Σ can be factored as

$$\Sigma = \Sigma^{1/2} \Sigma^{1/2},$$

where $\Sigma^{1/2}$ is the matrix square root (the unique positive definite matrix whose square is Σ).

Define the transformed variable

$$\mathbf{w} = \Sigma^{-1/2}(\mathbf{z} - \boldsymbol{\mu}).$$

This transformation does two things:

1. Centres the data at the origin.

2. Rescales and rotates so that the covariance matrix of \mathbf{w} is the identity matrix \mathbf{I} .

In the \mathbf{w} space, the data form a spherical cloud with unit variance in all directions and no correlations. This is called “whitening” or “sphering” the data.

Now compute the squared Euclidean length of \mathbf{w} :

$$\begin{aligned}\|\mathbf{w}\|^2 &= \mathbf{w}^\top \mathbf{w} \\ &= \left[\Sigma^{-1/2}(\mathbf{z} - \boldsymbol{\mu}) \right]^\top \left[\Sigma^{-1/2}(\mathbf{z} - \boldsymbol{\mu}) \right] \\ &= (\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1/2} \Sigma^{-1/2} (\mathbf{z} - \boldsymbol{\mu}) \\ &= (\mathbf{z} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{z} - \boldsymbol{\mu}) \\ &= d_{\text{Mah}}^2.\end{aligned}$$

Key Idea

Mahalanobis distance distance is ordinary Euclidean distance in “whitened” space — the space where the data have been transformed to have identity covariance. The transformation that achieves this is $\Sigma^{-1/2}$.

This insight will be central to Part III, where we develop whitening and the “P-sphere” as tools for understanding evolutionary constraints.

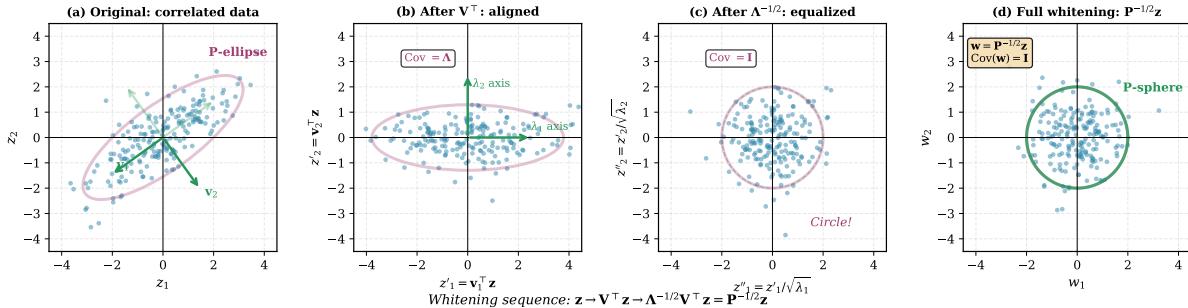


Figure 6.3: The whitening transformation step by step. (a) Original data with correlated traits; the covariance ellipse (\mathbf{P}) is tilted. (b) After rotation by \mathbf{V}^\top : data align with eigenvector axes; covariance is now diagonal ($\mathbf{\Lambda}$). (c) After scaling by $\mathbf{\Lambda}^{-1/2}$: variances are equalized; covariance becomes identity. (d) The full transformation $\mathbf{P}^{-1/2}\mathbf{z}$ produces spherical data. In this whitened space, the \mathbf{P} -sphere becomes the unit circle, and Mahalanobis distance equals Euclidean distance.

6.9 Mahalanobis distance distance between two points

So far we have measured distance from a point to the mean. But we can also measure the Mahalanobis distance distance between any two points:

$$d_{\text{Mah}}^2(\mathbf{z}_i, \mathbf{z}_j) = (\mathbf{z}_j - \mathbf{z}_i)^\top \Sigma^{-1} (\mathbf{z}_j - \mathbf{z}_i).$$

This asks: how different are these two phenotypes, measured in units that account for the population’s variance structure?

When comparing phenotypes, this is often more appropriate than asking how far each is from the mean. It tells us whether the difference between two individuals is large or small relative to typical variation in the population.

6.10 Connection to discriminant analysis

Mahalanobis distance has a natural application in classification. Suppose we have two groups (e.g., two species, or survivors versus non-survivors) with means μ_1 and μ_2 and a common within-group covariance matrix Σ_W .

To classify a new individual with phenotype \mathbf{z} , we compute the Mahalanobis distance from \mathbf{z} to each group mean:

$$\begin{aligned} d_1^2 &= (\mathbf{z} - \mu_1)^\top \Sigma_W^{-1} (\mathbf{z} - \mu_1), \\ d_2^2 &= (\mathbf{z} - \mu_2)^\top \Sigma_W^{-1} (\mathbf{z} - \mu_2), \end{aligned}$$

and assign the individual to the closer group.

This is the basis of linear discriminant analysis (LDA). The Mahalanobis distance ensures that classification respects the shape of within-group variation: a large difference along a high-variance direction counts less than a small difference along a low-variance direction.

6.11 Biological interpretation

In evolutionary biology, the Mahalanobis distance has a natural interpretation. If we use the phenotypic covariance matrix \mathbf{P} , then

$$d_{\mathbf{P}}^2 = (\mathbf{z} - \boldsymbol{\mu})^\top \mathbf{P}^{-1} (\mathbf{z} - \boldsymbol{\mu})$$

measures how unusual a phenotype is relative to the variation present in the population.

If we use the genetic covariance matrix \mathbf{G} , then

$$d_{\mathbf{G}}^2 = (\mathbf{z} - \boldsymbol{\mu})^\top \mathbf{G}^{-1} (\mathbf{z} - \boldsymbol{\mu})$$

measures how unusual a phenotype is relative to the *genetic* variation available. This is relevant for asking: how difficult would it be to evolve to this phenotype? Phenotypes far from the mean in low-genetic-variance directions are harder to reach by selection than phenotypes far from the mean in high-genetic-variance directions.

Key Idea

The choice of covariance matrix changes the question being asked:

- \mathbf{P}^{-1} : How unusual is this phenotype given the total variation in the population?
- \mathbf{G}^{-1} : How unusual is this phenotype given the genetic variation available for selection to act on?

6.12 What Mahalanobis distance requires

The Mahalanobis distance is well-defined only when the covariance matrix is invertible. This fails when:

- The covariance matrix is singular (has zero eigenvalues), meaning some linear combination of traits has zero variance.
- The sample size is smaller than the number of traits, making the sample covariance matrix rank-deficient.

In practice, researchers often use regularised covariance estimates or reduce dimensionality (e.g., via PCA) before computing Mahalanobis distances. We will discuss these issues further in Part IV.

6.13 Summary

In this chapter we have:

- Derived the Mahalanobis distance by inserting the inverse covariance matrix into the distance formula.
- Verified that in one dimension, Mahalanobis distance reduces to the absolute z -score.
- Seen geometrically how the inverse covariance matrix reshapes the unit ball from a sphere to an ellipsoid matching the data.
- Connected Mahalanobis distance to probability: for multivariate normal data, equal Mahalanobis distance means equal probability density.
- Computed a worked example showing how points at the same Euclidean distance have very different Mahalanobis distances.
- Understood Mahalanobis distance as Euclidean distance in whitened space—space where the covariance has been transformed to the identity.
- Noted the biological interpretations: \mathbf{P}^{-1} measures phenotypic unusualness, \mathbf{G}^{-1} measures genetic “difficulty to reach.”

We now have the conceptual foundation for Part III. The covariance matrix encodes the shape of the data cloud; its inverse defines a natural distance. But we have not yet said how to *find* the axes of the ellipse or compute the eigenvalues that determine its shape. That is the work of diagonalisation, which we take up next.

Exercises

Exercise 12.1 (Computing a covariance matrix). Five individuals are measured for two traits:

Individual	X	Y
1	2	4
2	3	5
3	5	7
4	4	6
5	6	8

1. Compute the mean of X and the mean of Y.
2. Compute the variance of X and the variance of Y.
3. Compute the covariance of X and Y using $\text{Cov}(X, Y) = \frac{1}{n-1} \sum_i (x_i - \bar{x})(y_i - \bar{y})$.
4. Assemble the 2×2 covariance matrix \mathbf{S} .
5. Verify that \mathbf{S} is symmetric.

Exercise 12.2 (Covariance matrix properties). Consider the covariance matrix

$$\mathbf{S} = \begin{pmatrix} 4 & 2 \\ 2 & 5 \end{pmatrix}.$$

1. What is the variance of trait 1? Of trait 2?
2. What is the covariance between the traits?
3. Compute the correlation: $r = \text{Cov}(X, Y)/(s_X s_Y)$.
4. Is this matrix positive definite? (Hint: compute its eigenvalues or check that $\det(\mathbf{S}) > 0$ and $\text{tr}(\mathbf{S}) > 0$.)

Exercise 12.3 (Mahalanobis distance by hand). Using the covariance matrix from Exercise 12.2, compute the Mahalanobis distance from the mean $(0, 0)$ to the point $(2, 1)$.

1. First, compute the inverse of \mathbf{S} .
2. Then compute $D^2 = \mathbf{x}^\top \mathbf{S}^{-1} \mathbf{x}$ where $\mathbf{x} = (2, 1)^\top$.
3. Take the square root to get D .
4. Compare to the Euclidean $\|\mathbf{x}\| = \sqrt{2^2 + 1^2}$.

Exercise 12.4 (Mahalanobis equals Euclidean when...). Show that Mahalanobis distance equals Euclidean distance when the covariance matrix is the identity matrix \mathbf{I} .

1. Write down the 2×2 identity matrix.
2. What does $\mathbf{S} = \mathbf{I}$ imply about the variances and covariance?
3. Compute $D^2 = \mathbf{x}^\top \mathbf{I}^{-1} \mathbf{x}$ and simplify.
4. Under what biological conditions might $\mathbf{S} \approx \mathbf{I}$?

Exercise 12.5 (Ellipses and probability). The set of points with Mahalanobis distance $D = 1$ from the mean forms an ellipse.

1. For a bivariate normal distribution, approximately what percentage of points lie within the $D = 1$ ellipse? (Hint: it's not 68%.)
2. How does this compare to the univariate case where about 68% of observations lie within 1 SD of the mean?
3. The $D = \sqrt{2}$ ellipse contains about 63% of observations for a bivariate normal. Why does the "1 SD" intuition not transfer directly to multiple dimensions?

Part III

Diagonalisation and Natural Axes

Chapter 7

Diagonalisation and Natural Axes

In Part II we saw that covariance matrices define ellipses and that the Mahalanobis distance distance uses the inverse covariance matrix to measure “unusualness.” But we repeatedly invoked eigenvalues and eigenvectors without explaining how to find them or what they mean.

This chapter fills that gap. We develop diagonalisation—the process of finding the natural axes of an ellipse—from first principles. By the end, you will understand eigenvalues and eigenvectors not as abstract algebra but as answers to a concrete geometric question: *in which directions does a matrix act as pure stretching?*

7.1 The question that leads to eigenvalues

Consider a symmetric matrix \mathbf{A} acting on vectors in the plane. For most vectors \mathbf{v} , the output \mathbf{Av} points in a different direction from \mathbf{v} . The matrix rotates as well as stretches.

But for some special vectors, the output points in the *same direction* as the input—or exactly opposite. The matrix stretches (or compresses) without rotating. These special vectors are called **eigenvectors**, and the stretching factors are called **eigenvalues**.

Formally, \mathbf{v} is an eigenvector of \mathbf{A} with eigenvalue λ if

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

The matrix \mathbf{A} acting on \mathbf{v} produces the same result as simply multiplying \mathbf{v} by the scalar λ .

Key Idea

eigenvectors are the directions along which a matrix acts as pure scaling. eigenvalues are the scaling factors. Finding them reveals the natural axes of the transformation.

7.2 A concrete example

Let us find the eigenvectors and eigenvalues of

$$\mathbf{A} = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}.$$

We seek vectors \mathbf{v} and scalars λ such that $\mathbf{Av} = \lambda\mathbf{v}$.

Rearranging:

$$\mathbf{Av} - \lambda\mathbf{v} = \mathbf{0} \quad \Rightarrow \quad (\mathbf{A} - \lambda\mathbf{I})\mathbf{v} = \mathbf{0}.$$

For a non-zero solution \mathbf{v} to exist, the matrix $(\mathbf{A} - \lambda\mathbf{I})$ must be singular. This happens when its determinant is zero:

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

This is the **characteristic equation**. For our matrix:

$$\det \begin{pmatrix} 3 - \lambda & 1 \\ 1 & 3 - \lambda \end{pmatrix} = (3 - \lambda)^2 - 1 = \lambda^2 - 6\lambda + 8 = (\lambda - 4)(\lambda - 2) = 0.$$

The eigenvalues are $\lambda_1 = 4$ and $\lambda_2 = 2$.

Finding the eigenvectors

For $\lambda_1 = 4$:

$$(\mathbf{A} - 4\mathbf{I})\mathbf{v} = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \mathbf{0}.$$

This gives $-v_1 + v_2 = 0$, so $v_1 = v_2$. The eigenvector (normalised to unit length) is

$$\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

For $\lambda_2 = 2$:

$$(\mathbf{A} - 2\mathbf{I})\mathbf{v} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \mathbf{0}.$$

This gives $v_1 + v_2 = 0$, so $v_2 = -v_1$. The eigenvector is

$$\mathbf{v}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Notice that \mathbf{v}_1 and \mathbf{v}_2 are perpendicular (their dot product is zero). This is not a coincidence—it is guaranteed for symmetric matrices.

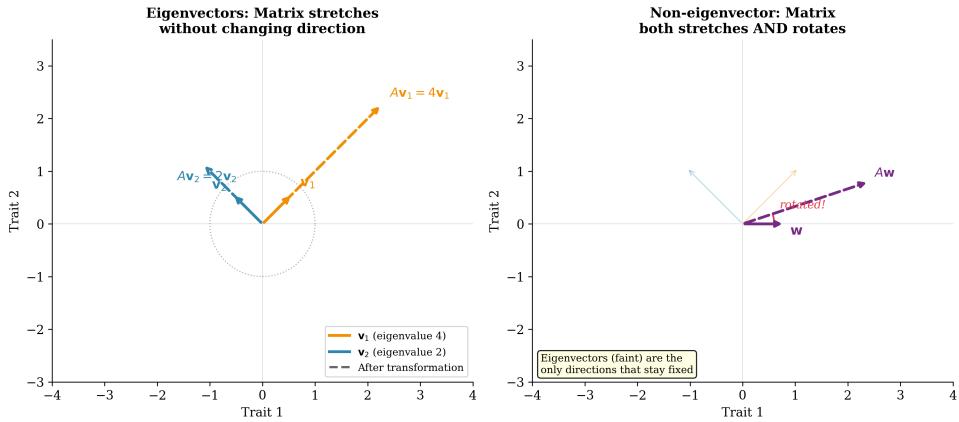


Figure 7.1: eigenvalue

- s] The matrix \mathbf{A} stretches space by factor 4 along v_1 (the diagonal where both traits increase together) and by factor 2 along v_2 (the diagonal where traits move in opposite directions). These are the natural axes of the transformation.

7.3 The spectral theorem: why symmetric matrices are special

Symmetric matrices have three remarkable properties that make them central to statistics and biology:

1. **Real eigenvalues.** The eigenvalues of a symmetric matrix are always real numbers, never complex.
2. **Orthogonal eigenvectors.** eigenvectors corresponding to different eigenvalues are perpendicular.
3. **Complete set.** A $p \times p$ symmetric matrix always has p eigenvectors that form an orthonormal basis for \mathbb{R}^p .

These properties are collectively known as the **spectral theorem**.

Key Idea

Every symmetric matrix can be understood as pure stretching along perpendicular axes. There is no rotation mixed in, no shear, no reflection—just stretching (or compressing) along p orthogonal directions.

Covariance matrices are symmetric by construction (the covariance of X with Y equals the covariance of Y with X). This is why ellipses, not parallelograms, describe their geometry.

7.4 diagonalisation: the matrix factorisation

Collect the eigenvectors as columns of a matrix \mathbf{V} :

$$\mathbf{V} = \begin{pmatrix} | & | \\ \mathbf{v}_1 & \mathbf{v}_2 \\ | & | \end{pmatrix}.$$

For our example:

$$\mathbf{V} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Collect the eigenvalues in a diagonal matrix Λ :

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}.$$

The spectral theorem guarantees that

$$\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^\top.$$

This is the **eigendecomposition** or **spectral decomposition** of \mathbf{A} .

Key Idea

The eigendecomposition $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^\top$ factorises a symmetric matrix into three parts:

- \mathbf{V}^\top : rotate from original axes to eigenvector axes;
- Λ : stretch along each eigenvector axis;
- \mathbf{V} : rotate back to original axes.

Let us verify this for our example:

$$\begin{aligned} \mathbf{V}\Lambda\mathbf{V}^\top &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 4 & 4 \\ 2 & -2 \end{pmatrix} \\ &= \frac{1}{2} \begin{pmatrix} 6 & 2 \\ 2 & 6 \end{pmatrix} = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix} = \mathbf{A}. \quad \checkmark \end{aligned}$$

7.5 Geometric interpretation: the ellipse revealed

Now we can see exactly what a covariance matrix describes geometrically.

If Σ is a covariance matrix with eigendecomposition $\Sigma = \mathbf{V}\Lambda\mathbf{V}^\top$, then:

- The **eigenvectors** $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p$ are the directions of the principal axes of the covariance ellipse.
- The **eigenvalues** $\lambda_1, \lambda_2, \dots, \lambda_p$ are the variances along those axes.
- The **semi-axis lengths** of the ellipse are $\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots, \sqrt{\lambda_p}$.

For the covariance matrix from Chapter 12,

$$\Sigma = \begin{pmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{pmatrix},$$

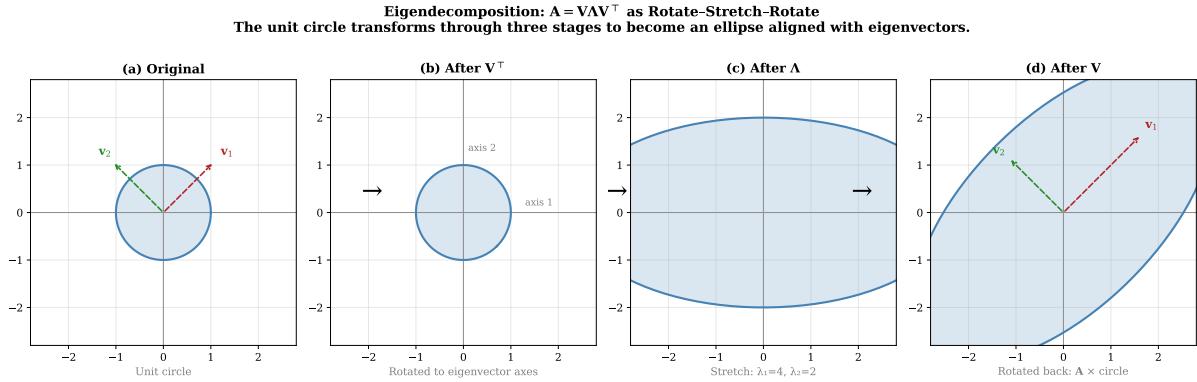


Figure 7.2: **The eigendecomposition as rotate–stretch–rotate.** The matrix $A = V\Lambda V^\top$ acts in three stages. (a) Original unit circle with eigenvectors marked (dashed). (b) Multiply by V^\top : rotate the coordinate system so that the eigenvectors align with the axes. (c) Multiply by Λ : stretch by $\lambda_1 = 4$ along the first axis and $\lambda_2 = 2$ along the second. (d) Multiply by V : rotate back to the original coordinates. The unit circle becomes an ellipse whose axes align with the eigenvectors and whose semi-axis lengths are $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$.

the eigenvalues are $\lambda_1 = 1.8$ and $\lambda_2 = 0.2$. The ratio $\lambda_1 / \lambda_2 = 9$ tells us the ellipse is elongated: its major axis is three times longer than its minor axis ($\sqrt{9} = 3$).

The eigenvectors are at 45° angles to the original axes—one along the direction where both traits increase together (the correlation direction), and one perpendicular to it.

7.6 Why diagonalisation simplifies everything

In the eigenvector coordinate system, the covariance matrix becomes diagonal:

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_p \end{pmatrix}.$$

A diagonal matrix is trivial to work with:

- **Inverse:** Λ^{-1} has entries $1/\lambda_1, 1/\lambda_2, \dots$
- **Square root:** $\Lambda^{1/2}$ has entries $\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots$
- **Powers:** Λ^k has entries $\lambda_1^k, \lambda_2^k, \dots$
- **Determinant:** $\det(\Lambda) = \lambda_1 \lambda_2 \cdots \lambda_p$
- **Trace:** $\text{tr}(\Lambda) = \lambda_1 + \lambda_2 + \cdots + \lambda_p$

Since $\Sigma = V\Lambda V^\top$, these operations extend to the original matrix. For example:

$$\Sigma^{-1} = V\Lambda^{-1}V^\top, \quad \Sigma^{1/2} = V\Lambda^{1/2}V^\top.$$

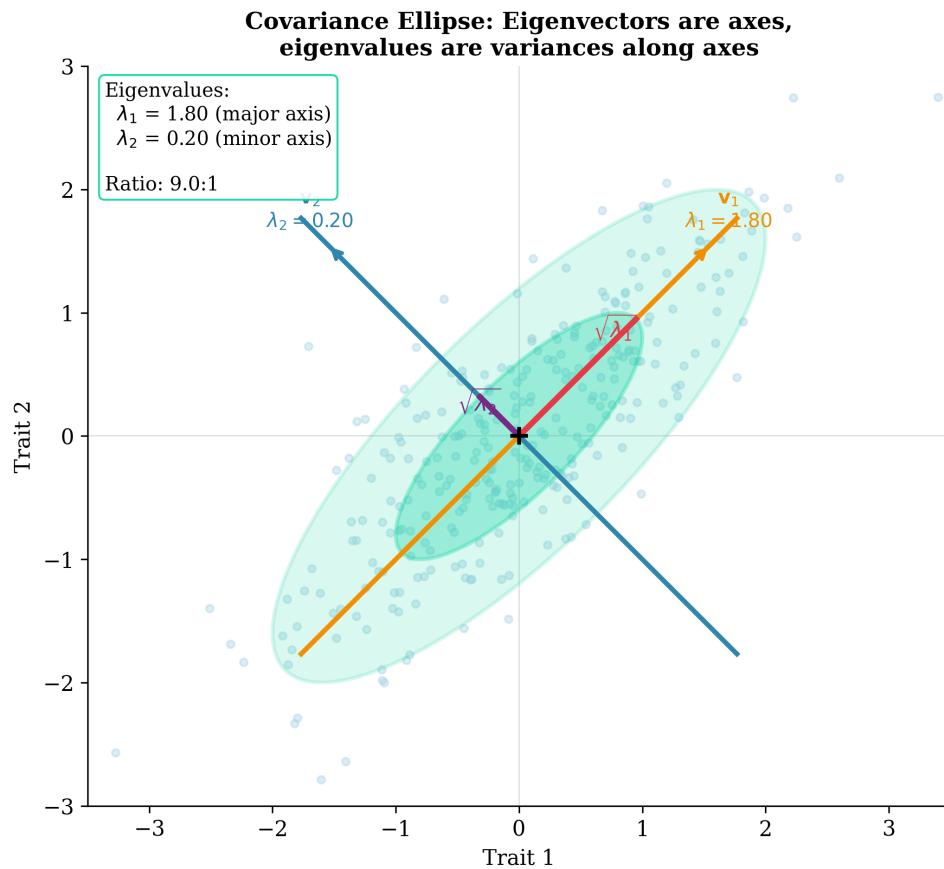


Figure 7.3: eigenvalue

s and eigenvectors of a covariance matrix] A covariance matrix defines an ellipse. The eigenvectors point along the principal axes; the eigenvalues are the variances (squared semi-axis lengths) along those axes.

Key Idea

diagonalisation converts hard matrix problems into easy scalar problems. Once you know the eigenvalues and eigenvectors, operations like inversion, taking square roots, and computing powers become simple.

7.7 The trace and determinant as summaries

Two numbers summarise the eigenvalue spectrum:

The trace. The trace of a matrix is the sum of its diagonal entries, which equals the sum of its eigenvalues:

$$\text{tr}(\Sigma) = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_p^2 = \lambda_1 + \lambda_2 + \cdots + \lambda_p.$$

For a covariance matrix, the trace is the **total variance**—the sum of variances across all traits. Geometrically, it measures the overall “size” of the ellipse.

