Statistical Theory for Linear Models

Longhai Li

2025-12-25

Preface

Key Features

This text adopts a geometric approach to the statistical theory of linear models, aiming to provide a deeper understanding than standard algebraic treatments. Key features include:

- **Projection Perspective:** We prioritize the geometric interpretation of least squares, viewing estimation as a projection of the response vector onto a model subspace. This visual framework unifies diverse topics—from simple regression to complex ANOVA designs—under a single theoretical umbrella. We emphasize the geometric perspective not merely for intuition, but as the most robust framework for mastering linear models. This approach offers three distinct advantages:
  - **Statistical Clarity:** Geometry provides the most natural path to understanding the sampling distributions of estimators. By viewing estimation as an orthogonal projection, the decomposition of sums of squares into independent components becomes visually obvious, demystifying how degrees of freedom relate to subspace dimensions rather than abstract algebraic constants.
  - **Computational Stability:** A geometric understanding is essential for implementing efficient and numerically stable algorithms. While the algebraic "Normal Equations" ($(X'X)^{-1}X'y$) are theoretically valid, they are often computationally hazardous. The geometric approach leads directly to superior methods—such as QR and Singular Value Decompositions—that are the backbone of modern statistical software.
  - **Generalizability:** The principles of projection and orthogonality extend far beyond the Gaussian linear model. These geometric insights provide the foundational intuition needed for tackling non-Gaussian optimization problems, including Generalized Linear Models (GLMs) and convex optimization, where solutions can often be viewed as projections onto convex sets.
- **Interactive Visualizations:** Abstract concepts are brought to life through interactive 3D plots. Readers can rotate and inspect vector spaces, residual planes, and projection geometries to build a tangible intuition for high-dimensional operations.
- **Computational Integration:** Theory is seamlessly integrated with practice. The text provides implementation examples using R (and Python), demonstrating how theoretical matrix equations translate directly into computational code.
- **Rigorous Foundations:** While visually driven, the text maintains mathematical rigor, covering essential topics such as spectral theory, the generalized inverseand the multivariate normal distribution to ensure a solid theoretical grounding.

Overview

This course is a rigorous examination of the general linear models using vector space theory, in particular the approach of regarding least square as projection. The topics includes: vector space; projection; matrix algebra; generalized inverses; quadratic forms; theory for point estimation; theory for hypothesis test; theory for non-full-rank models.

Audience

This book is designed for graduate students and advanced undergraduate students in statistics, data science, and related quantitative fields. It serves as a bridge between applied regression analysis and the theoretical foundations of linear models. Researchers and practitioners seeking a deeper geometric and algebraic understanding of the statistical methods they use daily will also find this text valuable.

Prerequisites

To get the most out of this book, readers should have a comfortable grasp of the following topics:

Linear Algebra: An elementary understanding of matrix operations is essential. You should be familiar with matrix multiplication, determinants, inversion, and the basic concepts of vector spaces (such as linear independence, basis vectors, and subspaces). While we review key spectral theory concepts (like eigenvalues and the singular value decomposition) in the early chapters, prior exposure to these ideas is helpful.

Probability and Statistics: A standard introductory course in probability and mathematical statistics is required. Readers should be familiar with random variables, expectation, variance, covariance, common probability distributions (especially the Normal distribution), and fundamental concepts of hypothesis testing and estimation.

1 Introduction to Linear Models

1.1 Multiple Linear Regression

Suppose we have observations on Y and X_j. The data can be represented in matrix form.

$$\underset{n\times 1}{y} = \underset{n\times p}{X}\beta + \underset{n\times 1}{\epsilon}$$

where the error terms are distributed as:

$$\epsilon \sim N_n(0, \sigma^2 I_n),$$

in which I_n is the identity matrix:

$$I_n = \begin{pmatrix} 1 & 0 & ... & 0 \\ 0 & 1 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & 1 \end{pmatrix}$$

The scalar equation for a single observation is:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \cdots + \beta_p X_{ip} + \epsilon_i$$

1.1.1 Examples

1.1.1.1 Polynomial Regression

Polynomial regression fits a curved line to the data points but remains linear in the parameters (β).

The model equation is:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_{p-1} x_i^{p-1}$$

1.1.1.2 Design Matrix Construction

The design matrix X is constructed by taking powers of the input variable.

$$y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_1^2 & ... & x_1^{p-1} \\ 1 & x_2 & x_2^2 & ... & x_2^{p-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & ... & x_n^{p-1} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

1.1.1.3 One-Way ANOVA

ANOVA can be expressed as a linear model using categorical predictors (dummy variables).

Suppose we have 3 groups (G_1, G_2, G_3) with observations:

$$Y_{ij} = \mu_i + \epsilon_{ij}, \quad \epsilon_{ij} \sim N(0, \sigma^2)$$

$$\begin{array}{ccc} G_1 & G_2 & G_3 \\ \boxed{\begin{matrix} Y_{11} \\ Y_{12} \end{matrix}} & \boxed{\begin{matrix} Y_{21} \\ Y_{22} \end{matrix}} & \boxed{\begin{matrix} Y_{31} \\ Y_{32} \end{matrix}} \end{array}$$

We construct the matrix X to select the group mean (μ) corresponding to the observation:

$$\underset{6\times 1}{y} = \underset{6\times 3}{X} \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} + \epsilon$$

$$\begin{bmatrix} Y_{11} \\ Y_{12} \\ Y_{21} \\ Y_{22} \\ Y_{31} \\ Y_{32} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} + \epsilon$$

1.1.1.4 Analysis of Covariance (ANCOVA)

ANCOVA combines continuous variables and categorical (dummy) variables in the same design matrix.

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} X_{1,cont} & 1 & 0 \\ X_{2,cont} & 1 & 0 \\ \vdots & 0 & 1 \\ X_{n,cont} & 0 & 1 \end{bmatrix} \beta + \epsilon$$

1.2 Least Squares Estimation

For the general linear model $y = X\beta + \epsilon$, the Least Squares estimator is:

$$\hat{\beta} = (X'X)^{-1}X'y$$

The predicted values ($\hat{y}$) are obtained via the Projection Matrix (Hat Matrix) P_X:

$$\hat{y} = X\hat{\beta} = X(X'X)^{-1}X'y = P_X y$$

The residuals and Sum of Squared Errors are:

$$\hat{e} = y - \hat{y}$$
$$SSE = ||\hat{e}||^2$$

The coefficient of determination is:

$$R^2 = \frac{SST - SSE}{SST}$$

where $SST = \sum(y_i - \bar{y})^2$.

1.3 Geometric Interpretation: Simplified View

We align the coordinate system to the models for clarity:

1. **Reduced Model (M_0)**: Represented by the **X-axis** (labeled j_3).
   - $\hat{y}_0$ is the projection of y onto this axis.
2. **Full Model (M_1)**: Represented by the **XY-plane** (the floor).
   - $\hat{y}_1$ is the projection of y onto this plane ($z = 0$).
3. **Observed Data (y)**: A point in 3D space.

The "improvement" due to adding predictors is the distance between $\hat{y}_0$ and $\hat{y}_1$.