The determinant. The determinant is the product of the eigenvalues:

$$\det(\Sigma) = \lambda_1 \lambda_2 \cdots \lambda_p.$$

Geometrically, the determinant is the squared volume of the ellipsoid. For a covariance matrix, it measures **generalised variance**—a single number capturing the overall spread, accounting for correlations.

If any eigenvalue is zero, the determinant is zero, indicating that the data lie in a lower-dimensional subspace. The matrix is then singular and cannot be inverted.

7.8 Variance in any direction: the quadratic form revisited

In Chapter 2 we introduced the quadratic form $\mathbf{v}^\top \Sigma \mathbf{v}$. Now we can interpret it precisely.

Let β be a unit vector representing a direction in trait space. The variance of the population in that direction is

$$\sigma_\beta^2 = \beta^\top \Sigma \beta.$$

Using the eigendecomposition $\Sigma = \mathbf{V} \Lambda \mathbf{V}^\top$:

$$\begin{aligned}\beta^\top \Sigma \beta &= \beta^\top \mathbf{V} \Lambda \mathbf{V}^\top \beta \\ &= (\mathbf{V}^\top \beta)^\top \Lambda (\mathbf{V}^\top \beta).\end{aligned}$$

Let $\mathbf{c} = \mathbf{V}^\top \beta$ be the coordinates of β in the eigenvector basis. Then

$$\beta^\top \Sigma \beta = \mathbf{c}^\top \Lambda \mathbf{c} = \sum_{i=1}^p \lambda_i c_i^2.$$

Key Idea

The variance in direction β is a weighted average of the eigenvalues, with weights $c_i^2 = (\beta \cdot \mathbf{v}_i)^2$ —the squared projections of β onto the eigenvectors.

This formula explains why:

- Variance is maximised ($= \lambda_1$) when β aligns with \mathbf{v}_1 , the first eigenvector.
- Variance is minimised ($= \lambda_p$) when β aligns with \mathbf{v}_p , the last eigenvector.
- For any other direction, variance is intermediate.

The eigenvalues bound the variance: $\lambda_{\min} \leq \beta^\top \Sigma \beta \leq \lambda_{\max}$.

7.9 Principal Component Analysis (PCA) in one paragraph

PCA is simply this: project the data onto the eigenvectors of the covariance matrix. The first principal component is the projection onto \mathbf{v}_1 ; it captures the direction of maximum variance.

The second principal component is the projection onto \mathbf{v}_2 ; it captures the most variance in the subspace orthogonal to the first. And so on.

The eigenvalues tell you how much variance each component captures. If λ_1 is much larger than the others, the first principal component carries most of the information, and the data are approximately one-dimensional despite having p measured traits.

We will explore PCA and related methods in Chapter 32.

7.10 Computing eigenvalues and eigenvectors in practice

For 2×2 matrices, you can solve the characteristic equation by hand. For larger matrices, use numerical algorithms. In R:

```
Sigma <- matrix(c(1.0, 0.8, 0.8, 1.0), nrow = 2)
eig <- eigen(Sigma)
eig$values      # eigenvalues
eig$vectors    # eigenvectors (columns)
```

In Python:

```
import numpy as np
Sigma = np.array([[1.0, 0.8], [0.8, 1.0]])
eigenvalues, eigenvectors = np.linalg.eigh(Sigma)
```

Note: `eigh` is for symmetric (Hermitian) matrices and guarantees real eigenvalues and orthogonal eigenvectors. Use it for covariance matrices.

7.11 A biological example: the G matrix

The additive genetic covariance matrix \mathbf{G} describes the genetic architecture underlying multiple traits. Its eigendecomposition reveals:

- **\mathbf{g}_{\max} :** The first eigenvector, pointing in the direction of maximum genetic variance. This is the “line of least resistance”—the direction evolution finds easiest.
- **Higher eigenvectors:** Directions of progressively less genetic variance. Evolution in these directions requires stronger selection to achieve the same response.
- **eigenvalues:** The genetic variances along each principal axis. Large eigenvalue ratios indicate that \mathbf{G} strongly channels evolution along certain directions.

When the \mathbf{G} matrix is highly eccentric (eigenvalues very unequal), the population can respond quickly to selection along \mathbf{g}_{\max} but responds sluggishly—or not at all—to selection perpendicular to it. This is the geometric foundation of evolutionary constraint.

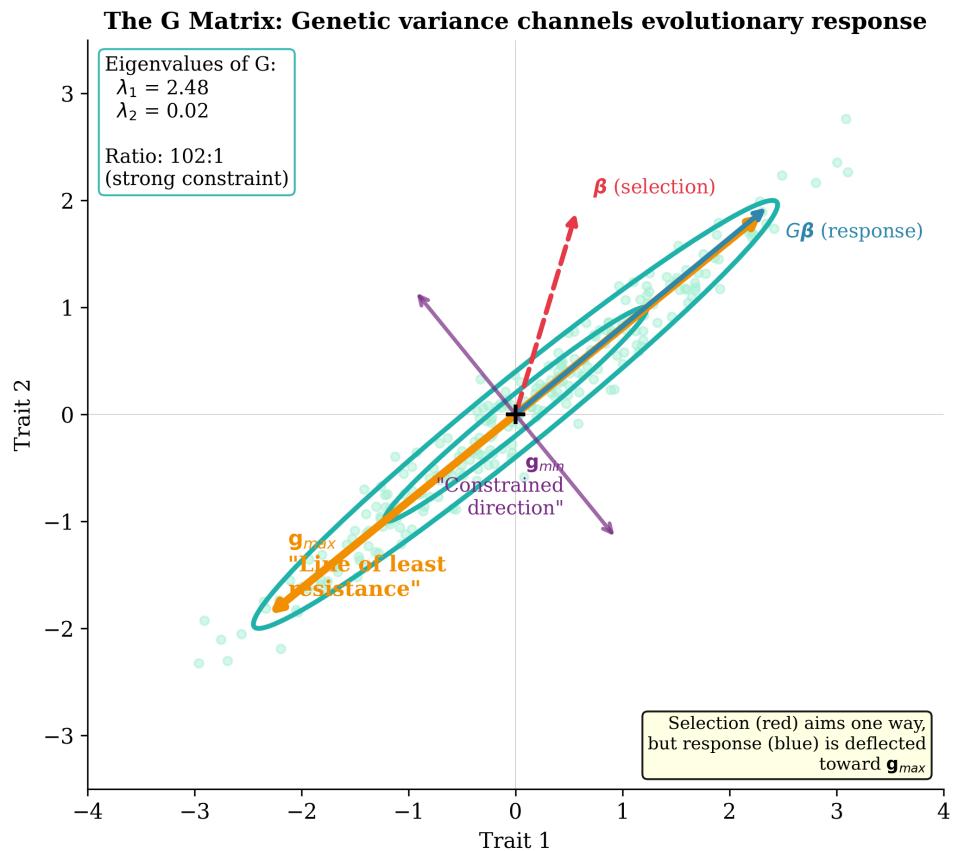


Figure 7.4: The G matrix defines a genetic ellipse. The first eigenvector g_{max} points along the direction of maximum genetic variance—the “line of least evolutionary resistance.”

7.12 positive definiteness and what eigenvalues tell us

A matrix is **positive definite** if all its eigenvalues are strictly positive. For covariance matrices, this means:

- Every direction has positive variance.
- The matrix can be inverted.
- The ellipse is a proper ellipse, not degenerate.

A matrix is **positive semi-definite** if all eigenvalues are non-negative (some may be zero). A zero eigenvalue means:

- Some linear combination of traits has zero variance.
- The data lie in a lower-dimensional subspace.
- The matrix cannot be inverted (it is singular).

In practice, estimated covariance matrices from finite samples may have small or even slightly negative eigenvalues due to sampling error. This can cause numerical problems and may require regularisation.

7.13 The condition number: how “ill-behaved” is the matrix?

The ratio of largest to smallest eigenvalue is the **condition number**:

$$\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}.$$

A large condition number indicates:

- The ellipse is highly elongated.
- The matrix is nearly singular.
- Numerical computations (like inversion) may be unstable.
- Small errors in estimating the matrix can cause large errors in derived quantities.

In evolutionary terms, a G matrix with large condition number channels evolution strongly along certain directions. In statistical terms, it makes estimation difficult.

7.14 Summary

In this chapter we have:

- Defined eigenvalues and eigenvectors as the answers to “in which directions does a matrix act as pure scaling?”
- Worked through a complete example: finding eigenvalues from the characteristic equation, then finding eigenvectors.
- Stated the spectral theorem: symmetric matrices have real eigenvalues and orthogonal eigenvectors.
- Introduced the eigendecomposition $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^\top$ and interpreted it as rotate–stretch–rotate–back.
- Connected eigenvalues to ellipse geometry: eigenvectors are axes, eigenvalues are variances along those axes.
- Showed that variance in any direction is a weighted average of eigenvalues, with weights given by squared projections.
- Applied these ideas to the G matrix: \mathbf{g}_{\max} is the direction of maximum genetic variance, and eigenvalue ratios quantify constraint.

We now have the tools to understand any symmetric matrix geometrically. In the next chapter, we use diagonalisation to construct the whitening transformation—the key to understanding directional heritability and the P-sphere.

Exercises

Exercise 32.1 (PCA by hand). Consider the covariance matrix

$$\mathbf{S} = \begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix}.$$

1. Find the eigenvalues of \mathbf{S} .
2. Find the eigenvectors of \mathbf{S} (normalised to unit length).
3. What proportion of total variance does PC1 explain?
4. Interpret PC1 and PC2 in terms of the original traits.

Exercise 32.2 (When all loadings are positive). In many morphological data sets, PC1 has all positive loadings.

1. What biological interpretation does this suggest?
2. Give an example of a trait set where you would *not* expect all PC1 loadings to be positive.
3. If PC1 is “size,” what does PC2 typically represent?

Exercise 32.3 (Covariance vs. correlation PCA). A data set has three traits with very different variances:

- Trait A: variance = 100
 - Trait B: variance = 10
 - Trait C: variance = 1
1. If you run PCA on the covariance matrix, which trait will dominate PC1?
 2. What happens if you standardise each trait to unit variance before computing PCA (equivalently, PCA on the correlation matrix)?
 3. When is covariance-based PCA appropriate? When is correlation-based PCA better?

Exercise 32.4 (MANOVA intuition). Two species of iris are measured for sepal length and sepal width. The within-species variation forms two overlapping ellipses; the between-species variation is the distance between their centroids.

1. Sketch this scenario with two partially overlapping ellipses.
2. MANOVA compares the “size” of between-group variation to within-group variation. In your sketch, are the groups well separated?
3. How would the separation change if the within-group ellipses were narrower?
4. How would it change if the centroids were farther apart?

Exercise 32.5 (Discriminant analysis). Using the iris scenario from Exercise 32.4:

1. The first discriminant function (DF1) is the direction that maximises between-group variance relative to within-group variance. Sketch the direction of DF1 on your diagram.
2. If you project all individuals onto DF1, what do you expect the resulting univariate distributions to look like?
3. How does DF1 relate to the eigenvectors of $\mathbf{W}^{-1}\mathbf{B}$?
4. If there are g groups, what is the maximum number of non-trivial discriminant functions?

Exercise 32.6 (Scree plots). A PCA of 10 traits yields eigenvalues: 5.2, 2.1, 1.0, 0.6, 0.4, 0.3, 0.2, 0.1, 0.08, 0.02.

1. Compute the proportion of variance explained by each PC.
2. Compute the cumulative proportion of variance.
3. Sketch a scree plot (eigenvalue vs. PC number).
4. How many PCs would you retain? Justify your choice.
5. What percentage of total variance do your retained PCs explain?

Chapter 8

Whitening and the P-sphere

In the previous chapter we learned to diagonalise symmetric matrices and interpret eigenvalues geometrically. Now we apply these tools to a specific problem: how do we compare genetic and phenotypic variation across directions in trait space?

This chapter introduces the whitening transformation and the concept of the P-sphere. These ideas unify several threads from earlier chapters and set the stage for understanding directional heritability and evolutionary constraint.

8.1 The problem: comparing \mathbf{G} and \mathbf{P}

Quantitative genetics gives us two fundamental matrices:

- The genetic covariance matrix \mathbf{G} , describing heritable variation.
- The phenotypic covariance matrix \mathbf{P} , describing total observed variation (genetic plus environmental).

For a single trait, heritability is the ratio $h^2 = V_G/V_P$. But with multiple traits, both \mathbf{G} and \mathbf{P} are matrices, not scalars. How do we generalise heritability to multiple dimensions?

One approach is to pick a direction β in trait space and ask: what fraction of phenotypic variance in that direction is genetic? This gives us the **directional heritability**:

$$h^2(\beta) = \frac{\beta^\top \mathbf{G} \beta}{\beta^\top \mathbf{P} \beta}.$$

The numerator is the genetic variance in direction β ; the denominator is the phenotypic variance in the same direction. Their ratio is a number between 0 and 1 (assuming \mathbf{G} and \mathbf{P} are properly estimated).

Key Idea

Directional heritability $h^2(\beta)$ measures what fraction of phenotypic variance is genetic along a specific direction in trait space. It generalises the scalar heritability $h^2 = V_G/V_P$ to multiple traits.

But here is the difficulty: the value of $h^2(\beta)$ depends on which direction we choose. Some directions may have high heritability (most variation is genetic), while others have low heritability (most variation is environmental). How do we summarise this variation across directions? And how do we sample directions “fairly”?

8.2 The naive approach and its problem

A natural idea is to sample directions uniformly from the unit sphere—all directions equally likely—and compute $h^2(\beta)$ for each.

But “uniform on the unit sphere” is ambiguous when traits have different scales. Consider two traits: body mass in kilograms and wing length in millimetres. A “uniform” sample in the original coordinates would be dominated by directions that emphasise the trait with larger numerical values.

Even after standardising each trait to have unit variance, there is still a problem. If traits are correlated, the phenotypic covariance matrix \mathbf{P} is not the identity. The directions that “look uniform” in Euclidean distance space are not uniform with respect to phenotypic variation.

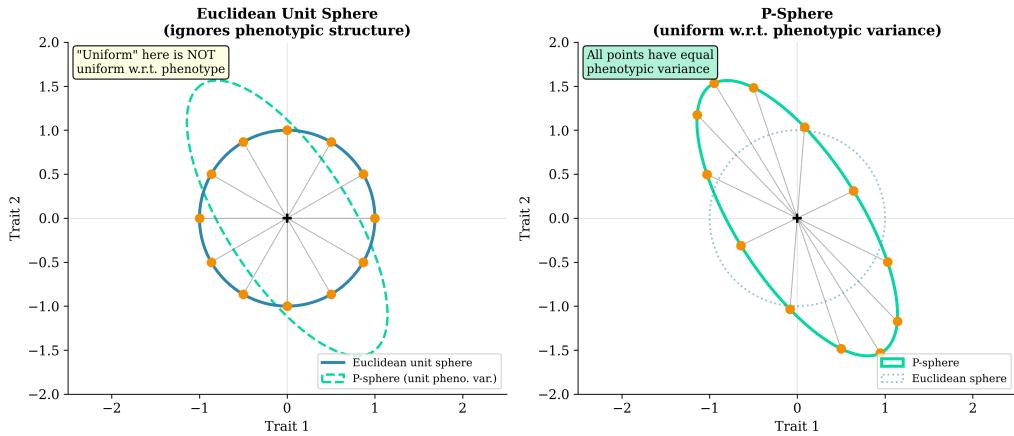


Figure 8.1: distance!Euclidean
sphere versus the P-sphere] Left: The unit sphere in original coordinates. Uniform sampling here ignores the correlation structure. Right: The P-sphere, where points are equidistant in Mahalanobis distance distance. Uniform sampling on the P-sphere respects the phenotypic covariance structure.

8.3 The P-sphere: uniform with respect to phenotype

The solution is to define “uniform” with respect to the phenotypic covariance matrix. Instead of the Euclidean distance unit sphere

$$\{\beta : \|\beta\|^2 = 1\} = \{\beta : \beta^\top \beta = 1\},$$

we use the **P-sphere**:

$$\{\beta : \beta^\top \mathbf{P}\beta = 1\}.$$

Points on the P-sphere all have unit phenotypic variance. This is the natural normalisation for comparing directions: we are asking “per unit of phenotypic variance, how much is genetic?”

The P-sphere is an ellipsoid in the original coordinate system, but it becomes a true sphere after the whitening transformation.

Key Idea

The P-sphere is the set of all directions with unit phenotypic variance. Sampling uniformly from the P-sphere means treating all phenotypically equivalent directions equally.

8.4 The whitening transformation

In Chapter 12 we saw that the Mahalanobis distance distance can be understood as Euclidean distance distance after a whitening transformation. Now we develop this idea systematically.

The whitening transformation uses the matrix square root of \mathbf{P}^{-1} . Define

$$\mathbf{P}^{-1/2} = \mathbf{V}_P \Lambda_P^{-1/2} \mathbf{V}_P^\top,$$

where \mathbf{V}_P contains the eigenvectors of \mathbf{P} and Λ_P is the diagonal matrix of eigenvalues. The matrix $\Lambda_P^{-1/2}$ has entries $1/\sqrt{\lambda_i}$ on the diagonal.

Apply this transformation to both the genetic and phenotypic matrices:

$$\begin{aligned}\mathbf{P}^* &= \mathbf{P}^{-1/2} \mathbf{P} \mathbf{P}^{-1/2} = \mathbf{I}, \\ \mathbf{G}^* &= \mathbf{P}^{-1/2} \mathbf{G} \mathbf{P}^{-1/2}.\end{aligned}$$

The phenotypic matrix becomes the identity—this is what “whitening” means. The genetic matrix becomes \mathbf{G}^* , sometimes called the **P-standardised genetic matrix** or the **G-P matrix**.

Key Idea

whitening by $\mathbf{P}^{-1/2}$ transforms the phenotypic matrix to the identity. In whitened space, the P-sphere becomes the ordinary unit sphere, and uniform sampling is straightforward.

8.5 A remarkable fact: eigenvalues of \mathbf{G}^* are directional heritabilities

Here is the key result. In whitened coordinates, let β^* be a unit vector (on the ordinary sphere, which is now also the P-sphere). The directional heritability is

$$\begin{aligned}h^2(\beta^*) &= \frac{(\beta^*)^\top \mathbf{G}^* \beta^*}{(\beta^*)^\top \mathbf{I} \beta^*} \\ &= (\beta^*)^\top \mathbf{G}^* \beta^*.\end{aligned}$$

This is a quadratic form in \mathbf{G}^* . From Chapter 20, we know that:

- The maximum value is the largest eigenvalue of \mathbf{G}^* .

- The minimum value is the smallest eigenvalue of \mathbf{G}^* .
- The eigenvectors of \mathbf{G}^* are the directions that achieve these extremes.

Key Idea

The eigenvalues of $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{G}\mathbf{P}^{-1/2}$ are the maximum and minimum directional heritabilities. The eigenvectors are the directions that achieve them.

This is a powerful result. It tells us that to understand the range of possible directional heritabilities, we need only diagonalise \mathbf{G}^* . The eigenvalues give us the bounds; the eigenvectors tell us where those bounds are achieved.

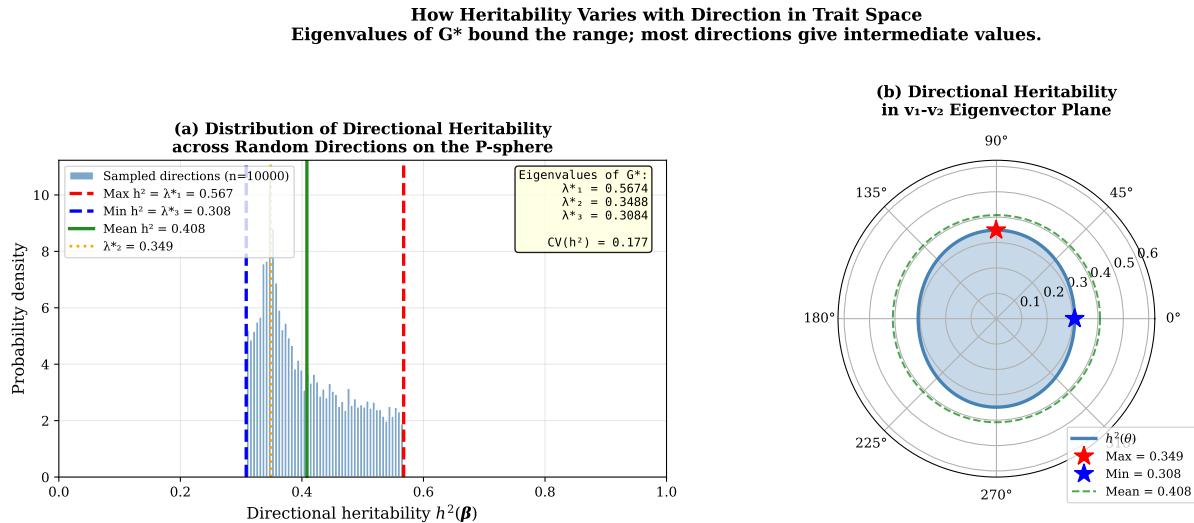


Figure 8.2: Distribution of directional heritability $h^2(\beta)$ across directions. (a) The \mathbf{G}^* ellipse (magenta) inside the P-sphere (green) in whitened space. Arrows indicate directions of maximum and minimum heritability, which are the eigenvectors of \mathbf{G}^* . (b) Histogram of h^2 values from 10,000 random directions sampled uniformly from the P-sphere. The shaded region indicates the “constraint trap zone” where heritability is well below average. (c) Heritability as a continuous function of direction angle, showing the 180° periodicity (opposite directions have identical h^2). The eigenvalues of \mathbf{G}^* bound the distribution.

8.6 The distribution of directional heritability

If we sample directions uniformly from the P-sphere (equivalently, the unit sphere in whitened space), what distribution of h^2 values do we get?

From Chapter 20, the quadratic form $(\beta^*)^\top \mathbf{G}^* \beta^*$ is a weighted average of the eigenvalues λ_i^* of \mathbf{G}^* , with weights given by squared projections onto the eigenvectors.

For uniform random directions on the sphere, there is a known formula for the variance of this quadratic form:

$$\text{Var}[h^2(\beta)] = \frac{2}{p+2} \cdot \text{Var}(\lambda^*),$$

where $\text{Var}(\lambda^*)$ is the variance of the eigenvalues of \mathbf{G}^* , and p is the number of traits.

The coefficient of variation of directional heritability is therefore

$$\text{CV}[h^2] = \sqrt{\frac{2}{p+2} \cdot V_{\text{rel}}(\mathbf{G}^*)},$$

where $V_{\text{rel}}(\mathbf{G}^*) = \text{Var}(\lambda^*)/\bar{\lambda}^{*2}$ is the relative variance of the eigenvalues.

Key Idea

The variability of directional heritability across directions depends on two factors:

1. The relative variance of the eigenvalues of \mathbf{G}^* —how eccentric is the P-standardised genetic ellipsoid?
2. The number of traits p —more traits mean more “averaging” and less variability.

8.7 Constraint traps

A **constraint trap** occurs when a direction has low heritability despite having substantial phenotypic variance. Selection in that direction produces little evolutionary response because the genetic variance is low relative to environmental variance.

In the \mathbf{G}^* framework, constraint traps correspond to directions near the eigenvectors of \mathbf{G}^* with small eigenvalues. These are directions where:

- The phenotypic variance is typical (by construction, we are on the P-sphere).
- The genetic variance is unusually low.
- The heritability $h^2(\beta)$ is near its minimum.

The danger is subtle. A breeder or natural selection might target a direction with plenty of phenotypic variation, expecting a response. But if that direction happens to be a constraint trap, the response will be disappointing—the variation is mostly environmental, not genetic.

8.8 Visualising G inside P

A useful visualisation is to plot the G ellipse and P ellipse together, centred at the same point. In two dimensions:

- The P ellipse shows where phenotypic variation extends.
- The G ellipse shows where genetic variation extends.
- Directions where G is thin relative to P are low-heritability directions.
- Directions where G nearly fills P are high-heritability directions.

After whitening, P becomes the unit circle. The \mathbf{G}^* ellipse sits inside it (assuming $h^2 \leq 1$ in all directions). The shape of \mathbf{G}^* relative to the circle reveals the constraint structure.

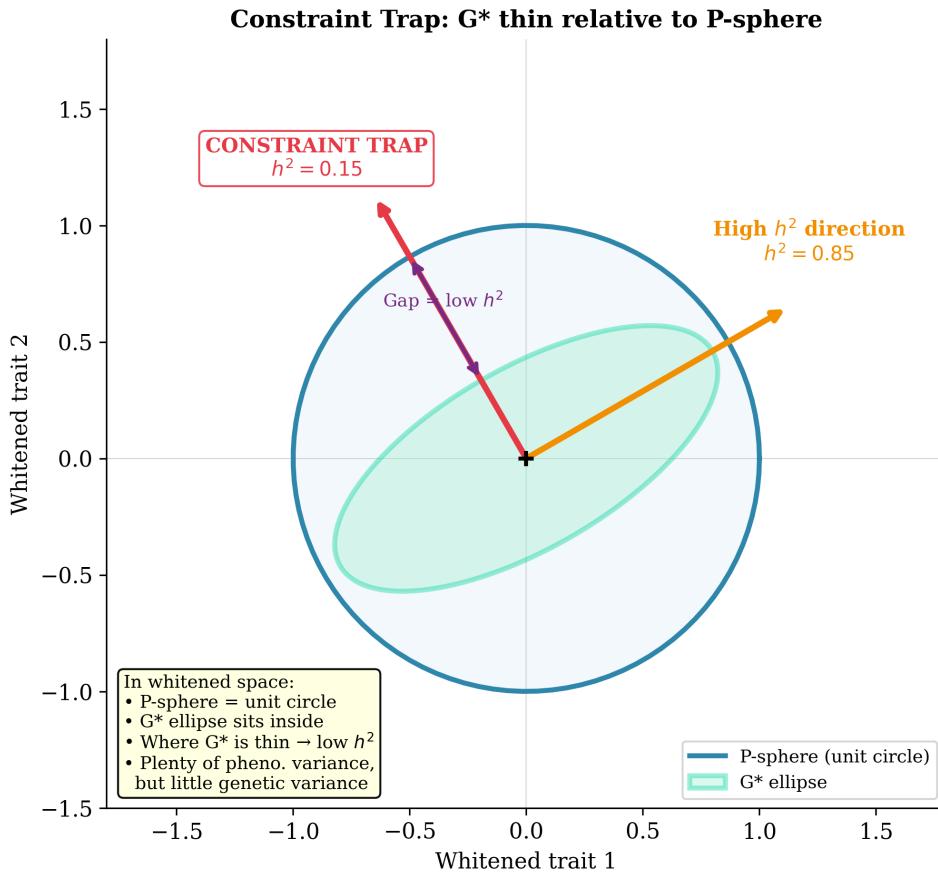


Figure 8.3: In whitened space, the G^* ellipse sits inside the P-sphere (which is now the unit circle). Directions where G^* is thin relative to the sphere are constraint traps: plenty of phenotypic variance, but little genetic variance.

8.9 Connection to the breeder's equation equation

Recall the multivariate breeder's equation equation:

$$\Delta \bar{z} = \mathbf{G}\mathbf{P}^{-1}\mathbf{S} = \mathbf{G}\beta,$$

where \mathbf{S} is the selection differential and $\beta = \mathbf{P}^{-1}\mathbf{S}$ is the selection gradient.

The response to selection depends on both \mathbf{G} and the direction of β . If β points in a high-heritability direction (large $\beta^\top \mathbf{G}\beta$ relative to $\beta^\top \mathbf{P}\beta$), the response is strong. If it points in a constraint trap, the response is weak.

The P-whitening framework makes this explicit. In whitened coordinates:

$$\Delta \bar{z}^* = \mathbf{G}^* \beta^*.$$

The response in whitened space is simply \mathbf{G}^* acting on the whitened selection gradient. The eigenstructure of \mathbf{G}^* directly determines how selection translates to response.

8.10 Computing \mathbf{G}^* in practice

Given estimates of \mathbf{G} and \mathbf{P} , here is how to compute \mathbf{G}^* :

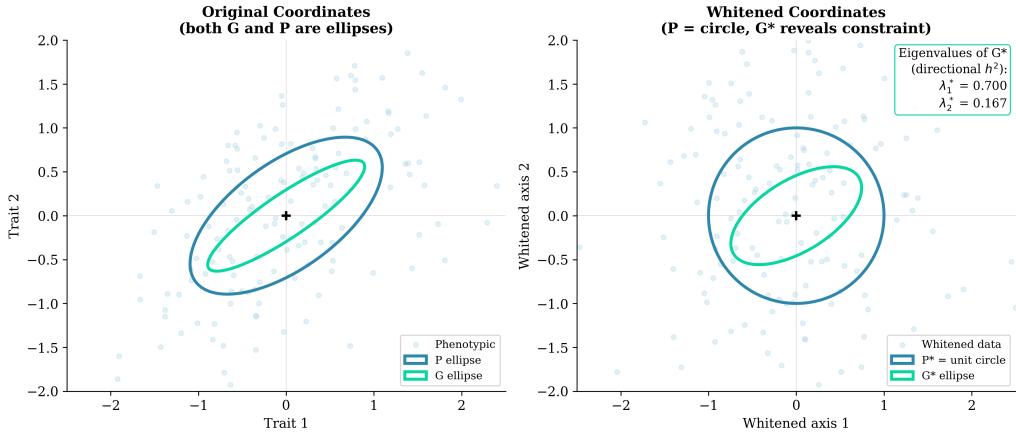


Figure 8.4: Left: In original coordinates, both G and P are ellipses. Right: After whitening, P becomes the unit circle, and G^* reveals where heritability is high (G^* close to the circle) or low (G^* far inside).

Step 1: Eigendecompose P .

$$P = V_P \Lambda_P V_P^\top.$$

Step 2: Compute $P^{-1/2}$.

$$P^{-1/2} = V_P \Lambda_P^{-1/2} V_P^\top,$$

where $\Lambda_P^{-1/2}$ has diagonal entries $1/\sqrt{\lambda_i}$.

Step 3: Transform G .

$$G^* = P^{-1/2} G P^{-1/2}.$$

Step 4: Eigendecompose G^* . The eigenvalues of G^* are the directional heritabilities along the principal axes. The eigenvectors (transformed back to original coordinates by $P^{1/2}$) are those principal axes.

In R:

```
# Eigendecompose P
eig_P <- eigen(P)
V_P <- eig_P$vectors
Lambda_P <- diag(eig_P$values)

# Compute P^{-1/2}
P_inv_sqrt <- V_P %*% diag(1/sqrt(eig_P$values)) %*% t(V_P)

# Transform G
G_star <- P_inv_sqrt %*% G %*% P_inv_sqrt

# Eigendecompose G*
eig_Gstar <- eigen(G_star)
# eig_Gstar$values are directional heritabilities
```

8.11 Why whitening matters

The whitening transformation is not just a mathematical convenience. It changes how we think about constraint.

Without whitening, we might compare directions using Euclidean distance angles and conclude that a direction is “close to \mathbf{g}_{\max} ” when it is actually far from it in phenotypic terms. Whitening ensures that our notion of “close” respects the phenotypic covariance structure.

It also simplifies sampling. To study the distribution of heritability across directions, we can sample uniformly from the ordinary sphere in whitened space, which is easy. Sampling uniformly from the P-sphere in original coordinates would require accounting for the elliptical geometry.

Key Idea

Whitening by \mathbf{P} is the multivariate generalisation of dividing by the phenotypic standard deviation. It puts all directions on an equal footing, so that comparisons are fair.

8.12 Summary

In this chapter we have:

- Introduced directional heritability $h^2(\beta)$ as the ratio of genetic to phenotypic variance in a given direction.
- Defined the P-sphere as the set of directions with unit phenotypic variance, and explained why sampling uniformly from the P-sphere is the right notion of “uniform.”
- Developed the whitening transformation using $\mathbf{P}^{-1/2}$, which converts the P-sphere to the ordinary unit sphere.
- Shown that the eigenvalues of $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{G}\mathbf{P}^{-1/2}$ are the extreme directional heritabilities, and the eigenvectors are the directions that achieve them.
- Connected the variance of directional heritability to the relative variance of the eigenvalues of \mathbf{G}^* .
- Defined constraint traps as directions where phenotypic variance is normal but genetic variance (and hence heritability) is low.
- Provided practical code for computing \mathbf{G}^* from estimates of \mathbf{G} and \mathbf{P} .

The whitening framework unifies the geometry of \mathbf{G} and \mathbf{P} into a single picture. In Part IV, we will apply these ideas to real biological questions: the \mathbf{G} matrix and its eigenstructure, fitness surfaces, and the analysis of selection and response.

Part IV

Evolutionary Applications

Chapter 9

The G Matrix and the Genetic Ellipsoid

We now have a complete geometric toolkit: vectors, matrices, eigenvalues, the Mahalanobis distance distance, and the whitening transformation. In this chapter we apply these tools to the central object of multivariate quantitative genetics: the additive genetic covariance matrix, \mathbf{G} .

The \mathbf{G} matrix is not merely a table of numbers. It is a shape—an ellipsoid in trait space that determines how populations can and cannot evolve. By understanding \mathbf{G} geometrically, we gain insight into evolutionary constraint, the “line of least resistance,” and why some trait combinations respond readily to selection while others do not.

9.1 What the \mathbf{G} matrix represents

The additive genetic covariance matrix \mathbf{G} summarises the heritable variation in a population. Its entries are:

- **Diagonal entries** $G_{ii} = V_{A,i}$: the additive genetic variance of trait i .
- **Off-diagonal entries** $G_{ij} = \text{Cov}_A(z_i, z_j)$: the additive genetic covariance between traits i and j .

These covariances arise from pleiotropy (single genes affecting multiple traits) and linkage disequilibrium (non-random association of alleles at different loci). When $G_{ij} > 0$, alleles that increase trait i tend also to increase trait j . When $G_{ij} < 0$, they tend to have opposite effects.

Key Idea

The \mathbf{G} matrix encodes the genetic architecture underlying multiple traits. It determines which trait combinations can be easily assembled by selection and which cannot.

9.2 The genetic ellipsoid

Like any symmetric positive semi-definite matrix, \mathbf{G} defines an ellipsoid. In two traits, this is an ellipse; in three traits, an ellipsoid; in p traits, a p -dimensional hyperellipsoid.

The ellipsoid has a concrete interpretation: it shows where genetic variation extends in trait space. Directions along the long axes of the ellipsoid have high genetic variance; directions along the short axes have low genetic variance.

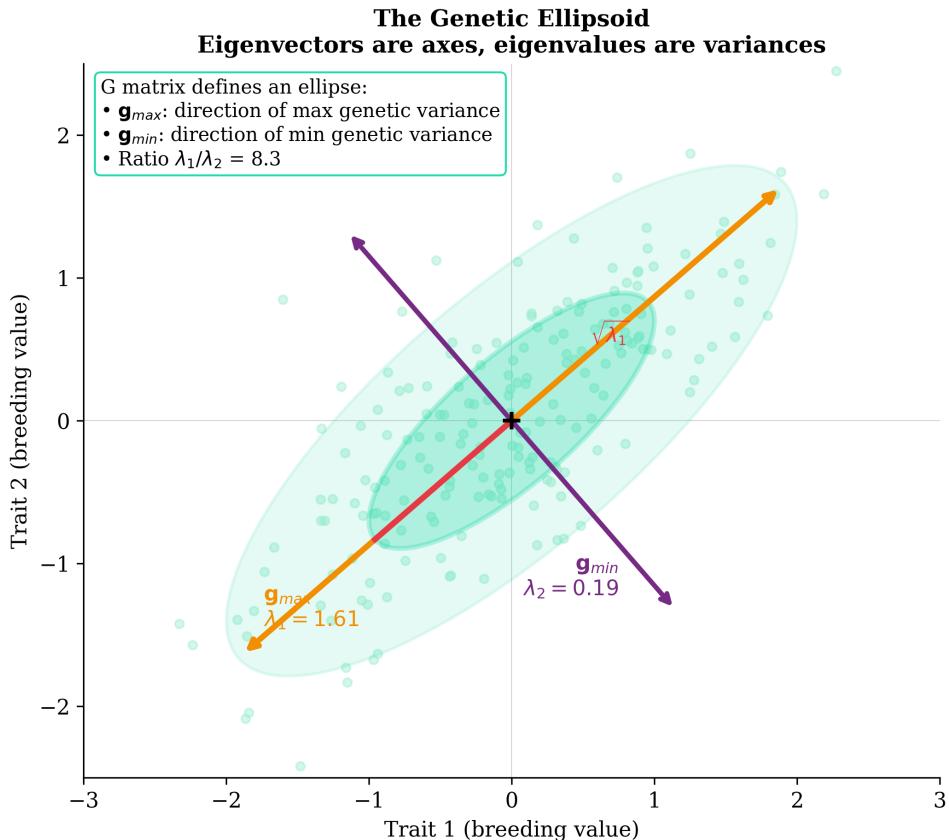


Figure 9.1: The G matrix defines an ellipsoid in trait space. The eigenvectors point along the principal axes; the eigenvalues are the genetic variances along those axes. The shape reveals which directions have abundant genetic variation and which are constrained.

Formally, if we diagonalise $\mathbf{G} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$:

- The eigenvectors $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_p$ are the principal axes of the genetic ellipsoid.
- The eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$ are the genetic variances along those axes.
- The semi-axis lengths are $\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots, \sqrt{\lambda_p}$.

9.3 \mathbf{g}_{\max} : the line of least evolutionary resistance

The first eigenvector of \mathbf{G} , denoted \mathbf{g}_{\max} , points in the direction of maximum genetic variance. This direction has been called the “line of least evolutionary resistance” because:

1. Selection along \mathbf{g}_{\max} produces the largest possible response per unit of selection intensity.
2. Even when selection targets a different direction, the response tends to be deflected toward \mathbf{g}_{\max} .

3. Over evolutionary time, populations may diverge primarily along \mathbf{g}_{\max} , not along the direction of selection.

Key Idea

\mathbf{g}_{\max} is the direction of maximum genetic variance. It represents the path of least resistance for evolutionary change—the direction the population “wants” to go, regardless of where selection points.

The smallest eigenvector, \mathbf{g}_{\min} , represents the direction of minimum genetic variance. Evolution in this direction is difficult: even strong selection produces little response because the necessary genetic variation is scarce.

9.4 The multivariate breeder's equation revisited

The importance of G geometry becomes clear through the breeder's equation equation. In its multivariate form:

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\beta,$$

where $\beta = \mathbf{P}^{-1}\mathbf{S}$ is the selection gradient and \mathbf{S} is the selection differential.

The response $\Delta \bar{\mathbf{z}}$ is not parallel to β unless \mathbf{G} is a scalar multiple of the identity (equal variances, no covariances). In general, \mathbf{G} rotates and stretches the selection gradient, deflecting the response toward directions of high genetic variance.

A worked example

Consider a G matrix with strong positive genetic correlation:

$$\mathbf{G} = \begin{pmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{pmatrix}.$$

The eigenvalues are $\lambda_1 = 1.8$ and $\lambda_2 = 0.2$. The first eigenvector points at 45° (both traits increasing together); the second points at 135° (traits in opposition).

Suppose selection favours increased trait 2 only: $\beta = (0, 1)^\top$.

The response is:

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\beta = \begin{pmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.8 \\ 1.0 \end{pmatrix}.$$

Selection targeted only trait 2, but the response includes an increase in trait 1 as well. The genetic correlation has “dragged” trait 1 along. The response vector $(0.8, 1.0)$ is deflected toward the 45° direction of \mathbf{g}_{\max} .

9.5 Evolvability: genetic variance in the direction of selection

Hansen and Houle introduced **evolvability** as the genetic variance in the direction of selection, scaled appropriately. For a selection gradient β , the evolvability is:

$$e(\beta) = \beta^\top \mathbf{G}\beta.$$

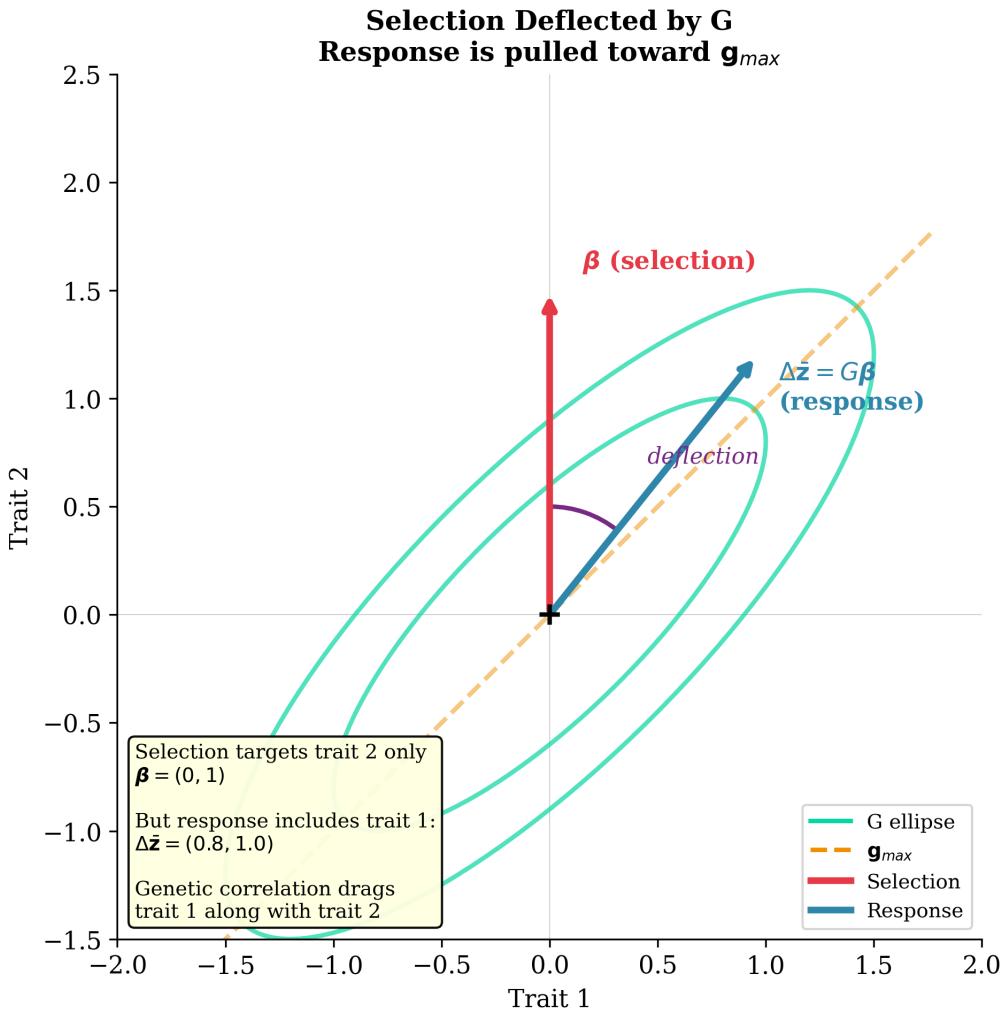


Figure 9.2: Selection (red arrow) aims in one direction, but the response (blue arrow) is deflected toward \mathbf{g}_{\max} . The more eccentric the G ellipse, the stronger the deflection.

When β is a unit vector, this is simply the genetic variance in that direction. When β is standardised by trait means (for mean-scaled evolvability), it measures the proportional response to proportional selection.

Key Idea

evolvability $e(\beta) = \beta^\top \mathbf{G} \beta$ measures the genetic variance available in the direction selection is pushing. High evolvability means large response; low evolvability means the population is constrained in that direction.

From Chapter 20, we know that:

$$\lambda_{\min} \leq \beta^\top \mathbf{G} \beta \leq \lambda_{\max}.$$

evolvability ranges from the smallest to the largest eigenvalue of \mathbf{G} . The ratio $\lambda_{\max}/\lambda_{\min}$ quantifies how much evolvability varies across directions—how “eccentric” the genetic ellipsoid is.

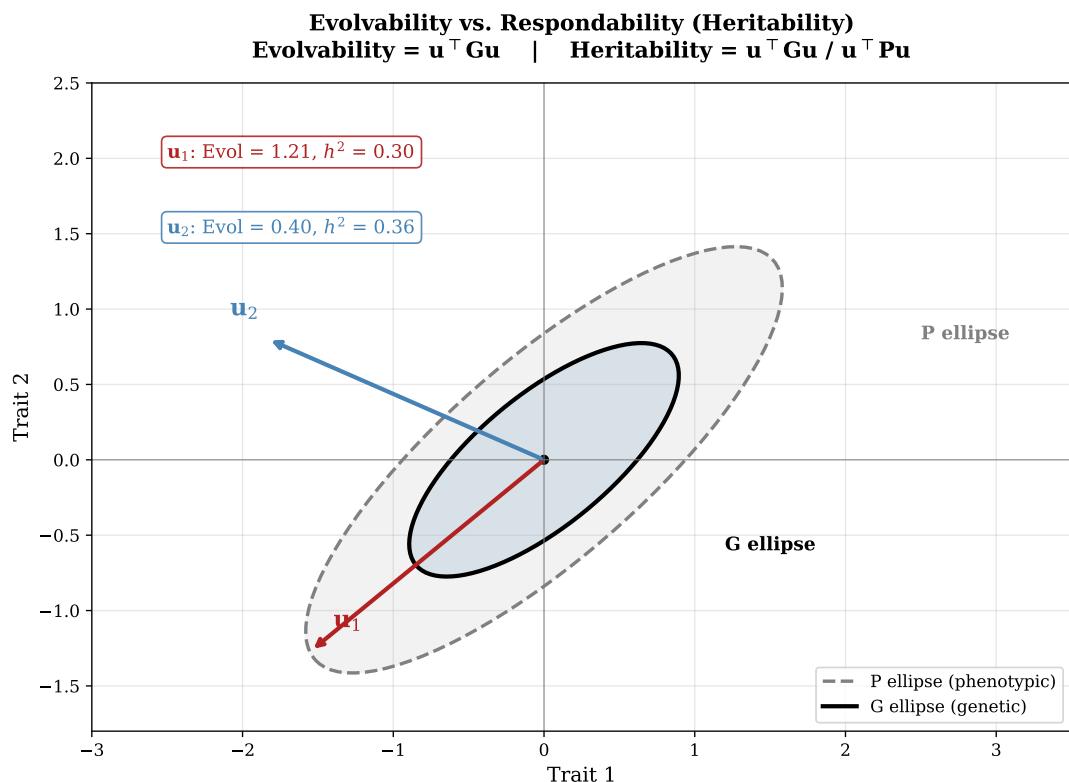


Figure 9.3: **Evolvability versus respondability.** Direction \mathbf{u}_1 (red) has high evolvability (genetic variance = 1.21) but lower heritability ($h^2 = 0.30$) because phenotypic variance is even higher. Direction \mathbf{u}_2 (blue) has lower evolvability (genetic variance = 0.40) but higher heritability ($h^2 = 0.37$). Evolvability tells you how much genetic variance is available; respondability (heritability) tells you what fraction of phenotypic variance is genetic. Both matter for predicting response to selection.

9.6 Respondability and the comparison with P

evolvability measures genetic variance alone. But response to selection also depends on how much of the phenotypic variance is genetic. This leads to **respondability**:

$$r(\beta) = \frac{\beta^\top G\beta}{\beta^\top P\beta} = h^2(\beta),$$

which is the directional heritability from Chapter 21.

Respondability asks: of the phenotypic variance in direction β , what fraction is genetic? A direction can have high evolvability (lots of genetic variance) but low respondability (even more environmental variance), or vice versa.

The P-whitening framework from Chapter 21 lets us study respondability systematically. The eigenvalues of $G^* = P^{-1/2}GP^{-1/2}$ are the respondabilities along the principal axes.

9.7 Constraint: when G limits evolution

The G matrix constrains evolution when:

1. **Low eigenvalues exist.** Some directions have little genetic variance. Evolution in those directions is slow or impossible.
2. **eigenvalues are unequal.** The genetic ellipsoid is eccentric. Response is channelled along \mathbf{g}_{\max} even when selection points elsewhere.
3. **G and selection are misaligned.** If selection targets a direction near \mathbf{g}_{\min} , response will be weak.

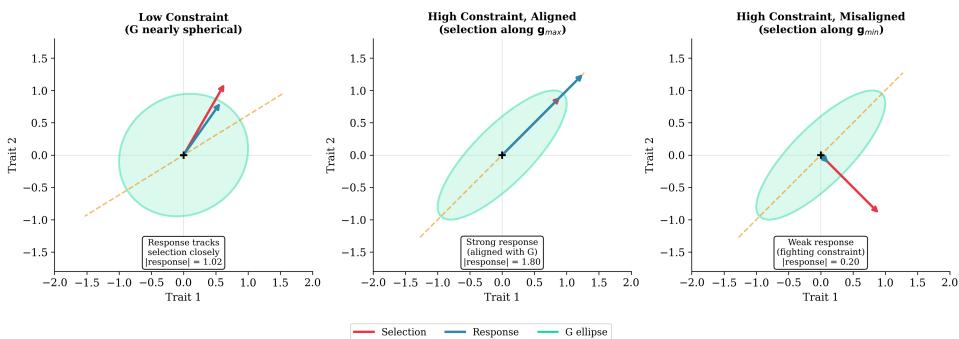


Figure 9.4: Three scenarios. Left: G nearly spherical—little constraint, response tracks selection. Centre: G eccentric but aligned with selection—response is strong. Right: G eccentric and misaligned—response is weak and deflected.

A useful summary statistic is the **eccentricity** of G, measured by the relative variance of its eigenvalues:

$$V_{\text{rel}}(G) = \frac{\text{Var}(\lambda)}{\bar{\lambda}^2} = \frac{\sum_i (\lambda_i - \bar{\lambda})^2 / p}{(\sum_i \lambda_i / p)^2}.$$

When $V_{\text{rel}} = 0$, all eigenvalues are equal and G is spherical (no constraint). When V_{rel} is large, G is highly eccentric (strong constraint).

9.8 Effective dimensionality

Another way to quantify constraint is through **effective dimensionality**—how many independent directions of genetic variation exist.

If all eigenvalues were equal, $\lambda_i = \bar{\lambda}$, we would have p effective dimensions. If one eigenvalue dominates, the effective dimensionality is closer to 1.

One common measure is:

$$n_{\text{eff}} = \frac{(\text{tr}\mathbf{G})^2}{\text{tr}(\mathbf{G}^2)} = \frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2}.$$

This equals p when all eigenvalues are equal and approaches 1 when one eigenvalue dominates.

Key Idea

Effective dimensionality measures how many “independent” directions of genetic variation the population has. Low effective dimensionality means genetic variation is concentrated in a few directions—the population is genetically constrained.

9.9 Empirical G matrices: what do they look like?

Decades of empirical work have revealed some patterns:

- **G matrices are often eccentric.** In many studies, the first few eigenvalues account for most of the genetic variance. Effective dimensionality is typically much less than p .
- **\mathbf{g}_{\max} often aligns with body size.** For morphological traits, the direction of maximum genetic variance frequently corresponds to overall size—all traits scaling together.
- **Genetic correlations can be strong.** Off-diagonal elements of G are often substantial, reflecting pervasive pleiotropy.
- **G varies among populations.** The G matrix is not fixed; it evolves and can differ between populations, species, and environments.

These patterns suggest that genetic constraint is common. Populations do not have equal access to all directions in trait space; evolution is channelled along particular paths.

9.10 Stability and estimation of G

Estimating G requires breeding designs (parent-offspring regression, half-sib designs, animal models) or genomic data. Estimation is challenging because:

- **Sample sizes are often small.** Estimating a $p \times p$ matrix requires estimating $p(p + 1)/2$ unique elements. With many traits, sampling error can be severe.

- **G can be singular or nearly singular.** If some trait combinations have near-zero genetic variance, the estimated G may have zero or negative eigenvalues due to sampling error.
- **G may not be stable.** If genetic architecture changes over time or differs among environments, a single G matrix may not capture the population's evolutionary potential.

These issues motivate careful statistical treatment: regularisation, Bayesian estimation, and sensitivity analyses. We should interpret G matrices with appropriate caution, especially their smaller eigenvalues.

9.11 G in the context of P

Throughout these notes we have emphasised comparing G and P. The P matrix describes total phenotypic variation; G describes the heritable component. Their relationship determines:

- **heritability:** The ratio of G to P, generalised to multiple traits through directional heritability $h^2(\beta)$.
- **Selection response:** The breeder's equation equation $\Delta\bar{z} = \mathbf{G}\mathbf{P}^{-1}\mathbf{S}$ involves both.
- **Constraint traps:** Directions where G is small relative to P—phenotypic variation exists, but it is mostly environmental.

The P-whitening transformation $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{G}\mathbf{P}^{-1/2}$ places G and P on a common footing. In whitened space, P is the identity, and \mathbf{G}^* directly reveals the heritability structure.

9.12 Summary

In this chapter we have:

- Interpreted the G matrix as an ellipsoid in trait space, with eigenvectors as principal axes and eigenvalues as genetic variances along those axes.
- Identified \mathbf{g}_{\max} as the direction of maximum genetic variance—the line of least evolutionary resistance.
- Shown how the breeder's equation equation $\Delta\bar{z} = \mathbf{G}\beta$ deflects selection response toward \mathbf{g}_{\max} .
- Defined evolvability as $\beta^\top \mathbf{G}\beta$ and respondability (directional heritability) as its ratio to phenotypic variance.
- Discussed constraint in terms of eigenvalue eccentricity and effective dimensionality.
- Summarised empirical patterns: G matrices are often eccentric, with \mathbf{g}_{\max} frequently aligned with body size.
- Noted challenges in estimating G and the importance of comparing G to P through P-whitening.

The G matrix is the engine of evolutionary response. Its shape determines which paths are open and which are blocked. In the next chapter, we turn to the other side of the equation: the fitness surface, encoded in the γ matrix, which determines where selection is pushing.

Chapter 10

The Fitness Surface and γ

The previous chapter explored the G matrix—the genetic variation that fuels evolutionary response. Now we turn to the other side of the equation: the fitness surface that determines where selection pushes the population.

The fitness surface is a landscape in trait space, with fitness as elevation. Peaks represent optimal phenotypes; valleys represent maladaptive ones. The shape of this surface—its slopes and curvatures—determines the direction and strength of natural selection. The matrix γ captures the curvature, revealing whether selection is stabilising, disruptive, or acting differently along different trait combinations.

10.1 Fitness as a surface over trait space

Imagine a population of organisms, each with a phenotype that can be represented as a point in trait space. Each phenotype has an associated fitness—the expected reproductive success of an individual with that phenotype.

We can think of fitness as a surface over trait space. In one trait, this is a curve: fitness w as a function of phenotype z . In two traits, it is a surface: $w(z_1, z_2)$. In p traits, it is a hypersurface that we cannot visualise directly but can analyse mathematically.

The key insight is that the local shape of the fitness surface determines selection. Near a peak, fitness decreases in all directions—this is stabilising selection. Near a saddle point, fitness increases in some directions and decreases in others—this is correlational selection. On a slope, fitness increases in one direction—this is directional selection.

10.2 The selection gradient β

The first-order description of the fitness surface is its slope. At any point in trait space, we can ask: in which direction does fitness increase most steeply?

The answer is the **selection gradient** β , defined as the vector of partial derivatives of relative fitness with respect to each trait:

$$\beta_i = \frac{\partial \ln w}{\partial z_i} = \frac{1}{w} \frac{\partial w}{\partial z_i}.$$

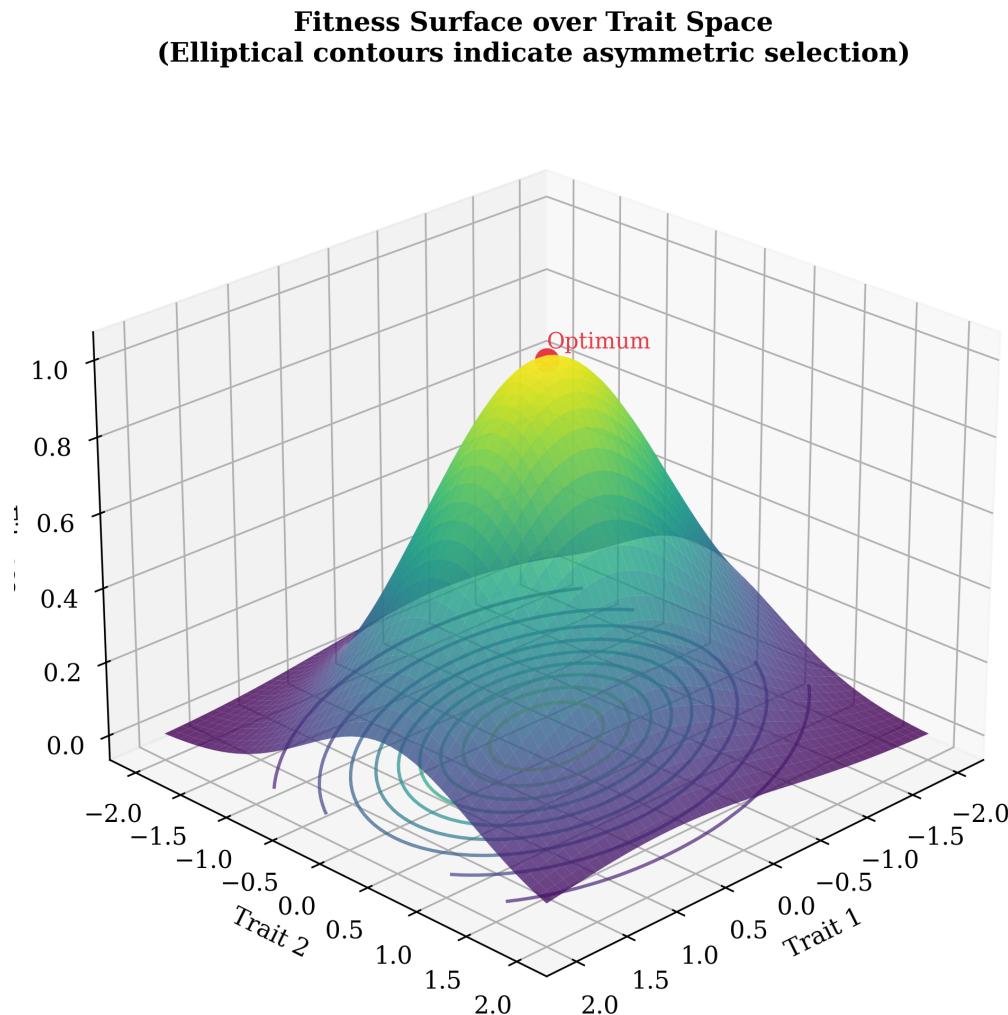


Figure 10.1: A fitness surface over two traits. The peak represents the optimum phenotype. Contour lines show phenotypes of equal fitness. The surface may be symmetric (circular contours) or asymmetric (elliptical contours), depending on how selection acts on different trait combinations.

In matrix notation:

$$\beta = \nabla \ln w.$$

The selection gradient points “uphill” on the fitness surface. Its magnitude indicates how steep the slope is; its direction indicates where selection is pushing.

Key Idea

The selection gradient β is the direction of steepest ascent on the fitness surface. It describes directional selection—the tendency for the population mean to move toward higher fitness.

From the Lande equation, the response to selection is $\Delta\bar{z} = G\beta$. The selection gradient tells us where selection wants to go; the G matrix determines how much of that desire is realised.

10.3 Beyond slopes: the curvature matrix γ

Slopes tell us about directional selection, but they miss an important aspect: is the fitness surface curved? Is selection pushing the population toward a peak (stabilising) or away from a valley (disruptive)?

Curvature is captured by second derivatives. The matrix of second partial derivatives of relative fitness is:

$$\gamma_{ij} = \frac{\partial^2 \ln w}{\partial z_i \partial z_j}.$$

This is the **quadratic selection gradient** or **gamma matrix**, denoted γ .

Key Idea

The γ matrix describes the curvature of the fitness surface. Its eigenvalues tell us whether selection is stabilising (negative curvature) or disruptive (positive curvature) along each principal axis.

Like G and P , γ is a symmetric matrix. It can be diagonalised, and its eigenstructure reveals the geometry of selection.

10.4 eigenvalue

s]Interpreting γ : the sign of eigenvalues

Suppose we diagonalise γ :

$$\gamma = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top.$$

The eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_p$ describe the curvature along the principal axes (eigenvectors) of the fitness surface.

Negative eigenvalue: stabilising selection. If $\lambda_i < 0$, the fitness surface curves downward along eigenvector \mathbf{v}_i . Moving away from the current mean in that direction decreases fitness. Selection is stabilising: it pushes the population back toward the centre.

Positive eigenvalue: disruptive selection. If $\lambda_i > 0$, the fitness surface curves upward along \mathbf{v}_i . Moving away from the mean increases fitness. Selection is disruptive: it pushes the population away from the centre, potentially toward two or more distinct phenotypes.

Zero eigenvalue: no curvature. If $\lambda_i = 0$, the fitness surface is flat along \mathbf{v}_i (at least locally). Selection has no stabilising or disruptive component in that direction—only directional selection (if any) from β .

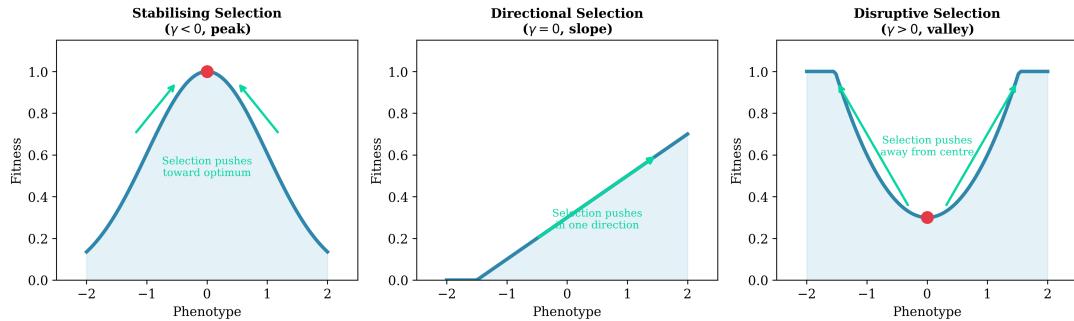


Figure 10.2: Three types of curvature in one dimension. Left: negative curvature (stabilising selection, fitness peak). Centre: zero curvature (directional selection, constant slope). Right: positive curvature (disruptive selection, fitness valley).

10.5 Correlational selection: the off-diagonal elements

The diagonal elements of γ describe curvature along each individual trait. The off-diagonal elements describe something more subtle: **correlational selection**.

If $\gamma_{12} \neq 0$, then the curvature of the fitness surface depends on trait combinations, not just individual traits. A positive γ_{12} means that fitness is higher when both traits are simultaneously high or simultaneously low (positive correlationfavoured). A negative γ_{12} means that fitness is higher when traits are in opposition (negative correlationfavoured).

Key Idea

Correlational selection ($\gamma_{ij} \neq 0$ for $i \neq j$) favours particular trait combinations. It can build or break genetic correlations over evolutionary time.

The eigenvectors of γ reveal the trait combinations along which selection acts most strongly (largest $|\lambda_i|$) and the directions along which selection is neutral (small $|\lambda_i|$).

10.6 Quadratic fitness functions

A common model for the fitness surface is a quadratic function centred on an optimum θ :

$$w(\mathbf{z}) = w_{\max} \exp \left(-\frac{1}{2} (\mathbf{z} - \boldsymbol{\theta})^\top \boldsymbol{\omega}^{-1} (\mathbf{z} - \boldsymbol{\theta}) \right).$$

Here $\boldsymbol{\omega}$ is a matrix that describes the “width” of the fitness peak. The quadratic form in the exponent is a Mahalanobis distance distance from the optimum, using $\boldsymbol{\omega}^{-1}$ as the metric.

For this Gaussian fitness function:

- The selection gradient at phenotype \mathbf{z} is $\boldsymbol{\beta} = \boldsymbol{\omega}^{-1}(\boldsymbol{\theta} - \mathbf{z})$.
- The curvature matrix is $\gamma = -\boldsymbol{\omega}^{-1}$.

The curvature is constant everywhere and equals the negative of the inverse width matrix. Narrow peaks (small $\boldsymbol{\omega}$) have strong curvature (large $|\gamma|$); wide peaks have weak curvature.

A Fitness Saddle from Correlational Selection (γ with Mixed Eigenvalues)
Stabilizing selection ($\lambda < 0$) along one axis, disruptive ($\lambda > 0$) along the perpendicular.

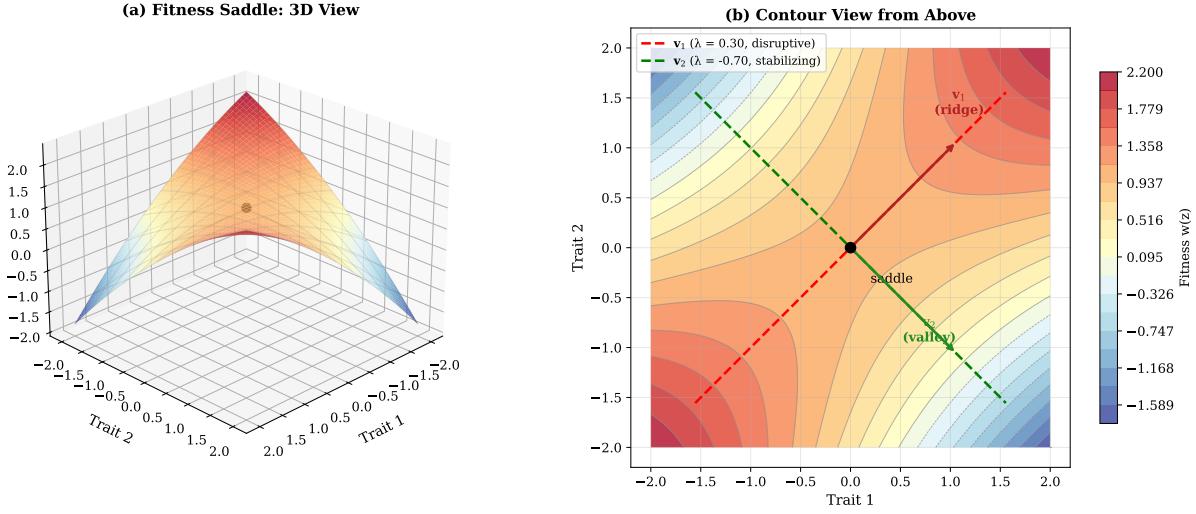


Figure 10.3: **A fitness saddle from correlational selection.** When γ has both positive and negative eigenvalues, the fitness surface is a saddle. (a) Three-dimensional view: fitness increases along one axis (the ridge, $\lambda = 0.3 > 0$, disruptive) and decreases along the other (the valley, $\lambda = -0.7 < 0$, stabilising). (b) Contour view from above: hyperbolic level curves indicate a saddle point. The eigenvectors of γ (dashed lines) point along the ridge and valley directions. Correlational selection ($\gamma_{12} \neq 0$) rotates these directions away from the original trait axes.

10.7 Estimating γ from data

In practice, γ is estimated by regressing fitness on traits and their products. The standard approach (Lande and Arnold 1983) uses:

$$w = \alpha + \sum_i \beta_i z_i + \frac{1}{2} \sum_i \gamma_{ii} z_i^2 + \sum_{i < j} \gamma_{ij} z_i z_j + \epsilon.$$

The linear coefficients estimate directional selection (β); the quadratic and cross-product coefficients estimate the curvature (γ).

There are subtleties:

- **Standardisation.** Traits are typically standardised to have mean zero and unit variance before analysis, so that coefficients are comparable across traits.
- **The factor of 2.** Note the $\frac{1}{2}$ in front of the diagonal quadratic terms. This ensures that γ_{ii} equals the second derivative $\partial^2 w / \partial z_i^2$, not half of it.
- **Relative fitness.** Fitness should be relativised (divided by mean fitness) so that the regression estimates selection gradients, not absolute fitness effects.
- **Sample size.** Estimating γ requires large samples because quadratic terms have less power than linear terms.

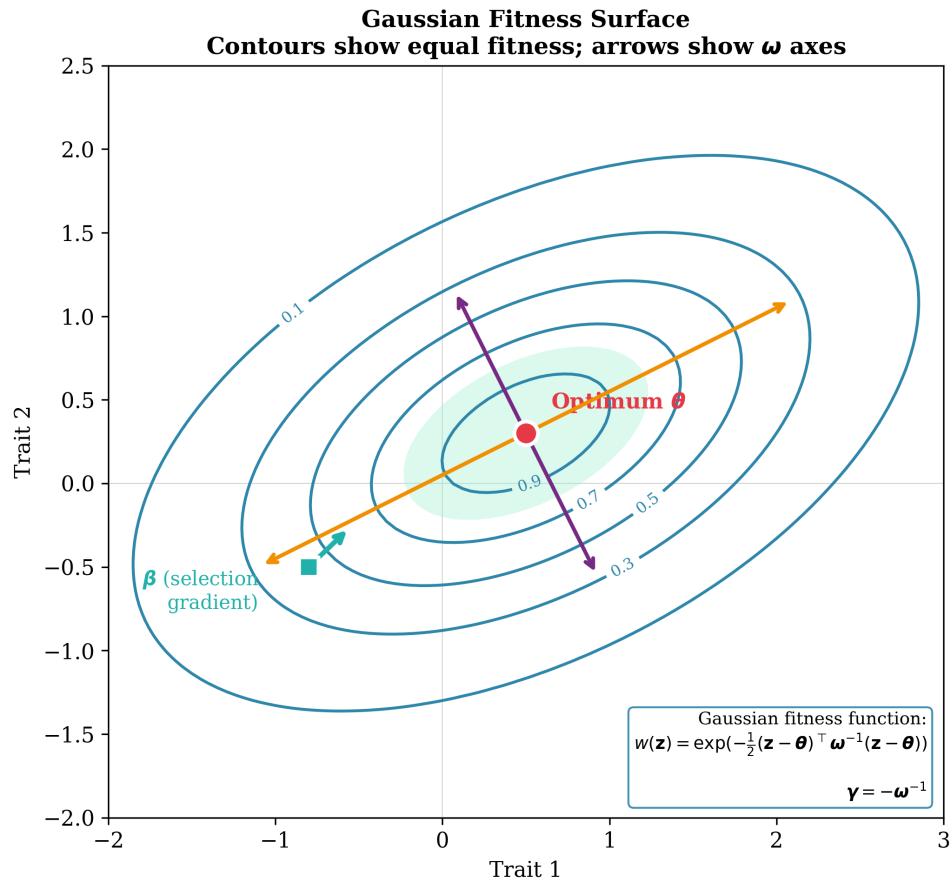


Figure 10.4: A Gaussian fitness surface with elliptical contours. The optimum is at θ . The width matrix ω determines how rapidly fitness declines away from the optimum in different directions.

10.8 The geometry of γ

Like G and P , we can visualise γ as an ellipse (in two traits) or ellipsoid (in higher dimensions). But the interpretation is different:

- For G and P , the ellipse shows where *variation* extends.
- For γ , the ellipse shows where *selection curvature* is strongest.

The eigenvectors of γ are the principal axes of the fitness surface—the directions along which curvature is purely stabilising or disruptive, with no correlational component. The eigenvalues are the curvatures along those axes.

If all eigenvalues of γ are negative and equal, the fitness surface is a symmetric peak with circular contours. If eigenvalues differ, the peak is elongated—selection is stronger in some directions than others.

10.9 Comparing γ and G: alignment matters

A central question in evolutionary biology is: how does the geometry of selection (encoded in γ) interact with the geometry of genetic variation (encoded in G)?

Aligned. If the eigenvectors of γ align with those of G, then selection is strongest along directions where genetic variation is abundant or scarce. This can either accelerate evolution (if strong selection meets abundant variation) or frustrate it (if strong selection meets scarce variation).

Misaligned. If γ and G are misaligned, the situation is more complex. Selection may push in one direction, but genetic variation may only be available in another. The response is a compromise, deflected by G away from where γ would drive it.

Key Idea

The interaction between γ (the geometry of selection) and G (the geometry of genetic variation) determines whether evolution is fast or slow, aligned or deflected. Understanding both matrices is essential for predicting evolutionary trajectories.

10.10 γ and the maintenance of genetic variation

The curvature matrix also influences the maintenance of genetic variation. Under stabilising selection (γ with negative eigenvalues), selection removes variation by favouring intermediate phenotypes. The stronger the curvature (more negative eigenvalues), the faster variation is eroded.

This creates a puzzle: if stabilising selection is common, why do populations retain genetic variation? Possible answers include:

- mutation-selection balance.
- Fluctuating selection (the optimum moves over time).
- Correlational constraints (selection on one trait limited by correlated response in others).
- Frequency-dependent or spatially varying selection.

The eigenstructure of γ helps quantify how rapidly selection should erode variation along different axes, informing these debates.

10.11 A worked example

Consider a study measuring survival as a function of two morphological traits in a bird population. After standardising traits and relativising fitness, a quadratic regression yields:

$$\beta = \begin{pmatrix} 0.15 \\ 0.08 \end{pmatrix}, \quad \gamma = \begin{pmatrix} -0.12 & 0.06 \\ 0.06 & -0.08 \end{pmatrix}.$$

Interpreting β . Both selection gradients are positive: directional selection favours larger values of both traits. Trait 1 is under stronger directional selection than trait 2.

Interpreting γ . Both diagonal elements are negative: stabilising selection on each trait individually. The off-diagonal element is positive: correlational selection favours positive trait combinations (both high or both low).

Eigendecomposition.

$$\lambda_1 = -0.04, \quad \lambda_2 = -0.16.$$

Both eigenvalues are negative, confirming overall stabilising selection. But selection is much stronger ($|\lambda_2| = 0.16$) along the second eigenvector than the first ($|\lambda_1| = 0.04$). The fitness peak is elongated.

The eigenvectors reveal that the direction of weakest stabilising selection (λ_1) is approximately the positive diagonal (both traits high together), while the direction of strongest stabilising selection (λ_2) is the negative diagonal (traits in opposition).

This makes biological sense: the population can tolerate variation in overall size (both traits scaling together) but is strongly selected against unusual proportions (one trait high, the other low).

10.12 Canonical analysis of the fitness surface

Phillips and Arnold (1989) introduced **canonical analysis** to understand the geometry of γ . The idea is to:

1. Diagonalise γ to find its eigenvectors and eigenvalues.
2. Express the fitness surface in terms of these principal axes.
3. Interpret which trait combinations are under strong versus weak stabilising or disruptive selection.

In the canonical basis, the fitness function becomes:

$$w = \bar{w} + \sum_i \theta_i m_i + \frac{1}{2} \sum_i \lambda_i m_i^2,$$

where m_i is the projection of the phenotype onto the i th eigenvector, θ_i is the directional selection along that axis (the projection of β), and λ_i is the curvature.

This separates selection into independent components along orthogonal axes, making interpretation cleaner.

10.13 Summary

In this chapter we have:

- Introduced the fitness surface as a landscape over trait space, with slopes (directional selection) and curvatures (stabilising or disruptive selection).
- Defined the selection gradient β as the slope of the fitness surface and the curvature matrix γ as its second derivatives.
- Interpreted eigenvalues of γ : negative means stabilising, positive means disruptive, zero means flat.
- Explained correlational selection through off-diagonal elements of γ .
- Described Gaussian fitness surfaces, where $\gamma = -\omega^{-1}$.
- Outlined how to estimate γ from fitness data using quadratic regression.
- Emphasised that the interaction between γ and G determines evolutionary trajectories.
- Worked through an example showing how to interpret eigenvalues and eigenvectors of γ .
- Introduced canonical analysis as a tool for decomposing selection along principal axes.

With G describing what evolution *can* do and γ describing what selection *wants*, we now have both pieces of the puzzle. In the next chapter, we turn to the statistical tools—PCA, MANOVA, and related methods—that let us estimate and compare these matrices from data.

Chapter 11

PCA, MANOVA, and Projections

The previous chapters developed the geometry of G , P , and γ . We saw that these matrices are ellipsoids, that their eigenstructure reveals principal axes, and that comparing matrices illuminates evolutionary constraint and selection. But how do we estimate these matrices and test hypotheses about them?

This chapter connects the geometric framework to statistical practice. We cover Principal Component Analysis (PCA), Multivariate Analysis of Variance (MANOVA), and related projection methods. The unifying theme is that all these techniques are applications of eigendecomposition to biological questions.

11.1 The bridge from geometry to statistics

Every statistical method in this chapter does the same thing at its core: it takes a covariance matrix (or a ratio of covariance matrices), finds its eigenvalues and eigenvectors, and interprets them biologically.

- **PCA** eigendecomposes a single covariance matrix to find directions of maximum variance.
- **MANOVA** compares covariance matrices (among-group vs. within-group) to test whether groups differ.
- **Canonical correlationAnalysis (CCA)** finds directions that maximise correlation between two sets of variables.
- **Discriminant Analysis** finds directions that best separate groups.

Once you understand diagonalisation, these methods become variations on a single theme.

Key Idea

PCA, MANOVA, CCA, and discriminant analysis are all eigendecompositions of covariance matrices or their ratios. The geometry we have developed provides the interpretive framework for all of them.

11.2 Principal Component Analysis (PCA)

PCA is the most widely used multivariate technique in biology. Its goal is dimensionality reduction: represent the variation in p traits using fewer than p derived variables, while losing as little information as possible.

The procedure

Given data on p traits measured on n individuals:

1. Compute the sample covariance matrix \mathbf{S} (or correlation matrix \mathbf{R} if traits are on different scales).
2. Eigendecompose: $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^\top$.
3. The eigenvectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p$ are the **principal component loadings**.
4. The eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$ are the variances along each principal component.
5. Project each individual onto the eigenvectors to get **principal component scores**.

The first principal component (PC1) is the direction of maximum variance in the data—exactly the first eigenvector of \mathbf{S} . PC2 is the direction of maximum variance orthogonal to PC1, and so on.

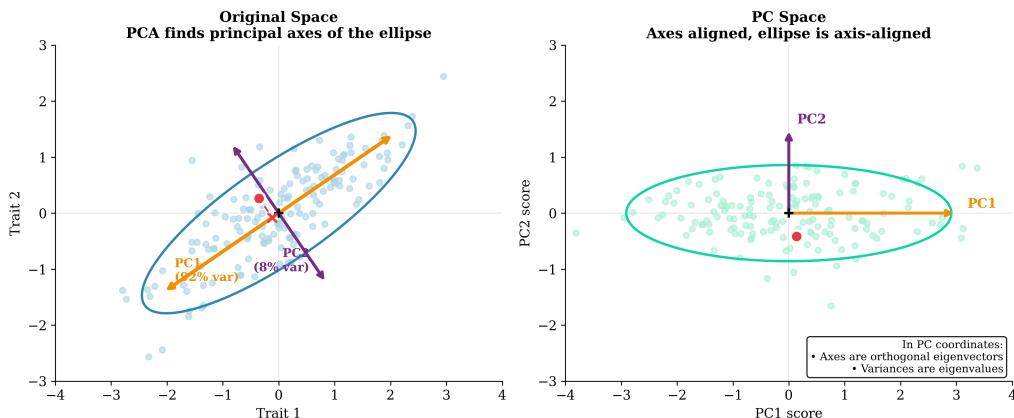


Figure 11.1: PCA

[geometry] PCA finds the principal axes of the data ellipse. PC1 points along the direction of maximum variance; PC2 is orthogonal. Projecting data onto these axes gives the principal component scores.

Interpretation

The eigenvalues tell us how much variance each PC captures. A common summary is the proportion of variance explained:

$$\text{Proportion for PC}_k = \frac{\lambda_k}{\sum_{i=1}^p \lambda_i} = \frac{\lambda_k}{\text{tr}(\mathbf{S})}.$$

If the first few eigenvalues are large and the rest are small, most variation lies in a low-dimensional subspace. We can then work with just PC1 and PC2 (for example) without losing much information.

The eigenvectors (loadings) tell us what each PC represents biologically. If all loadings have the same sign, PC1 represents overall “size.” If loadings have mixed signs, the PC represents a contrast between trait groups.

Key Idea

PCA is eigendecomposition of the covariance matrix. The eigenvalues measure how much variance each direction captures; the eigenvectors define those directions. PCA does not assume any group structure—it describes the total variation in the sample.

Covariance vs. correlationPCA

A crucial choice is whether to analyse the covariance matrix \mathbf{S} or the correlationmatrix \mathbf{R} .

- **Covariance PCA:** Uses raw (centred) data. Traits with larger variances dominate the analysis. Appropriate when traits are on comparable scales and absolute variances are meaningful.
- **correlationPCA:** Standardises each trait to unit variance before analysis. All traits contribute equally regardless of original scale. Appropriate when traits are on different scales (e.g., length in mm vs. mass in g).

In evolutionary biology, covariance PCA is often preferred when analysing G or P matrices because the absolute magnitudes of genetic variances are biologically meaningful. correlationPCA is useful for exploratory analysis of phenotypic data on mixed scales.

11.3 MANOVA: comparing groups

While PCA describes variation within a single sample, MANOVA asks whether multiple groups differ in their multivariate means. It is the multivariate generalisation of ANOVA.

The setup

Suppose we have k groups (e.g., populations, treatments, species) and measure p traits on individuals within each group. MANOVA tests the null hypothesis that all group means are equal:

$$H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \cdots = \boldsymbol{\mu}_k.$$

The geometry

MANOVA decomposes total variation into among-group and within-group components, just as univariate ANOVA does. The key objects are:

- **Among-group matrix \mathbf{B} :** Measures how group means differ from the grand mean. Large \mathbf{B} indicates groups are spread out in trait space.
- **Within-group matrix \mathbf{W} :** Measures variation within groups, pooled across groups. This is the “error” variation.
- **Total matrix $\mathbf{T} = \mathbf{B} + \mathbf{W}$:** Total variation ignoring group structure.

The MANOVA test asks: is \mathbf{B} large relative to \mathbf{W} ? If groups are very different (large \mathbf{B}) and within-group variation is small (small \mathbf{W}), we reject the null hypothesis.

Test statistics

Several test statistics are used, all based on eigenvalues of $\mathbf{W}^{-1}\mathbf{B}$:

- **Wilks' Λ :** $\Lambda = \det(\mathbf{W}) / \det(\mathbf{T}) = \prod_i (1 + \lambda_i)^{-1}$, where λ_i are eigenvalues of $\mathbf{W}^{-1}\mathbf{B}$.
- **Pillai's trace:** $\sum_i \lambda_i / (1 + \lambda_i)$.
- **Hotelling-Lawley trace:** $\sum_i \lambda_i$.
- **Roy's largest root:** λ_1 (the largest eigenvalue).

These statistics have different properties. Wilks' Λ is most common; Pillai's trace is most robust to violations of assumptions; Roy's largest root is most powerful when groups differ along a single dimension.

Key Idea

MANOVA tests whether groups differ by comparing among-group variation (\mathbf{B}) to within-group variation (\mathbf{W}). The eigenvalues of $\mathbf{W}^{-1}\mathbf{B}$ quantify how much groups differ along each discriminant axis.

11.4 discriminant analysis

discriminant analysis is closely related to MANOVA. While MANOVA tests *whether* groups differ, discriminant analysis finds the directions along which they differ most and uses these for classification.

Linear discriminant analysis (LDA)

LDA finds linear combinations of traits that maximise the ratio of among-group to within-group variance. These are the eigenvectors of $\mathbf{W}^{-1}\mathbf{B}$.

The first discriminant function (DF1) is the direction along which groups are most separated relative to within-group spread. DF2 is the next-best direction orthogonal to DF1, and so on.

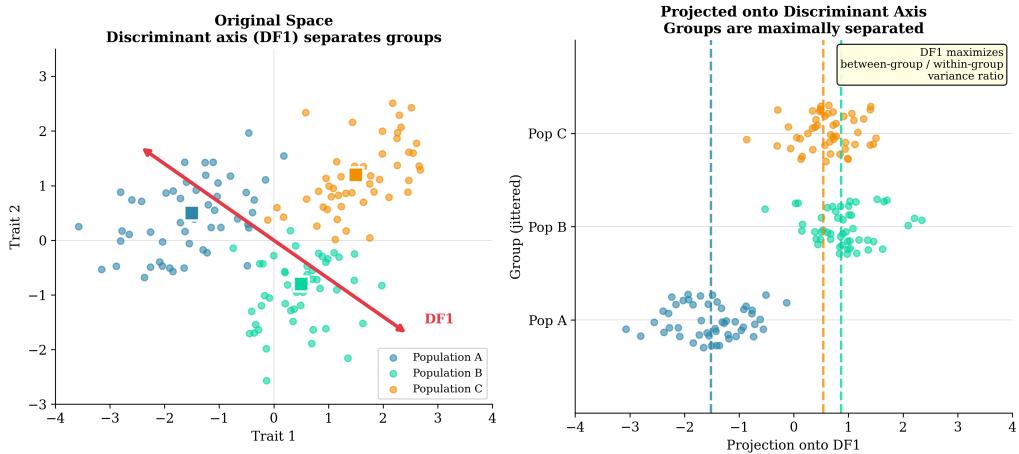


Figure 11.2: discriminant analysis

] discriminant analysis finds directions that separate groups. The first discriminant function maximises the ratio of among-group to within-group variance. Projecting data onto this axis best reveals group differences.

Connection to Mahalanobis distance distance

The Mahalanobis distance distance between two group means, using the pooled within-group covariance, is:

$$D^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \mathbf{W}^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2).$$

This is exactly the squared distance along the discriminant axis connecting the two groups. discriminant analysis and Mahalanobis distance distance are two views of the same geometry.

11.5 Canonical correlationAnalysis (CCA)

CCA extends correlationto multiple variables on each side. Given two sets of variables (e.g., morphological traits and physiological traits), CCA finds linear combinations of each set that are maximally correlated.

The setup

Let \mathbf{x} be a vector of p variables and \mathbf{y} be a vector of q variables. We seek coefficients \mathbf{a} and \mathbf{b} such that the correlationbetween $\mathbf{a}^\top \mathbf{x}$ and $\mathbf{b}^\top \mathbf{y}$ is maximised.

The solution involves eigendecomposition of matrices built from the covariance structure. The eigenvalues are the squared canonical correlations; the eigenvectors define the canonical variates.

Biological applications

CCA is useful for:

- Relating genotype to phenotype (which genetic combinations predict which phenotypic combinations?).
- Relating morphology to performance (which body shapes predict which functional capacities?).
- Relating traits to environmental variables.

Canonical Correlation Analysis: Finding Maximally Correlated Linear Combinations
 CCA finds directions \mathbf{a}_i in X-space and \mathbf{b}_i in Y-space such that their projections are maximally correlated.

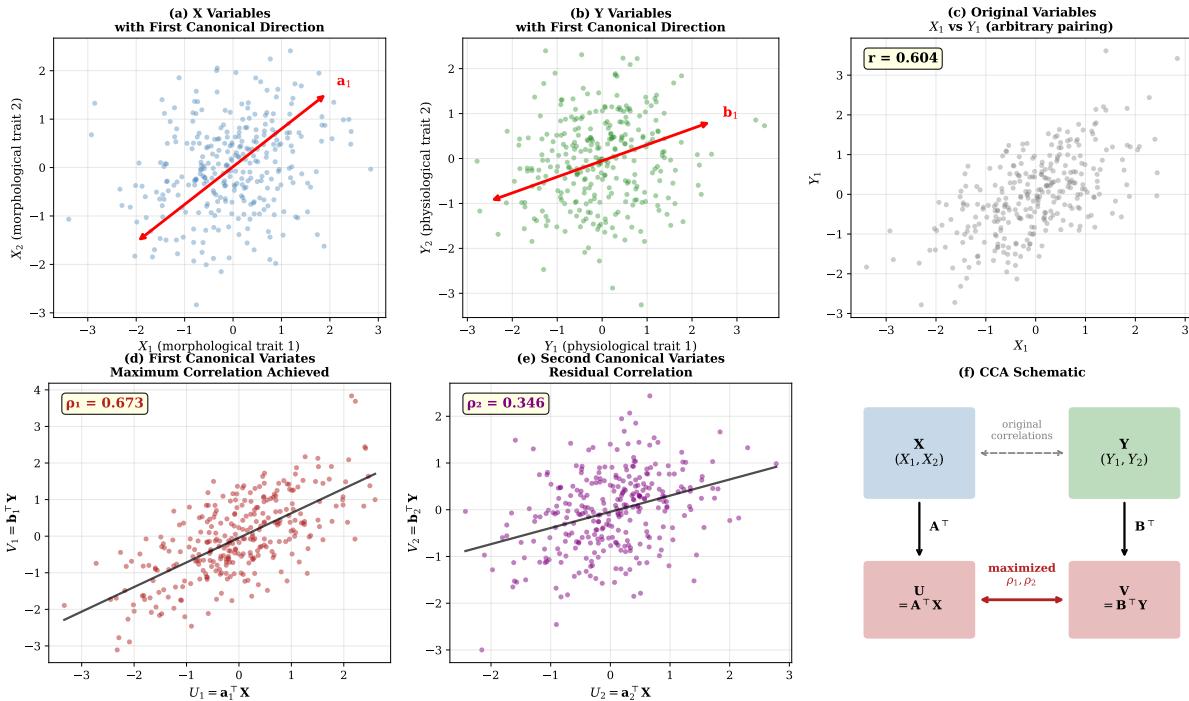


Figure 11.3: Geometry of Canonical Correlation Analysis (CCA). (a) The \mathbf{X} variable set (e.g., morphological traits) with its covariance ellipse and first canonical direction \mathbf{a}_1 . (b) The \mathbf{Y} variable set (e.g., performance traits) with its covariance ellipse and corresponding canonical direction \mathbf{b}_1 . (c) First canonical variate pair: projections onto \mathbf{a}_1 and \mathbf{b}_1 achieve maximum correlation (r_1). (d) Second canonical variate pair: orthogonal to the first, capturing residual correlation (r_2). CCA finds linear combinations of each variable set that are maximally correlated, revealing the underlying dimensions linking morphology to performance, or genotype to phenotype.

11.6 Projection pursuit: the general principle

All these methods share a common structure: they find “interesting” directions in high-dimensional space by optimising some criterion.

- PCA maximises variance.
- discriminant analysis maximises group separation.
- CCA maximises correlation between variable sets.

This perspective, called **projection pursuit**, suggests that we can invent new methods by defining new criteria for “interesting” directions. In evolutionary biology, natural criteria include:

- Directions of maximum genetic variance (\mathbf{g}_{\max}).
- Directions of maximum selection (γ eigenvectors).
- Directions of maximum or minimum heritability (\mathbf{G}^* eigenvectors).

Key Idea

Projection pursuit is the general framework: find directions that optimise some criterion. PCA, discriminant analysis, and CCA are special cases. Evolutionary biologists can define biologically motivated criteria.

11.7 Comparing G matrices

A major application of these methods is comparing G matrices across populations or species. Do different populations have the same pattern of genetic constraints?

Common principal components

Flury’s method tests whether two or more G matrices share the same eigenvectors (common principal components) even if eigenvalues differ. This tests whether the “shape” of genetic constraint is conserved.

Hierarchical models allow partial sharing:

- **Equal matrices:** Same eigenvectors and eigenvalues.
- **Proportional matrices:** Same eigenvectors, eigenvalues differ by a constant factor.
- **Common principal components:** Same eigenvectors, different eigenvalues.
- **Partial CPC:** Some eigenvectors shared, others not.
- **Unrelated:** No structural similarity.

Random skewers

An alternative approach is to compare how matrices respond to random selection vectors. Generate many random unit vectors β , compute the response $\mathbf{G}\beta$ for each matrix, and correlate the responses.

If two G matrices give similar responses to the same selection pressures, they are functionally similar even if their eigenvectors differ.

Krzanowski's subspace comparison

Krzanowski's method compares the subspaces spanned by the first k eigenvectors of two matrices. It asks: do the major axes of variation span similar directions?

The comparison uses angles between subspaces. If the leading eigenvectors of two matrices point in similar directions, the subspaces overlap substantially.

11.8 Estimation issues

Estimating covariance matrices is statistically challenging, especially with many traits.

Sample size requirements

A $p \times p$ covariance matrix has $p(p + 1)/2$ unique elements. With p traits, you need far more than p observations to estimate the matrix reliably. Rules of thumb suggest $n > 10p$ or even $n > 20p$ for stable estimates.

When $n < p$, the sample covariance matrix is singular (has zero eigenvalues). This occurs frequently in genomic studies where thousands of markers are measured on hundreds of individuals.

Shrinkage and regularisation

When sample sizes are small relative to dimensionality, shrinkage estimators improve accuracy by pulling eigenvalues toward a common value (reducing the extremes). Common approaches include:

- **Ledoit-Wolf shrinkage:** Shrinks toward a scaled identity matrix.
- **Ridge regularisation:** Adds a small constant to the diagonal before inversion.
- **Factor models:** Assume the covariance arises from a few latent factors.

These methods trade bias for reduced variance, often improving predictions and avoiding numerical instability.

Bayesian estimation

Bayesian methods place prior distributions on covariance matrices and estimate posterior distributions. This naturally handles small samples and provides uncertainty quantification for eigenvalues and eigenvectors.

Animal models in quantitative genetics typically use Bayesian MCMC to estimate G matrices, providing credible intervals for genetic parameters.

11.9 Visualising high-dimensional patterns

With more than three traits, direct visualisation is impossible. Several approaches help:

- **Scree plots:** Plot eigenvalues against their rank to see how many dimensions capture most variation.
- **Biplots:** Overlay individuals (as points) and variables (as arrows) in the PC1-PC2 plane.
- **Heatmaps:** Display the covariance or correlationmatrix as a coloured grid.
- **Parallel coordinates:** Draw each individual as a line connecting their values on parallel trait axes.

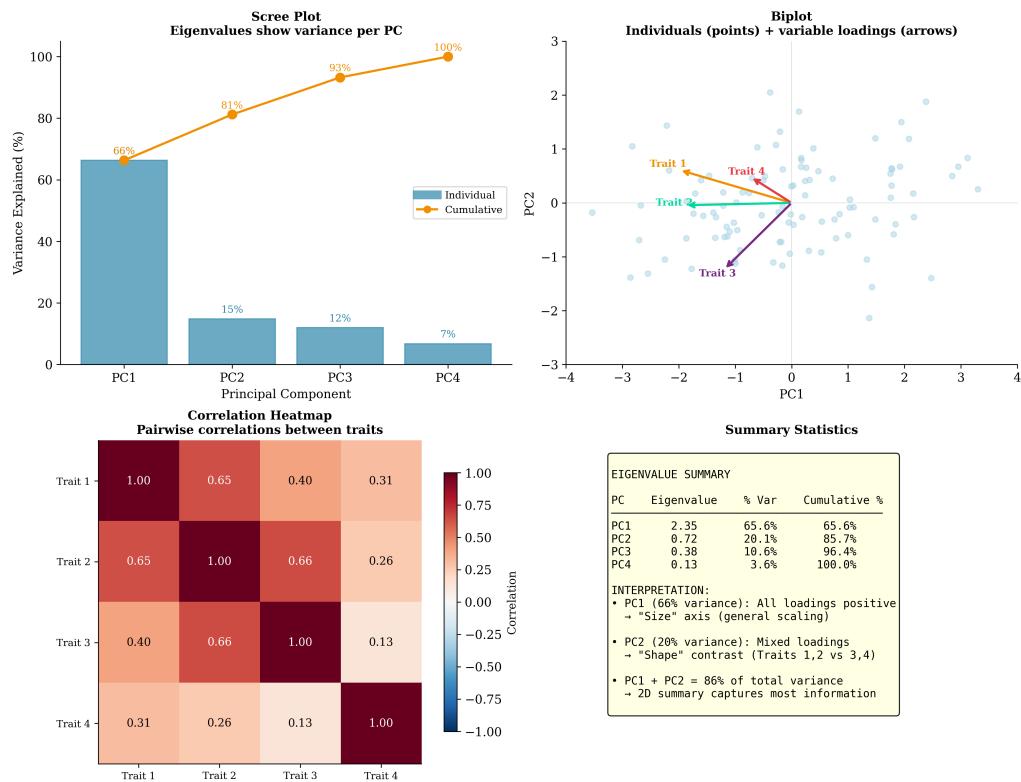


Figure 11.4: Methods for visualising multivariate data. Top left: scree plot of eigenvalues. Top right: biplot showing individuals and variable loadings. Bottom: correlation heatmap.

11.10 Worked example: PCA of a phenotypic dataset

Consider measurements of four traits on 150 individuals from three populations. We perform PCA on the pooled covariance matrix.

Step 1: Compute covariance matrix.

$$\mathbf{S} = \begin{pmatrix} 1.20 & 0.85 & 0.42 & 0.31 \\ 0.85 & 1.05 & 0.51 & 0.28 \\ 0.42 & 0.51 & 0.78 & 0.15 \\ 0.31 & 0.28 & 0.15 & 0.55 \end{pmatrix}.$$

Step 2: Eigendecomposition. eigenvalues: $\lambda_1 = 2.35$, $\lambda_2 = 0.72$, $\lambda_3 = 0.38$, $\lambda_4 = 0.13$.

Proportion of variance: PC1 captures $2.35/3.58 = 66\%$, PC2 captures 20% , PC3 captures 11% , PC4 captures 4% .

Step 3: Interpret loadings. The first eigenvector has all positive loadings: $(0.58, 0.56, 0.42, 0.31)$. This represents overall “size”—individuals with high PC1 scores are large on all traits.

The second eigenvector has mixed signs: $(0.21, 0.18, -0.65, 0.71)$. This contrasts traits 3 and 4—individuals with high PC2 scores have relatively low trait 3 and high trait 4.

Step 4: Biological interpretation. With 66% of variance in PC1 (size), the dominant pattern is allometric scaling. PC2 captures shape variation independent of size. For many biological questions, PC1 and PC2 together (86% of variance) provide an adequate low-dimensional summary.

11.11 Summary

In this chapter we have:

- Connected the geometric framework to statistical methods: PCA, MANOVA, discriminant analysis, and CCA are all eigendecompositions.
- Detailed PCA as eigendecomposition of the covariance matrix, finding directions of maximum variance.
- Explained MANOVA as a comparison of among-group (\mathbf{B}) to within-group (\mathbf{W}) variation, testing whether groups differ.
- Introduced discriminant analysis for finding directions that best separate groups and for classification.
- Described CCA for finding maximally correlated combinations of two variable sets.
- Presented the projection pursuit framework as a unifying perspective: all methods find “interesting” directions by optimising some criterion.
- Discussed methods for comparing G matrices: common principal components, random skewers, and subspace comparison.
- Addressed estimation challenges: sample size requirements, shrinkage, regularisation, and Bayesian approaches.

- Introduced visualisation tools: scree plots, biplots, heatmaps.
- Worked through a PCA example, interpreting eigenvalues as variance captured and eigenvectors as biological patterns.

With this chapter, we have completed the core of Part IV. The matrices G , P , and γ are no longer abstract—they are objects we can estimate, visualise, compare, and interpret. In Part V, we turn to practice: worked examples that integrate all these ideas, and extensions that connect to current research on directional heritability and evolutionary constraint.

Part V

Practice and Extensions

Chapter 12

Worked Examples: Complete Analyses

This chapter brings together everything we have learned. We work through complete analyses from raw data to biological interpretation, showing each step explicitly. The goal is not just to demonstrate techniques, but to illustrate the thought process: when to use each tool, how to check assumptions, and how to connect mathematical results to biological meaning.

We present three examples of increasing complexity:

1. A two-trait analysis of a G matrix, computing directional heritabilities by hand.
2. A four-trait analysis comparing G and P, with P-whitening.
3. A selection analysis combining β and γ with the G matrix.

Each example follows the same arc: state the data, check assumptions, compute the relevant eigendecompositions, and interpret the results biologically.

12.1 Example 1: Two-trait G matrix analysis

The data

A plant breeding program has estimated the following additive genetic covariance matrix for flowering time (days) and plant height (cm):

$$\mathbf{G} = \begin{pmatrix} 25 & 15 \\ 15 & 36 \end{pmatrix}.$$

The phenotypic covariance matrix is:

$$\mathbf{P} = \begin{pmatrix} 50 & 20 \\ 20 & 60 \end{pmatrix}.$$

Our goals are to:

1. Find the principal axes of genetic variation.
2. Compute heritability in several directions.
3. Identify the directions of maximum and minimum heritability.

Chapter 12 Example 1: Two-Trait G Matrix Analysis

The G ellipse (genetic) sits inside the P ellipse (phenotypic); their ratio varies by direction.

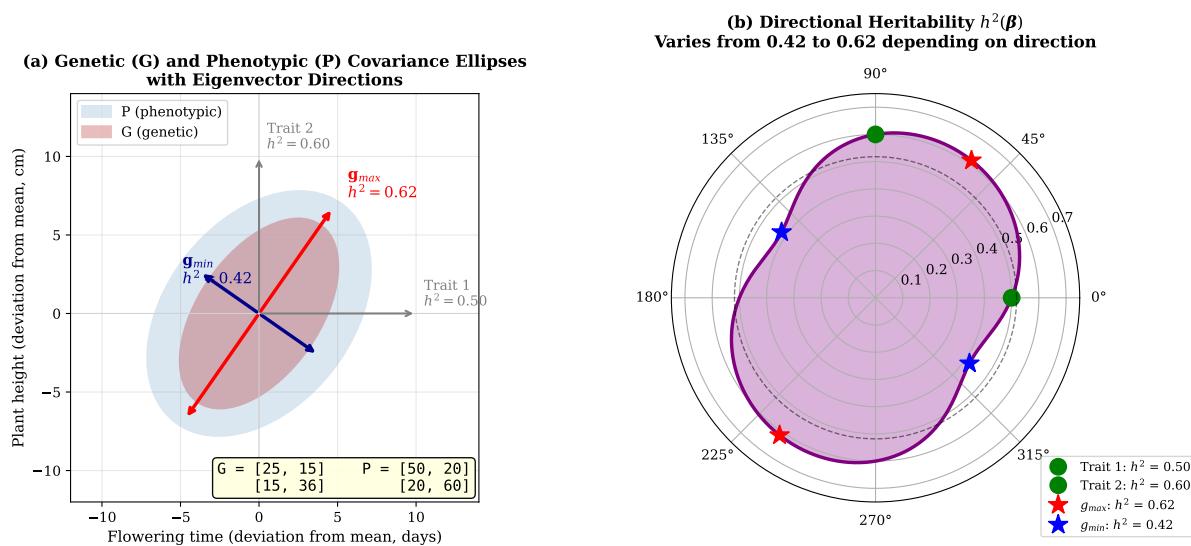


Figure 12.1: Complete analysis of the two-trait worked example. (a) Genetic (G, blue) and phenotypic (P, green) ellipses in original trait space, with g_{max} and g_{min} marked. (b) The whitened view: G^* ellipse inside the P-sphere, showing directions of extreme heritability. (c) Directional heritability $h^2(\theta)$ as a function of direction, ranging from 0.42 to 0.62. (d) Summary statistics from the analysis. (e) The breeder's equation in action: selection gradient β is deflected toward g_{max} in the response $\Delta\bar{z}$. (f) Variance decomposition by direction, showing the gap between phenotypic and genetic variance (environmental variance) varies with direction.

Step 1: Eigendecompose \mathbf{G}

The characteristic equation for \mathbf{G} is:

$$\det(\mathbf{G} - \lambda\mathbf{I}) = (25 - \lambda)(36 - \lambda) - 15^2 = 0.$$

Expanding:

$$\lambda^2 - 61\lambda + (25 \times 36 - 225) = \lambda^2 - 61\lambda + 675 = 0.$$

Using the quadratic formula:

$$\lambda = \frac{61 \pm \sqrt{61^2 - 4 \times 675}}{2} = \frac{61 \pm \sqrt{3721 - 2700}}{2} = \frac{61 \pm \sqrt{1021}}{2} = \frac{61 \pm 31.95}{2}.$$

So $\lambda_1 = 46.48$ and $\lambda_2 = 14.52$.

For $\lambda_1 = 46.48$, the eigenvector satisfies:

$$\begin{pmatrix} 25 - 46.48 & 15 \\ 15 & 36 - 46.48 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \mathbf{0}.$$

From the first row: $-21.48v_1 + 15v_2 = 0$, so $v_2/v_1 = 21.48/15 = 1.43$.

Normalising: $\mathbf{g}_{\max} = (0.573, 0.820)^\top$.

Similarly, $\mathbf{g}_{\min} = (0.820, -0.573)^\top$ (orthogonal to \mathbf{g}_{\max}).

Interpretation. The direction of maximum genetic variance points mostly toward height (coefficient 0.820) with a positive contribution from flowering time (0.573). Plants with high breeding values tend to be both tall and late-flowering. The genetic variance along this axis is 46.48.

The direction of minimum genetic variance contrasts the traits: tall but early-flowering, or short but late-flowering. Genetic variance along this axis is only 14.52—about one-third of the maximum.

Step 2: Compute univariate heritabilities

For each trait separately:

$$h_{\text{time}}^2 = \frac{G_{11}}{P_{11}} = \frac{25}{50} = 0.50, \quad h_{\text{height}}^2 = \frac{G_{22}}{P_{22}} = \frac{36}{60} = 0.60.$$

Both traits have moderate heritability.

Step 3: Compute directional heritabilities

For an arbitrary direction β , the directional heritability is:

$$h^2(\beta) = \frac{\beta^\top \mathbf{G} \beta}{\beta^\top \mathbf{P} \beta}.$$

Along \mathbf{g}_{\max} . Let $\beta = (0.573, 0.820)^\top$.

Numerator:

$$\begin{aligned}\beta^\top \mathbf{G} \beta &= (0.573, 0.820) \begin{pmatrix} 25 & 15 \\ 15 & 36 \end{pmatrix} \begin{pmatrix} 0.573 \\ 0.820 \end{pmatrix} \\ &= (0.573, 0.820) \begin{pmatrix} 26.63 \\ 38.12 \end{pmatrix} = 46.48.\end{aligned}$$

Denominator:

$$\begin{aligned}\beta^\top \mathbf{P} \beta &= (0.573, 0.820) \begin{pmatrix} 50 & 20 \\ 20 & 60 \end{pmatrix} \begin{pmatrix} 0.573 \\ 0.820 \end{pmatrix} \\ &= (0.573, 0.820) \begin{pmatrix} 45.05 \\ 60.66 \end{pmatrix} = 75.55.\end{aligned}$$

So $h^2(\mathbf{g}_{\max}) = 46.48/75.55 = 0.615$.

Along \mathbf{g}_{\min} . Let $\beta = (0.820, -0.573)^\top$.

By similar calculation:

$$\beta^\top \mathbf{G} \beta = 14.52, \quad \beta^\top \mathbf{P} \beta = 34.45.$$

So $h^2(\mathbf{g}_{\min}) = 14.52/34.45 = 0.421$.

Along flowering time only. Let $\beta = (1, 0)^\top$.

Then $h^2 = G_{11}/P_{11} = 25/50 = 0.50$.

Step 4: Find extreme heritabilities via \mathbf{G}^*

To find the true maximum and minimum heritabilities across all directions, we compute $\mathbf{G}^* = \mathbf{P}^{-1/2} \mathbf{G} \mathbf{P}^{-1/2}$ and find its eigenvalues.

First, eigendecompose \mathbf{P} :

$$\mathbf{P} = \mathbf{V}_P \Lambda_P \mathbf{V}_P^\top.$$

The eigenvalues of \mathbf{P} are $\lambda_{P,1} = 75.62$ and $\lambda_{P,2} = 34.38$, with corresponding eigenvectors.

Then:

$$\mathbf{P}^{-1/2} = \mathbf{V}_P \Lambda_P^{-1/2} \mathbf{V}_P^\top.$$

Computing $\mathbf{G}^* = \mathbf{P}^{-1/2} \mathbf{G} \mathbf{P}^{-1/2}$ and finding its eigenvalues gives:

$$\lambda_1^* = 0.619, \quad \lambda_2^* = 0.419.$$

These are the maximum and minimum directional heritabilities.

Key Idea

The eigenvalues of $\mathbf{G}^* = \mathbf{P}^{-1/2} \mathbf{G} \mathbf{P}^{-1/2}$ give the extreme directional heritabilities. Here, heritability ranges from 0.42 to 0.62 depending on direction—a range of 0.20, or about 40% of the minimum value.

Summary table

Direction	Genetic variance	heritability
Flowering time only	25.0	0.50
Height only	36.0	0.60
\mathbf{g}_{\max}	46.5	0.62
\mathbf{g}_{\min}	14.5	0.42
Maximum h^2 direction	—	0.62
Minimum h^2 direction	—	0.42

In this example, the direction of maximum genetic variance (\mathbf{g}_{\max}) is close to the direction of maximum heritability, but they need not coincide in general. The former maximises $\beta^T \mathbf{G} \beta$; the latter maximises the ratio $\beta^T \mathbf{G} \beta / \beta^T \mathbf{P} \beta$. When \mathbf{G} and \mathbf{P} have different orientations, these directions can differ substantially.

12.2 Example 2: Four-trait G-P comparison with whitening

The data

A study of a passerine bird population estimates \mathbf{G} and \mathbf{P} for four morphological traits: wing length, tarsus length, bill depth, and bill width. The matrices are:

$$\mathbf{G} = \begin{pmatrix} 0.80 & 0.45 & 0.20 & 0.15 \\ 0.45 & 0.60 & 0.25 & 0.18 \\ 0.20 & 0.25 & 0.35 & 0.28 \\ 0.15 & 0.18 & 0.28 & 0.30 \end{pmatrix},$$

$$\mathbf{P} = \begin{pmatrix} 1.20 & 0.55 & 0.30 & 0.22 \\ 0.55 & 0.95 & 0.35 & 0.25 \\ 0.30 & 0.35 & 0.55 & 0.40 \\ 0.22 & 0.25 & 0.40 & 0.50 \end{pmatrix}.$$

Step 1: Basic checks

Before analysis, we verify that both matrices are positive definite (all eigenvalues positive) and that $G_{ii} \leq P_{ii}$ for each trait (genetic variance should not exceed phenotypic variance).

eigenvalues of \mathbf{G} : 1.36, 0.44, 0.21, 0.04. All positive— \mathbf{G} is positive definite.

eigenvalues of \mathbf{P} : 1.95, 0.67, 0.46, 0.12. All positive— \mathbf{P} is positive definite.

Diagonal check: $G_{ii} \leq P_{ii}$ for all i . Yes: $0.80 < 1.20$, $0.60 < 0.95$, $0.35 < 0.55$, $0.30 < 0.50$.

Step 2: Univariate heritabilities

Trait	G_{ii}	P_{ii}	h^2
Wing length	0.80	1.20	0.67
Tarsus length	0.60	0.95	0.63
Bill depth	0.35	0.55	0.64
Bill width	0.30	0.50	0.60

All traits have similar, moderately high heritabilities (0.60–0.67). A univariate analysis would conclude that all four traits are roughly equally heritable. But this masks important directional variation.

Step 3: Compute \mathbf{G}^* and its eigenstructure

We compute $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{GP}^{-1/2}$ using the eigendecomposition of \mathbf{P} .

The eigenvalues of \mathbf{G}^* are the directional heritabilities along the principal axes of P-whitened space:

λ_1^*	λ_2^*	λ_3^*	λ_4^*
0.71	0.65	0.45	0.34

Step 4: Interpret the heritability distribution

The eigenvalues range from 0.34 to 0.71. This means:

- Maximum directional heritability: 0.71 (71% of variance genetic).
- Minimum directional heritability: 0.34 (34% genetic).
- Range: 0.37 (more than double the minimum).

Mean heritability (average of eigenvalues): $\bar{h}^2 = 0.54$.

Coefficient of variation of eigenvalues:

$$\text{CV}(\lambda^*) = \frac{\text{SD}(\lambda^*)}{\text{mean}(\lambda^*)} = \frac{0.150}{0.54} = 0.28.$$

From the formula $\text{CV}(h^2) = \sqrt{2/(p+2)} \times \text{CV}(\lambda^*)$:

$$\text{CV}(h^2) = \sqrt{2/6} \times 0.28 = 0.577 \times 0.28 = 0.16.$$

The coefficient of variation of directional heritability is about 16%. This indicates moderate constraint heterogeneity—some directions are substantially more heritable than others.

Step 5: Identify constraint traps

The eigenvector corresponding to $\lambda_4^* = 0.34$ defines the direction of minimum heritability. Examining its loadings:

$$\mathbf{v}_4^* = (0.14, -0.19, 0.66, -0.71)^\top.$$

This direction contrasts the bill traits (depth positive, width negative) with small contributions from the body-size traits. Selection for birds with deep but narrow bills, or vice versa, would face a constraint trap: only 34% of phenotypic variance in this direction is genetic.

In contrast, the direction of maximum heritability ($\lambda_1^* = 0.71$) has loadings:

$$\mathbf{v}_1^* = (0.79, 0.61, 0.05, 0.04)^\top.$$

This direction loads heavily on wing and tarsus—overall body “size.” Selection for larger or smaller birds has high heritability; 71% of variance is genetic.

Key Idea

Size variation has high heritability (71%); bill-shape variation has lower heritability (34%). A breeding program targeting bill proportions would face stronger constraints than one targeting overall body size.

Step 6: Geometric interpretation

Figure 12.2 illustrates these results geometrically.

In the original trait coordinates, the phenotypic covariance matrix \mathbf{P} defines an ellipsoid whose cross-sections in any two-trait plane are ellipses. The genetic covariance matrix \mathbf{G} defines a second ellipsoid nested inside the first. For this bird population, the \mathbf{G} ellipsoid is somewhat narrower along directions involving bill shape than along overall size.

whitening by $\mathbf{P}^{-1/2}$ maps the \mathbf{P} ellipsoid to a sphere: in whitened coordinates every direction has unit phenotypic variance. In this whitened space, \mathbf{G}^* appears as an ellipsoid whose axes have lengths given by the square roots of the eigenvalues λ_i^* . The long axis corresponds to the high-heritability “size” direction ($h^2 = 0.71$); the short axis corresponds to the low-heritability “shape” direction ($h^2 = 0.34$).

12.3 Example 3: Selection analysis with \mathbf{G} and γ

The data

A study measures survival in relation to two traits (standardised to mean zero, unit variance). The estimated selection gradients are:

$$\boldsymbol{\beta} = \begin{pmatrix} 0.18 \\ 0.12 \end{pmatrix}, \quad \boldsymbol{\gamma} = \begin{pmatrix} -0.15 & 0.08 \\ 0.08 & -0.10 \end{pmatrix}.$$

The \mathbf{G} matrix (in standardised units) is:

$$\mathbf{G} = \begin{pmatrix} 0.45 & 0.30 \\ 0.30 & 0.35 \end{pmatrix}.$$

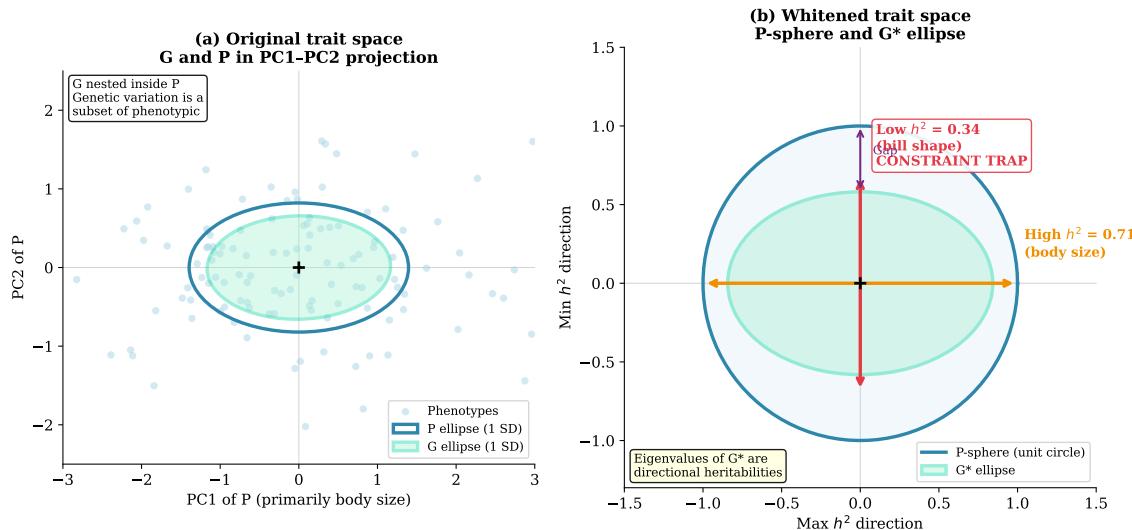


Figure 12.2: Schematic view of the four-trait bird example. (a) Genetic (**G**) and phenotypic (**P**) ellipses in a two-trait projection of the original trait space. (b) Whitened trait space: the **P**-sphere (unit circle) and the **G*** ellipse showing directional heritabilities. The long axis corresponds to the high-heritability size direction ($h^2 = 0.71$); the short axis corresponds to the low-heritability bill-shape direction ($h^2 = 0.34$), a constraint trap.

Step 1: Interpret β

Both elements of β are positive: directional selection favours increases in both traits. Trait 1 is under stronger directional selection (0.18 vs. 0.12).

The magnitude $\|\beta\| = \sqrt{0.18^2 + 0.12^2} = 0.216$ gives overall strength of directional selection.

The direction of selection:

$$\frac{\beta}{\|\beta\|} = (0.832, 0.555)^\top.$$

Step 2: Interpret γ

Both diagonal elements are negative: stabilising selection on each trait individually. The off-diagonal is positive: correlational selection favours positive trait combinations (both high or both low together).

Eigendecomposition of γ :

$$\lambda_1^\gamma = -0.04, \quad \lambda_2^\gamma = -0.21.$$

Both negative, confirming overall stabilising selection. But selection is much stronger ($|\lambda| = 0.21$) along the second eigenvector than the first ($|\lambda| = 0.04$).

The eigenvectors are:

$$\begin{aligned} \mathbf{v}_1^\gamma &= (0.59, 0.81)^\top \text{ (weak stabilising)}, \\ \mathbf{v}_2^\gamma &= (0.81, -0.59)^\top \text{ (strong stabilising)}. \end{aligned}$$

Interpretation. Stabilising selection is weak along the direction where both traits increase together—a “size” axis. Stabilising selection is strong along the direction where traits oppose each other—a “contrast” axis. The fitness surface is elongated: a ridge running along the positive diagonal, with steep sides.

Step 3: Predict response to selection

Using the Lande equation:

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\boldsymbol{\beta} = \begin{pmatrix} 0.45 & 0.30 \\ 0.30 & 0.35 \end{pmatrix} \begin{pmatrix} 0.18 \\ 0.12 \end{pmatrix} = \begin{pmatrix} 0.117 \\ 0.096 \end{pmatrix}.$$

The predicted response is $(0.117, 0.096)$ —increases in both traits, with trait 1 responding slightly more.

The response direction is:

$$\frac{\Delta \bar{\mathbf{z}}}{\|\Delta \bar{\mathbf{z}}\|} = (0.773, 0.634)^\top.$$

Comparing to the selection direction $(0.832, 0.555)^\top$, we see the response is deflected toward the direction of higher genetic variance (trait 1), but the deflection is modest because the genetic correlation is positive and selection favours both traits.

Step 4: Alignment of \mathbf{G} and $\boldsymbol{\gamma}$

Eigendecompose \mathbf{G} :

$$\lambda_1^G = 0.70, \quad \lambda_2^G = 0.10.$$

The eigenvector for λ_1^G is $(0.76, 0.65)^\top$ —similar to the direction of weak stabilising selection (\mathbf{v}_1^γ).

Key Idea

The direction of maximum genetic variance (\mathbf{g}_{\max}) aligns with the direction of weak stabilising selection. This is favourable: the population can move along the fitness ridge without fighting strong curvature. Evolution is channeled but not blocked.

Conversely, the direction of minimum genetic variance aligns with the direction of strong stabilising selection. Even if selection pushed toward unusual trait combinations (one high, one low), the population would struggle to respond because:

1. Genetic variance is low in that direction ($\lambda_2^G = 0.10$).
2. Stabilising selection is strong ($\lambda_2^\gamma = -0.21$).

This alignment is likely not coincidental. Theory predicts that mutation-selection balance tends to erode variance in directions of strong stabilising selection while preserving variance along fitness ridges.

Step 5: Long-term prediction

Under mutation-selection balance, the population is expected to maintain variation primarily along \mathbf{g}_{\max} (the fitness ridge). The combination of high genetic variance, weak stabilising selection, and positive correlational selection along the size axis suggests that size variation will persist. Variation in trait contrast will be more rapidly eroded.

This example illustrates why both \mathbf{G} and γ matter. Knowing \mathbf{G} alone tells us about evolutionary potential; knowing γ alone tells us about the fitness landscape. Only by comparing their geometries can we predict whether evolution will be fast or slow, direct or deflected.

12.4 Computational tools

All calculations in this chapter can be performed by hand for two traits, but become tedious for more. Here is R code implementing the key steps:

```
# Given G and P matrices (bird example)
G <- matrix(c(0.80, 0.45, 0.20, 0.15,
             0.45, 0.60, 0.25, 0.18,
             0.20, 0.25, 0.35, 0.28,
             0.15, 0.18, 0.28, 0.30), 4, 4)

P <- matrix(c(1.20, 0.55, 0.30, 0.22,
             0.55, 0.95, 0.35, 0.25,
             0.30, 0.35, 0.55, 0.40,
             0.22, 0.25, 0.40, 0.50), 4, 4)

# Step 1: Basic checks
cat("G eigenvalues:", round(eigen(G)$values, 3), "\n")
cat("P eigenvalues:", round(eigen(P)$values, 3), "\n")
cat("All positive?", all(eigen(G)$values > 0) &
    all(eigen(P)$values > 0), "\n")

# Step 2: Compute P^{-1/2}
eig_P <- eigen(P)
V_P <- eig_P$vectors
P_inv_sqrt <- V_P %*% diag(1/sqrt(eig_P$values)) %*% t(V_P)

# Step 3: Compute G*
G_star <- P_inv_sqrt %*% G %*% P_inv_sqrt

# Step 4: Eigenvalues of G* are directional heritabilities
eig_Gstar <- eigen(G_star)
h2_dir <- eig_Gstar$values
cat("Directional heritabilities:", round(h2_dir, 3), "\n")
cat("Max h2:", round(max(h2_dir), 3), "\n")
cat("Min h2:", round(min(h2_dir), 3), "\n")
```

```

# Step 5: CV of directional heritability
mean_h2 <- mean(h2_dir)
sd_h2 <- sd(h2_dir)
cv_lambda <- sd_h2 / mean_h2
p <- length(h2_dir)
cv_h2 <- sqrt(2/(p+2)) * cv_lambda
cat("Mean h2:", round(mean_h2, 3), "\n")
cat("CV(h2):", round(cv_h2, 3), "\n")

# Step 6: Identify constraint directions
cat("\nMax h2 direction (loadings):\n")
print(round(eig_Gstar$vectors[, 1], 2))
cat("\nMin h2 direction (loadings):\n")
print(round(eig_Gstar$vectors[, p], 2))

```

12.5 Summary

In this chapter we have:

- Worked through a complete two-trait analysis: eigendecomposition of G , calculation of directional heritabilities, and identification of extreme values via G^* .
- Extended to four traits, demonstrating P-whitening and interpretation of the eigenvalue spectrum as the distribution of directional heritability.
- Combined G with γ to analyse how genetic constraint interacts with the fitness surface geometry.
- Shown that alignment between G and γ determines whether evolution is facilitated or frustrated.
- Provided R code for computing G^* and its eigenstructure.

These examples illustrate the payoff of the geometric perspective. Matrices are not just tables of numbers—they are shapes that constrain and channel evolution. By visualising G , P , and γ as ellipsoids and understanding their eigenstructure, we gain insight into evolutionary potential and constraint that would be invisible from univariate analyses alone.

Exercises

Exercise 1 (Covariance ellipses with the same trace). Construct two 2×2 covariance matrices that have the same trace (sum of diagonal elements) but different eigenvalues. For each matrix:

1. Compute the eigenvalues and eigenvectors.
2. Sketch the corresponding covariance ellipse.
3. Explain how the trace can be the same while the shape differs.

Exercise 2 (\mathbf{G}^* and the P-sphere for two traits). Using the plant example from Example 1:

1. Compute $\mathbf{P}^{-1/2}$ explicitly.
2. Compute $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{GP}^{-1/2}$.
3. Draw the unit circle (the P-sphere) and sketch the ellipse defined by \mathbf{G}^* .
4. Mark the directions of maximum and minimum heritability.

Exercise 3 (Directional heritability in a chosen direction). In the bird example from Example 2:

1. Define a direction corresponding to “bill shape” (e.g., increasing bill depth while decreasing bill width).
2. Compute $h^2(\mathbf{u})$ for this direction.
3. Compare to the extreme values from \mathbf{G}^* .

Exercise 4 ($\mathbf{G}\text{-}\gamma$ alignment). Consider the selection example in Example 3:

1. Compute the angle between \mathbf{g}_{\max} and \mathbf{v}_1^γ .
2. How would the evolutionary trajectory change if these were perpendicular?
3. Describe a scenario where misalignment would strongly frustrate evolutionary change.

Chapter 13

directional heritability and the Geometry of Constraint

This final chapter connects the geometric framework to a frontier research question: how does heritability vary across directions in trait space, and what does this variation tell us about evolutionary constraint?

We have seen that the eigenvalues of $\mathbf{G}^* = \mathbf{P}^{-1/2}\mathbf{GP}^{-1/2}$ are the directional heritabilities along principal axes. But most selection does not align with principal axes. What is the *distribution* of heritability across all possible directions? How do we characterise, measure, and interpret this distribution?

These questions lead to the concept of **constraint heterogeneity**: the degree to which heritability varies across directions. When constraint heterogeneity is high, some directions are evolutionary highways while others are dead ends. Understanding this heterogeneity is essential for predicting evolutionary trajectories and designing effective breeding programs.

13.1 From eigenvalues to distributions

The eigenvalues of \mathbf{G}^* give us the extreme directional heritabilities:

$$h_{\min}^2 = \lambda_p^* \leq h^2(\boldsymbol{\beta}) \leq \lambda_1^* = h_{\max}^2.$$

But what about all the directions in between? If we sample directions uniformly from the P-sphere (the set of unit phenotypic variance directions), what distribution of h^2 values do we observe?

From Chapter 21, we know that the quadratic form $h^2(\boldsymbol{\beta}) = \boldsymbol{\beta}^\top \mathbf{G}^* \boldsymbol{\beta}$ (for unit vectors in whitened space) is a weighted average of the eigenvalues λ_i^* , with weights given by squared projections onto eigenvectors.

The mathematics of random quadratic forms gives us a precise characterisation. For a random unit vector uniformly distributed on the sphere, the variance of the quadratic form is:

$$\text{Var}[h^2(\boldsymbol{\beta})] = \frac{2}{p(p+2)} \sum_{i < j} (\lambda_i^* - \lambda_j^*)^2 = \frac{2}{p+2} \text{Var}(\lambda^*),$$

where $\text{Var}(\lambda^*)$ is the variance of the eigenvalues of \mathbf{G}^* .

Key Idea

The variance of directional heritability across random directions is proportional to the variance of the eigenvalues of \mathbf{G}^* . If eigenvalues are similar, heritability is similar in all directions. If eigenvalues differ greatly, heritability varies dramatically with direction.

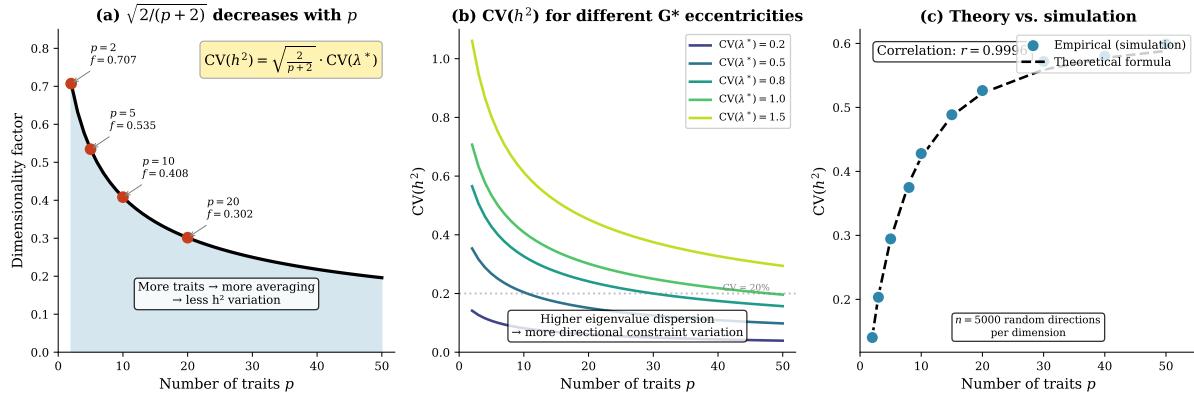


Figure 13.1: How dimensionality affects the variation in directional heritability. (a) The dimensionality factor $\sqrt{2/(p+2)}$ decreases with the number of traits p . In two dimensions, this factor equals 0.707; by $p = 50$, it has fallen to 0.196. This decay occurs because random directions in high dimensions tend to “average out” the eigenvalues of \mathbf{G}^* , reducing the variation in $h^2(\beta)$ across directions. (b) The coefficient of variation of directional heritability, $CV(h^2) = \sqrt{2/(p+2)} \times CV(\lambda^*)$, shown for different levels of eigenvalue dispersion. Higher $CV(\lambda^*)$ (more eccentric \mathbf{G}^* ellipsoid) produces greater directional variation in heritability, but this effect is attenuated as dimensionality increases. (c) Simulation verification. Blue points show empirical $CV(h^2)$ computed from 5,000 random directions sampled uniformly on the unit sphere; the dashed line shows the theoretical prediction. The near-perfect agreement ($r > 0.99$) validates the formula $CV^2(h^2) = \frac{2}{p+2} \times V_{\text{rel}}(\mathbf{G}^*)$ derived in Section 13.1.

13.2 The coefficient of variation of directional heritability

A natural measure of constraint heterogeneity is the coefficient of variation of directional heritability:

$$CV(h^2) = \frac{SD(h^2)}{E(h^2)}.$$

Using the results above and the fact that $E(h^2) = \bar{\lambda}^*$ (the mean eigenvalue), we obtain:

$$CV(h^2) = \sqrt{\frac{2}{p+2}} \times CV(\lambda^*),$$

where $CV(\lambda^*) = SD(\lambda^*)/\bar{\lambda}^*$ is the coefficient of variation of the eigenvalues.

This formula reveals two factors controlling heritability variation:

1. **eigenvalue dispersion:** $CV(\lambda^*)$ measures how different the principal heritabilities are. Large dispersion means some directions have much higher heritability than others.

2. Dimensionality: The factor $\sqrt{2/(p+2)}$ decreases with the number of traits p . In high dimensions, random directions tend to “average out” the eigenvalues, reducing heritability variation.

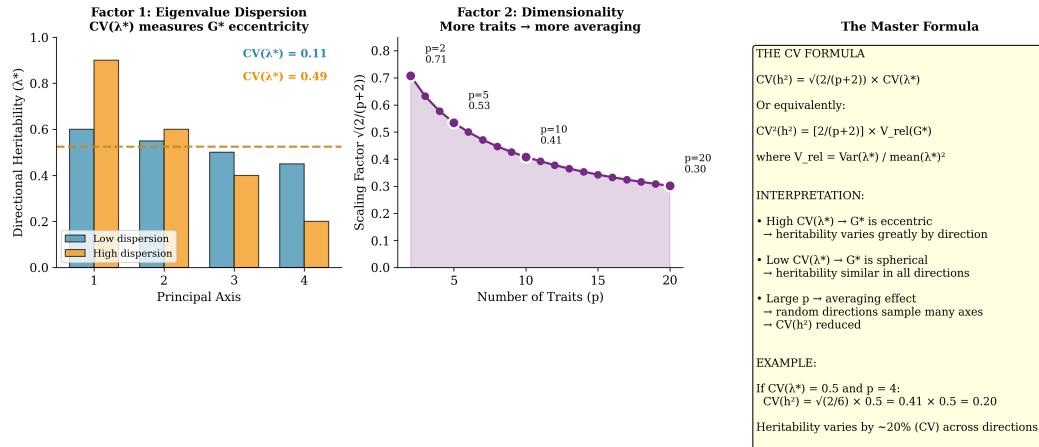


Figure 13.2: heritability

] The coefficient of variation of directional heritability depends on two factors: the dispersion of G^* eigenvalues and the number of traits. Higher eigenvalue dispersion increases CV; more traits decrease CV through averaging.

13.3 eigenvalue

variance]Relative eigenvalue variance: $V_{rel}(G^*)$

A key quantity is the relative variance of the eigenvalues:

$$V_{rel}(G^*) = \frac{\text{Var}(\lambda^*)}{\bar{\lambda}^{*2}} = CV(\lambda^*)^2.$$

This dimensionless quantity measures the “eccentricity” of the G^* ellipsoid. When $V_{rel} = 0$, all eigenvalues are equal (G^* is spherical, no constraint). As V_{rel} increases, the ellipsoid becomes more elongated and constraint heterogeneity increases.

The formula for $CV(h^2)$ becomes:

$$CV(h^2) = \sqrt{\frac{2}{p+2} \times V_{rel}(G^*)}.$$

This elegant relationship connects three quantities:

- $CV(h^2)$: observable variation in heritability across directions.
- $V_{rel}(G^*)$: geometric property of the G-P relationship.
- p : the number of traits.

Key Idea

The formula $\text{CV}^2(h^2) = \frac{2}{p+2} \times V_{\text{rel}}(\mathbf{G}^*)$ connects the distributional property (heritability variation) to the geometric property (\mathbf{G}^* eccentricity). This is the bridge between theory and empirical observation.

13.4 Constraint traps revisited

In Chapter 21 we introduced constraint traps: directions where phenotypic variance is normal but heritability is low. Now we can quantify how severe these traps are.

The minimum directional heritability is λ_p^* , the smallest eigenvalue of \mathbf{G}^* . If λ_p^* is much smaller than the mean $\bar{\lambda}^*$, there exist directions where selection will be ineffective despite observable phenotypic variation.

Define the **constraint severity** as:

$$\text{Severity} = 1 - \frac{\lambda_p^*}{\bar{\lambda}^*}.$$

This measures how much worse the worst direction is compared to the average. Severity of 0 means all directions are equally heritable; severity approaching 1 means the worst direction has near-zero heritability.

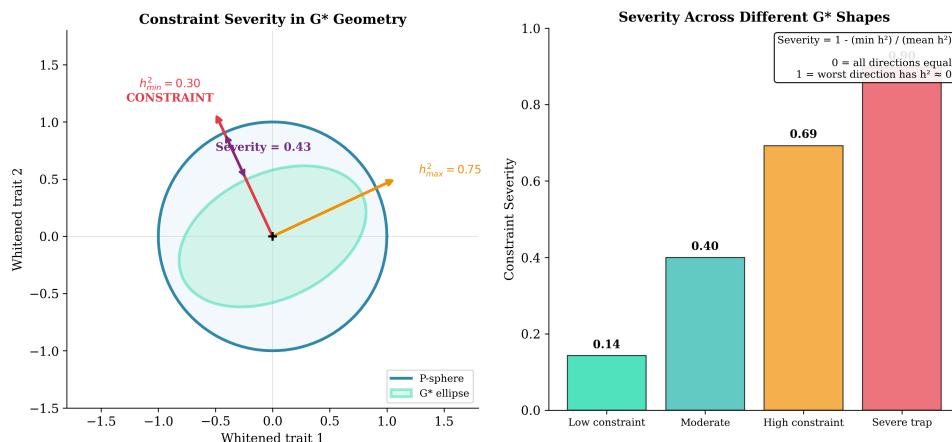


Figure 13.3: Constraint severity measures how much worse the minimum heritability is compared to the mean. High severity indicates the existence of severe constraint traps.

13.5 Selection strength and constraint risk

Whether a constraint trap matters biologically depends on selection strength. Consider a selection gradient β with magnitude $\|\beta\|$.

The expected response to selection is:

$$\text{E}[\Delta \bar{z}] \approx \bar{\lambda}^* \times \|\beta\|$$

for random direction (roughly speaking).

But the variance of response depends on heritability variance:

$$\text{Var}[\Delta \bar{z}] \propto \text{Var}(h^2) \times \|\beta\|^2.$$

When selection is strong ($\|\beta\|$ large), even modest $\text{CV}(h^2)$ produces large variation in response. Some selection directions produce robust response; others produce disappointing results.

This interaction matters for:

- **Breeding programs:** Strong artificial selection may hit constraint traps.
- **Conservation:** Populations facing strong environmental change may be unable to respond if change targets constrained directions.
- **Evolutionary prediction:** Predicting long-term response requires knowing not just mean heritability but its directional distribution.

13.6 Empirical estimation

To apply these concepts, we need to estimate G and P from data and compute \mathbf{G}^* and its eigenvalues.

Estimation pipeline

1. **Estimate G and P.** Use standard quantitative genetics methods: half-sib designs, parent-offspring regression, animal models, or genomic approaches.
2. **Check positive definiteness.** Ensure both G and P have all positive eigenvalues. Sampling error can produce negative eigenvalues in G, especially for the smallest values.
3. **Compute P-whitening matrix.** Eigendecompose P and construct $\mathbf{P}^{-1/2}$.
4. **Compute \mathbf{G}^* .** Form $\mathbf{G}^* = \mathbf{P}^{-1/2} \mathbf{G} \mathbf{P}^{-1/2}$.
5. **Eigendecompose \mathbf{G}^* .** The eigenvalues are directional heritabilities; eigenvectors are the principal constraint axes.
6. **Compute summary statistics.** Mean, variance, CV of eigenvalues; constraint severity; V_{rel} .

R implementation

```
compute_h2_distribution <- function(G, P) {
  # Check positive definite\index{subject}{matrix!positive definite}ness
  if (any(eigen(G)$values <= 0))
    warning("G has non-positive eigenvalues")
  if (any(eigen(P)$values <= 0))
```

```

stop("P must be positive definite\\index[subject]{matrix!positive definite}")

# P-whitening
eig_P <- eigen(P)
P_inv_sqrt <- eig_P$vectors %*%
  diag(1/sqrt(eig_P$values)) %*%
  t(eig_P$vectors)

# Compute G*
G_star <- P_inv_sqrt %*% G %*% P_inv_sqrt

# Eigendecompose G*
eig_Gstar <- eigen(G_star)
lambda_star <- eig_Gstar$values

# Summary statistics
p <- nrow(G)
mean_h2 <- mean(lambda_star)
var_lambda <- var(lambda_star)
cv_lambda <- sqrt(var_lambda) / mean_h2
V_rel <- var_lambda / mean_h2^2
cv_h2 <- sqrt(2/(p+2)) * cv_lambda

h2_min <- min(lambda_star)
severity <- 1 - h2_min / mean_h2

list(
  lambda_star = lambda_star,
  mean_h2 = mean_h2,
  cv_h2 = cv_h2,
  V_rel = V_rel,
  h2_min = h2_min,
  h2_max = max(lambda_star),
  constraint_severity = severity,
  eigenvectors = eig_Gstar$vectors
)
}
}

```

13.7 Case study: empirical G-P analysis

Let us apply these methods to a real example. Consider a dataset of G and P matrices from a published study of floral traits in a plant species.

$$\mathbf{G} = \begin{pmatrix} 0.42 & 0.28 & 0.15 \\ 0.28 & 0.38 & 0.22 \\ 0.15 & 0.22 & 0.31 \end{pmatrix}, \quad \mathbf{P} = \begin{pmatrix} 0.85 & 0.35 & 0.20 \\ 0.35 & 0.72 & 0.28 \\ 0.20 & 0.28 & 0.55 \end{pmatrix}.$$

Results

Eigenvalues of \mathbf{G}^* : (0.71, 0.52, 0.35).

Summary statistics:

Statistic	Value
Mean h^2	0.53
Max h^2	0.71
Min h^2	0.35
CV(h^2)	0.24
$V_{\text{rel}}(\mathbf{G}^*)$	0.15
Constraint severity	0.34

Interpretation

The mean heritability is moderate (0.53), but it ranges from 0.35 to 0.71 depending on direction. The CV of 24% indicates substantial heterogeneity. The constraint severity of 0.34 means the worst direction has heritability 34% below the average.

The eigenvector for minimum heritability reveals which trait combination faces the strongest constraint. If this direction corresponds to a biologically important target (e.g., a pollinator preference axis), the breeding implications are significant.

13.8 Implications for breeding and evolution

Breeding programs

Traditional breeding often focuses on individual-trait heritabilities. Our framework reveals that:

1. The direction of selection matters as much as its strength.
2. Index selection that combines traits should be evaluated for directional heritability, not just component heritabilities.
3. Constraint traps can be identified in advance and avoided or addressed through introgression of new genetic variation.

Evolutionary biology

For understanding evolution in natural populations:

1. The rate of adaptation depends on whether selection aligns with high-heritability directions.
2. Long-term evolutionary trajectories may be biased toward \mathbf{g}_{\max} or its P-standardised equivalent.

3. Phenotypic divergence among populations may reflect the geometry of G^* as much as the direction of selection.

Conservation

For populations facing environmental change:

1. The capacity to adapt depends on whether required trait changes align with heritable directions.
2. Populations with high $CV(h^2)$ are more vulnerable—they may have adequate mean heritability but face traps in critical directions.
3. Assisted gene flow could be targeted to increase heritability in constrained directions.

13.9 Extensions and open questions

The framework presented here opens several research directions:

- **Non-Gaussian distributions:** The formula $CV^2(h^2) = \frac{2}{p+2} V_{\text{rel}}$ assumes uniform sampling on the P-sphere. For specific selection scenarios, the distribution may differ.
- **Estimation uncertainty:** G and P are estimated with error. How does this uncertainty propagate to $CV(h^2)$ and constraint severity?
- **Temporal stability:** Does the G - P geometry remain stable across generations, or does constraint heterogeneity itself evolve?
- **Environmental dependence:** Does P (and hence G^*) change across environments? Are constraint traps consistent or context-dependent?
- **Genomic architecture:** Can we predict G^* eigenstructure from knowledge of gene networks and pleiotropy?

13.10 Summary

In this chapter we have:

- Derived the distribution of directional heritability across random selection directions.
- Shown that $CV^2(h^2) = \frac{2}{p+2} V_{\text{rel}}(G^*)$ connects the distributional property to G^* eigenstructure.
- Defined constraint severity as a measure of how bad the worst constraint trap is.
- Provided an estimation pipeline and R code for computing these quantities from G and P estimates.
- Illustrated the analysis with a case study.

- Discussed implications for breeding, evolutionary biology, and conservation.
- Outlined open research questions.

This chapter brings us full circle. We began with distance and the simple question of why we square. We built up through covariance matrices, Mahalanobis distance distance, eigen-decomposition, and P-whitening. Now we can answer sophisticated questions about evolutionary constraint: not just “is this trait heritable?” but “how does heritability vary across the space of possible selection targets?”

The geometric perspective unifies it all. Matrices are shapes; eigenvalues measure extent; eigenvectors define axes. With these tools, the complexity of multivariate evolution becomes tractable—not simple, but navigable.

The ellipse has been our guide throughout. May it serve you well in your own research.

Epilogue: The Shape of Things

We began with a point in a paddock and a question about distance. We end with ellipsoids in high-dimensional space and a framework for understanding evolutionary constraint. The journey has been long, but the core idea has remained simple: *symmetric matrices describe shapes.*

This is worth dwelling on. A matrix is just a table of numbers—rows and columns, entries that can be added and multiplied. Yet when that matrix is symmetric and positive definite, it becomes something more. It becomes a geometry. The eigenvalues measure how far the shape extends; the eigenvectors point along its natural axes. The determinant captures the volume; the trace captures the total extent. Invert the matrix, and you flip the geometry inside out—what was long becomes short, what was wide becomes narrow.

Once you see matrices as shapes, the tools of multivariate statistics become intuitive. PCA is not a mysterious algorithm; it is the simple act of finding the axes of an ellipse. MANOVA is not an arbitrary test statistic; it is a comparison of two shapes—one describing variation among groups, the other describing variation within. The Mahalanobis distance distance is not a formula to memorise; it is Euclidean distance distance measured after reshaping the space to match the data.

And the G matrix—that central object of evolutionary quantitative genetics—is not merely a summary of breeding values. It is the shape of what evolution can do. Its long axes are the directions of easy change; its short axes are the directions of constraint. When we ask whether a population can respond to selection, we are asking whether the selection gradient points along a long axis or a short one. When we compare G matrices across species, we are comparing the shapes of evolutionary possibility.

What geometry gives us

The geometric perspective offers three gifts.

The first is **intuition**. Equations can be opaque; shapes are visible. When you read that the response to selection is $\Delta\bar{z} = \mathbf{G}\beta$, you might see symbols. But when you visualise G as an ellipse and β as an arrow, you see immediately that the response will be deflected toward the long axis. The algebra confirms what the picture reveals.

The second gift is **unification**. The methods scattered across textbooks—PCA, discriminant analysis, canonical correlation, MANOVA, Mahalanobis distance distance, the breeder's equation equation—are not separate techniques requiring separate intuitions. They are all eigen-decompositions, all ways of finding the natural axes of some ellipsoid. Learn the geometry once, and you understand them all.

The third gift is **new questions**. Once you see G as a shape, you can ask: how does this shape vary across environments? How does it change over evolutionary time? How does it compare to the shape of the fitness surface? These questions were always implicit in the formalism, but the geometric perspective makes them vivid. It turns parameter estimation into shape description, and shape description invites comparison.

The limits of ellipses

I have told a particular story in these notes—one centred on symmetric matrices and their ellipsoids. This is not the only story that could be told.

Ellipses assume linearity. The covariance matrix captures only the linear relationships among traits; it misses curvature, thresholds, and interactions. When the true relationships are non-linear, the ellipse is an approximation—sometimes good, sometimes misleading.

Ellipses assume multivariate normality, or at least that the second moments are sufficient statistics. For heavy-tailed distributions or multimodal populations, the covariance matrix tells only part of the story.

And ellipses assume stationarity. The G matrix is not fixed; it evolves. Mutation introduces new variation, selection erodes it, drift reshapes it. The geometry we estimate today may not be the geometry of tomorrow. The ellipse is a snapshot, not a law.

These limitations are real, but they do not diminish the value of the geometric perspective. They define its scope. Linearity is often a good first approximation. Normality is often reasonable for quantitative traits. And stationarity, while imperfect, is often sufficient for the timescales we study. The ellipse is a model, and like all models, it is useful precisely because it is simpler than reality.

The view from here

If you have followed these notes from the beginning, you now possess a way of seeing. When you encounter a covariance matrix, you see an ellipse. When you read an eigenvalue, you see an axis length. When you compute a Mahalanobis distance distance, you see a rescaling of space. This way of seeing is not a trick or a shortcut; it is the thing itself. The geometry is not a metaphor for the mathematics—it is the mathematics, visualised.

This perspective will serve you in many contexts. In quantitative genetics, it illuminates the breeder's equation equation and the geometry of constraint. In multivariate statistics, it unifies PCA, MANOVA, and discriminant analysis. In machine learning, it underlies principal component regression, regularisation, and the geometry of high-dimensional data. In physics, it connects to quadratic forms, moment of inertia tensors, and the geometry of configuration spaces.

The same ideas, the same shapes, appearing across fields. This is not coincidence. It is the deep structure of linear algebra asserting itself wherever quantities vary together.

An invitation

These notes are an invitation, not an endpoint. I have shown you how to see matrices as shapes; I have not shown you everything those shapes can reveal.

There are questions I have not addressed. How do we estimate G matrices reliably when sample sizes are small? How do we test whether two G matrices differ? How do we model the evolution of G itself? How do we connect the geometry of genetic variation to the molecular mechanisms that generate it?

These are research questions, not textbook exercises. They are the frontier. And if these notes have done their job, you are now equipped to approach that frontier with geometric intuition as your guide.

A final image

Picture a high-dimensional ellipsoid—the G matrix of some population, suspended in the space of all possible phenotypes. Its axes point in directions we cannot visualise directly, but we can measure their lengths. Some are long: abundant genetic variation, easy evolutionary change. Some are short: scarce variation, constrained response.

Now imagine a selection gradient—an arrow pointing toward some optimum, some direction that would increase fitness. The arrow may point anywhere. It may align with a long axis, and evolution will be swift. It may point toward a short axis, and evolution will be slow, frustrated, deflected.

The shape of the ellipsoid and the direction of the arrow—these two things, together, determine what will happen. The G matrix is potential; selection is actuality. Their interaction is evolution.

This is the image I hope you carry forward. Not a formula to be memorised, but a shape to be seen. Not a technique to be applied, but a geometry to be understood.

The ellipse has been our guide. May it serve you well.

Daniel Ortiz-Barrientos

Brisbane, 2025

References

- Blows, M. W. (2007)** A tale of two matrices: multivariate approaches in evolutionary biology. *Journal of Evolutionary Biology*, 20(1), 1–8.
- Lande, R. (1979)** Quantitative genetic analysis of multivariate evolution, applied to brain:body size allometry. *Evolution*, 33(1), 402–416.
- Lande, R., & Arnold, S. J. (1983)** The measurement of selection on correlated characters. *Evolution*, 37(6), 1210–1226.
- Lynch, M., & Walsh, B. (1998)** *Genetics and Analysis of Quantitative Traits*. Sinauer Associates.
- Phillips, P. C., & Arnold, S. J. (1989)** Visualizing multivariate selection. *Evolution*, 43(6), 1209–1222.
- Schlüter, D. (1996)** Adaptive radiation along genetic lines of least resistance. *Evolution*, 50(5), 1766–1774.
- Strang, G. (2016)** *Introduction to Linear Algebra* (5th ed.). Wellesley-Cambridge Press.
- Walsh, B., & Lynch, M. (2018)** *Evolution and Selection of Quantitative Traits*. Oxford University Press.

Appendix: Mathematical and Statistical Background

This appendix collects the minimum linear algebra and probability background assumed in the main text. It is not a complete course. Instead, it provides a compact reference for four key ingredients:

- vectors and matrices, and how to multiply them;
- eigenvalues and eigenvectors;
- basic probability notions: variance, covariance, and the multivariate normal;
- selection gradients and Lande's equation.

Readers who have seen these topics before can skim or skip this appendix. Readers for whom these ideas are new may find it helpful to read this appendix alongside Chapters 0–2 and Chapters 10–12.

13.11 Vectors and matrices

Vectors

A *vector* is an ordered list of numbers that we usually picture as an arrow in a trait space. For p traits, we write a phenotype as a column vector

$$\mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{pmatrix}.$$

Each entry z_i is the value of trait i . We use bold letters for vectors (\mathbf{z} , \mathbf{x} , \mathbf{u}) and reserve plain italics (z_i , x_i) for individual components.

The *length* (Euclidean norm) of a vector is

$$\|\mathbf{z}\| = \sqrt{z_1^2 + z_2^2 + \cdots + z_p^2}.$$

The *dot product* (inner product) of two vectors \mathbf{x} and \mathbf{y} is

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} = x_1 y_1 + x_2 y_2 + \cdots + x_p y_p.$$

Geometrically, this dot product relates to the angle θ between \mathbf{x} and \mathbf{y} via

$$\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\| \|\mathbf{y}\| \cos \theta.$$

Matrices as linear maps

A *matrix* is a rectangular array of numbers. An $m \times n$ matrix has m rows and n columns. We write matrices in bold capitals; for example,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}.$$

In this book, matrices are always used as *linear maps*: machines that take a vector in and return a vector out. If \mathbf{A} is $m \times n$ and \mathbf{x} is an $n \times 1$ column vector, the product $\mathbf{y} = \mathbf{Ax}$ is an $m \times 1$ column vector whose entries are

$$y_i = \sum_{j=1}^n a_{ij} x_j, \quad i = 1, \dots, m.$$

So each entry y_i is a dot product between the i th row of \mathbf{A} and \mathbf{x} .

Linear combinations view

There is another useful way to see the matrix–vector product. Let the columns of \mathbf{A} be denoted by $\mathbf{a}_1, \dots, \mathbf{a}_n$. Then

$$\mathbf{Ax} = x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \cdots + x_n \mathbf{a}_n.$$

So \mathbf{Ax} is a *linear combination* of the columns of \mathbf{A} , with coefficients x_1, \dots, x_n . This viewpoint is important when we interpret covariance matrices as sets of basis vectors that span trait space.

A simple example

Consider

$$\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}, \quad \mathbf{x} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

Then

$$\mathbf{Ax} = \begin{pmatrix} 2 \cdot 1 + 1 \cdot 2 \\ 1 \cdot 1 + 3 \cdot 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 7 \end{pmatrix}.$$

Geometrically, the matrix \mathbf{A} stretches and shears the plane. The vector $(1, 2)^\top$ is moved to $(4, 7)^\top$.

Matrix–matrix multiplication

If \mathbf{A} is $m \times n$ and \mathbf{B} is $n \times k$, we can form the product $\mathbf{C} = \mathbf{AB}$, which is $m \times k$. By definition, the (i, ℓ) entry of \mathbf{C} is

$$c_{i\ell} = \sum_{j=1}^n a_{ij} b_{j\ell}.$$

Geometrically, applying \mathbf{B} and then \mathbf{A} to a vector \mathbf{x} gives

$$\mathbf{A}(\mathbf{B}\mathbf{x}) = (\mathbf{AB})\mathbf{x}.$$

Matrix multiplication corresponds to *composition* of linear maps: do \mathbf{B} first, then \mathbf{A} . In general, we have $\mathbf{AB} \neq \mathbf{BA}$; matrix multiplication is not commutative.

13.12 eigenvalues and eigenvectors

Definition

Let \mathbf{A} be a square $p \times p$ matrix. A non-zero vector \mathbf{v} is an *eigenvector* of \mathbf{A} with eigenvalue λ if

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

This means that when \mathbf{A} acts on \mathbf{v} , it does not change its direction, only its length. The factor λ is the stretch (or compression) factor along the direction \mathbf{v} .

Symmetric matrices

In this book we work almost exclusively with symmetric matrices: $\mathbf{A} = \mathbf{A}^\top$. Covariance matrices, the \mathbf{G} and \mathbf{P} matrices, and the quadratic selection matrix γ are all symmetric by construction.

Symmetric matrices have two important properties:

- All eigenvalues are real.
- There exists an orthonormal basis of eigenvectors.

This means we can write a symmetric matrix as

$$\mathbf{A} = \mathbf{Q}\Lambda\mathbf{Q}^\top,$$

where \mathbf{Q} is an orthogonal matrix whose columns are unit eigenvectors of \mathbf{A} , and Λ is a diagonal matrix whose entries are the corresponding eigenvalues.

This decomposition is called an *eigendecomposition* or *spectral decomposition*.

A two-dimensional example

Consider

$$\mathbf{A} = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}.$$

We look for λ and non-zero $(v_1, v_2)^\top$ such that $\mathbf{Av} = \lambda\mathbf{v}$. The characteristic equation is

$$\det(\mathbf{A} - \lambda\mathbf{I}) = \det \begin{pmatrix} 3 - \lambda & 1 \\ 1 & 3 - \lambda \end{pmatrix} = (3 - \lambda)^2 - 1 = 0.$$

So $(3 - \lambda)^2 = 1$ and therefore $\lambda = 4$ or $\lambda = 2$.

For $\lambda = 4$, we solve

$$\begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \mathbf{0},$$

which yields $v_1 = v_2$. A normalised eigenvector is

$$\mathbf{v}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

For $\lambda = 2$, we solve

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \mathbf{0},$$

which yields $v_1 = -v_2$. A normalised eigenvector is

$$\mathbf{v}_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Geometrically, \mathbf{v}_1 points along the line $x = y$ and is stretched by a factor of 4, while \mathbf{v}_2 points along $x = -y$ and is stretched by a factor of 2. Any vector in the plane can be written as a combination of these two directions.

Eigenstructure of covariance matrices

A covariance matrix Σ is symmetric and positive definite (all eigenvalues positive). Its eigen-decomposition

$$\Sigma = \mathbf{Q}\Lambda\mathbf{Q}^\top$$

has a clear geometric interpretation:

- the columns of \mathbf{Q} give the principal axes (directions) of variation;
- the eigenvalues on the diagonal of Λ give the variances along those axes.

In two dimensions, the contours of a multivariate normal distribution with covariance matrix Σ are ellipses whose axes are aligned with the eigenvectors of Σ and whose axis lengths are proportional to $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$.

The same idea extends to the genetic covariance matrix \mathbf{G} and the phenotypic covariance matrix \mathbf{P} .

13.13 Basic probability and covariance

Random variables and expectation

A *random variable* X is a numerical quantity that takes different values with certain probabilities. We write its *expectation* (mean) as $\mathbb{E}[X]$. For a discrete variable taking values x_i with probabilities p_i ,

$$\mathbb{E}[X] = \sum_i x_i p_i.$$

For a continuous variable with density $f(x)$,

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f(x) dx.$$

In practice, we often estimate the mean from data x_1, \dots, x_n as the sample mean

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i.$$

Variance and covariance

The *variance* of X measures spread:

$$\text{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2].$$

The *covariance* between two random variables X and Y is

$$\text{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])].$$

Covariance is positive when large values of X tend to occur with large values of Y , negative when large values of X tend to occur with small values of Y , and near zero when there is no consistent linear relation.

From data pairs (x_i, y_i) , we estimate covariance as

$$\widehat{\text{Cov}}(X, Y) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}).$$

Covariance matrices

For a random vector $\mathbf{Z} = (Z_1, \dots, Z_p)^\top$, the covariance matrix Σ is the $p \times p$ matrix with entries

$$\Sigma_{ij} = \text{Cov}(Z_i, Z_j).$$

So

$$\Sigma = \begin{pmatrix} \text{Var}(Z_1) & \text{Cov}(Z_1, Z_2) & \cdots & \text{Cov}(Z_1, Z_p) \\ \text{Cov}(Z_2, Z_1) & \text{Var}(Z_2) & \cdots & \text{Cov}(Z_2, Z_p) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(Z_p, Z_1) & \text{Cov}(Z_p, Z_2) & \cdots & \text{Var}(Z_p) \end{pmatrix}.$$

The diagonal entries record variances of individual traits; the off-diagonals record covariances between traits. The matrices \mathbf{P} and \mathbf{G} used throughout the book are examples of covariance matrices.

The multivariate normal distribution

A p -dimensional random vector \mathbf{Z} is said to have a multivariate normal distribution with mean vector μ and covariance matrix Σ , written $\mathbf{Z} \sim \mathcal{N}(\mu, \Sigma)$, if its density is

$$f(\mathbf{z}) = \frac{1}{(2\pi)^{p/2}\sqrt{\det(\Sigma)}} \exp\left(-\frac{1}{2}(\mathbf{z} - \mu)^\top \Sigma^{-1}(\mathbf{z} - \mu)\right).$$

The quantity

$$(\mathbf{z} - \mu)^\top \Sigma^{-1}(\mathbf{z} - \mu)$$

is the squared Mahalanobis distance from \mathbf{z} to the mean μ . Contours of equal density are ellipsoids defined by constant Mahalanobis distance distance.

The multivariate normal plays a central role in quantitative genetics because sums of many small, independent genetic and environmental effects tend to generate approximately normal trait distributions.

13.14 Selection gradients and Lande's equation

Phenotypes and fitness

Let \mathbf{z} be a vector of traits for an individual, and let w be its absolute fitness (for example, number of surviving offspring).

We define *relative fitness* as

$$\tilde{w} = \frac{w}{\bar{w}},$$

where \bar{w} is the mean fitness in the population. Relative fitness has mean 1. Using relative fitness simplifies many expressions.

Selection differentials and gradients

The *selection differential* for trait i is the difference between the mean of trait i after selection and the mean before selection. For a vector of traits, the vector of selection differentials is

$$\mathbf{s} = \text{Cov}(\mathbf{z}, \tilde{w}),$$

where the covariance is taken component-wise: $s_i = \text{Cov}(z_i, \tilde{w})$.

The *directional selection gradient* β is defined as the vector of partial regression coefficients of relative fitness on traits. In matrix notation, if \mathbf{P} is the phenotypic covariance matrix of \mathbf{z} , then

$$\beta = \mathbf{P}^{-1}\mathbf{s}.$$

Equivalently, β is the vector that minimises the mean squared error in the linear approximation

$$\tilde{w} \approx \alpha + \beta^\top \mathbf{z}.$$

The elements β_i measure the strength and direction of linear selection on trait i , holding the other traits constant.

The multivariate breeder's equation equation

Let \mathbf{G} be the additive genetic covariance matrix for the traits \mathbf{z} . Under standard assumptions (additive gene action, weak selection, random mating, no environmental change across generations), the multivariate breeder's equation equation is

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\mathbf{P}^{-1}\mathbf{s},$$

where $\Delta \bar{\mathbf{z}}$ is the expected change in the mean trait vector from one generation to the next.

Substituting $\mathbf{s} = \mathbf{P}\beta$ gives

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\mathbf{P}^{-1}\mathbf{P}\beta = \mathbf{G}\beta.$$

This is *Lande's equation*.

Key Idea

Lande's equation

$$\Delta \bar{\mathbf{z}} = \mathbf{G}\beta$$

states that the expected evolutionary response vector is obtained by multiplying the directional selection gradient β by the additive genetic covariance matrix \mathbf{G} . The pattern and amount of genetic variance (encoded in \mathbf{G}) filter and redirect the effect of selection.

In the main chapters we interpret this equation geometrically: \mathbf{f} is a direction in trait space describing how fitness changes; \mathbf{G} rescales and rotates this direction according to the available additive variance, yielding the response vector $\Delta \bar{\mathbf{z}}$.

Quadratic selection

When selection is not purely directional, we can expand relative fitness to second order in the traits:

$$\tilde{w} \approx \alpha + \beta^\top \mathbf{z} + \frac{1}{2} \mathbf{z}^\top \gamma \mathbf{z},$$

where γ is a symmetric matrix of quadratic selection gradients. The diagonal entries of γ describe stabilising or disruptive selection on individual traits, and the off-diagonal entries describe correlational selection between traits.

The matrix γ defines the local curvature of the fitness surface near the mean. In later chapters we combine \mathbf{G} and γ to study how genetic variance interacts with this curvature to shape evolutionary trajectories and constraints.

Further reading

For more detailed treatments of these topics, see standard texts in linear algebra (for example, Strang's *Introduction to Linear Algebra*) and in quantitative genetics (for example, Walsh and Lynch's *Evolution and Selection of Quantitative Traits*). The aim of this appendix is not to replace such texts, but to provide a compact yet formal reminder of the concepts and notation used throughout this book.

Hints and Selected Solutions

This appendix provides hints and partial solutions for selected exercises. Full solutions are not given—working through the problems yourself is essential for developing intuition.

Chapter 0

Exercise 0.1. The centroid is $(\bar{L}, \bar{W}) = (4.4, 2.3)$. After adding plant F, the centroid shifts toward $(6.0, 3.5)$; compute the new mean of all six observations.

Exercise 0.4. Euclidean distance is $\sqrt{2^2 + 3^2} = \sqrt{13} \approx 3.6$ mm (if both traits are in mm). This number has no direct biological interpretation; it mixes millimetres of length with millimetres of width.

Chapter 1

Exercise 1.1. (a) $\|(3, 4)\| = 5$. (b) $\|(1, 1, 1)\| = \sqrt{3}$. (c) $\|(2, -2, 1)\| = 3$. (d) $\|(1, 0, 0, 0, 1)\| = \sqrt{2}$.

Exercise 1.3. (c) The dot product of $(1, 2)$ and $(-2, 1)$ is $1(-2) + 2(1) = 0$. A dot product of zero means the vectors are orthogonal (perpendicular).

Exercise 1.4. (a) $\text{proj}_{(1,0)}(3, 4) = (3, 0)$. This is the “shadow” of $(3, 4)$ on the x -axis.

Chapter 2

Exercise 2.1. \mathbf{A} stretches vectors by factor 2 in the x -direction while leaving the y -component unchanged. The unit circle becomes an ellipse with semi-axes 2 (horizontal) and 1 (vertical).

Exercise 2.2. \mathbf{R} rotates vectors by 90 counterclockwise. Applying \mathbf{R} twice gives $\mathbf{R}^2 = -\mathbf{I}$, which is rotation by 180.

Exercise 2.4. $\mathbf{AB} = \begin{pmatrix} 0 & 2 \\ 3 & 0 \end{pmatrix}$ and $\mathbf{BA} = \begin{pmatrix} 0 & 3 \\ 2 & 0 \end{pmatrix}$. They are not equal: matrix multiplication is not commutative.

Chapter 10

Exercise 10.2. Mean = 6. Deviations: $-4, -2, 0, 2, 4$. Squared deviations: $16, 4, 0, 4, 16$. Variance = $(16 + 4 + 0 + 4 + 16)/5 = 8$ (population variance) or $40/4 = 10$ (sample variance with $n - 1$).

Exercise 10.4. Sum of squared distances: at $c = 2$: 12; at $c = 3$: 8; at $c = 4$: 12. The minimum is at the mean, $c = 3$.

Exercise 10.5. In cm: $d = \sqrt{(12 - 10)^2 + (2.5 - 2)^2} = \sqrt{4.25} \approx 2.06$. In mm for wing: $d = \sqrt{(120 - 100)^2 + (2.5 - 2)^2} = \sqrt{400.25} \approx 20.0$. The wing difference now dominates because it's measured in larger numbers.

Chapter 11

Exercise 11.1. With height in metres: $d(A, B) \approx 2.00$, $d(A, C) \approx 20.0$. With height in cm: $d(A, B) \approx 5.4$, $d(A, C) \approx 22.4$. Converting to cm increases the height contribution, but since weight differences dominate anyway, $d(A, C)$ changes less proportionally.

Exercise 11.4. Euclidean distance is appropriate when: (1) all traits are measured in the same units with similar scales, and (2) traits are uncorrelated. Example where it works: measurements all in mm on a single structure. Example where it fails: mass (kg) and length (mm) of animals.

Chapter 12

Exercise 12.1. $\bar{X} = 4$, $\bar{Y} = 6$. $\text{Var}(X) = 2.5$, $\text{Var}(Y) = 2.5$. $\text{Cov}(X, Y) = 2.5$. So $\mathbf{S} = \begin{pmatrix} 2.5 & 2.5 \\ 2.5 & 2.5 \end{pmatrix}$. (Note: this matrix is singular—the traits are perfectly correlated.)

Exercise 12.3. $\mathbf{S}^{-1} = \frac{1}{16} \begin{pmatrix} 5 & -2 \\ -2 & 4 \end{pmatrix}$. $D^2 = (2, 1)\mathbf{S}^{-1}(2, 1)^\top = \frac{1}{16}(20 - 8 - 4 + 4) = 0.75$. $D = \sqrt{0.75} \approx 0.87$. Euclidean distance is $\sqrt{5} \approx 2.24$.

Exercise 12.5. For a bivariate normal, about 39% of observations lie within the $D = 1$ ellipse (not 68%). The univariate “68% within 1 SD” rule does not generalise directly because probability spreads across two dimensions.

Chapter 32

Exercise 32.1. Eigenvalues: $\lambda_1 = 8, \lambda_2 = 2$. Eigenvectors: $\mathbf{v}_1 = (1, 1)/\sqrt{2}, \mathbf{v}_2 = (1, -1)/\sqrt{2}$. PC1 explains $8/10 = 80\%$ of variance. PC1 is “both traits together” (size); PC2 is “traits in opposition” (shape/contrast).

Exercise 32.3. Covariance PCA: PC1 will be dominated by Trait A (variance 100). After standardising, all traits contribute more equally. Use covariance PCA when traits are in the same units and you want variance differences to matter. Use correlation PCA when traits are on different scales or you want to weight them equally.

Exercise 32.6. Total variance = 10. PC1 explains 52%, PC2 explains 21%, PC3 explains 10%. Cumulative: 52%, 73%, 83%. A common rule is to retain PCs until cumulative variance exceeds 80%, suggesting 3 PCs. The “elbow” in the scree plot also suggests 3.

Subject Index

- breeder's equation, 22, 76, 82, 87, 90, 118, 131, 132, 141
- characteristic equation, 60, 66, 68, 112, 138
- characteristic equation, 112
- constraint
- trap, 116
- correlation, 37, 39, 40, 42, 43, 50, 51, 63, 72, 82, 92, 98, 100, 102, 103, 106
- covariance matrix, 21, 23, 31–34, 42–44, 47–49, 51, 53–55, 59, 62–66, 71, 72, 80, 98–100, 105–107, 110, 116, 132, 138, 140, 141
- diagonalisation, 22, 55, 59, 61, 63, 64, 68, 98
- discriminant analysis, 54, 98, 101–104, 107, 132
- distance
- Euclidean, 7, 11, 12, 23, 28, 32–34, 36–44, 47, 48, 50–53, 55, 72, 73, 78, 131
 - Mahalanobis, 21, 34, 39, 41–44, 47–55, 59, 72, 73, 80, 92, 102, 130–132, 140
- eigendecomposition
- of G matrix, 112
- eigenvalue, 20, 21, 32, 48, 55, 59–68, 73–75, 77, 78, 80–83, 85–87, 91, 94–102, 104–108, 113–116, 120, 122–126, 131, 132, 137, 138
- eigenvector, 11, 20, 21, 32, 59–68, 74, 75, 77, 78, 81, 82, 87, 91, 92, 94–102, 104, 105, 107, 108, 113, 116–118, 120, 126, 128, 130, 131, 137, 138
- evolvability, 82, 83, 85, 87
- G matrix, 110
- \mathbf{G}^* matrix, 113
- G– γ alignment, 118
- γ matrix, 116
- \mathbf{g}_{\max} , 112
- heritability, 26, 68, 71–78, 85, 87, 104, 108, 110, 112–117, 120–126, 128–130
- directional, 112, 122
- MANOVA, 97, 98, 100, 101, 107, 131, 132
- matrix
- positive definite, 47, 52, 67, 114, 126, 131, 138
 - positive semi-definite, 67, 80
- mutation-selection balance, 95, 118, 119
- P matrix, 110
- P-sphere, 53, 68, 71–76, 78, 117, 121, 122, 129
- PCA, 55, 65, 66, 97–100, 103, 104, 106–108, 131, 132
- positive definiteness, 114
- quadratic form, 21, 23, 47, 65, 73, 74, 92, 112, 122, 132
- R code, 119
- selection
- correlational, 117
 - stabilising, 117
- selection gradient, 117
- whitening transformation, 53, 68, 71, 73, 75–78, 85, 87, 114, 116, 120, 130
- whitening transformation, 114
- worked example
- four-trait G-P comparison, 114
 - selection analysis, 116
 - two-trait G matrix, 110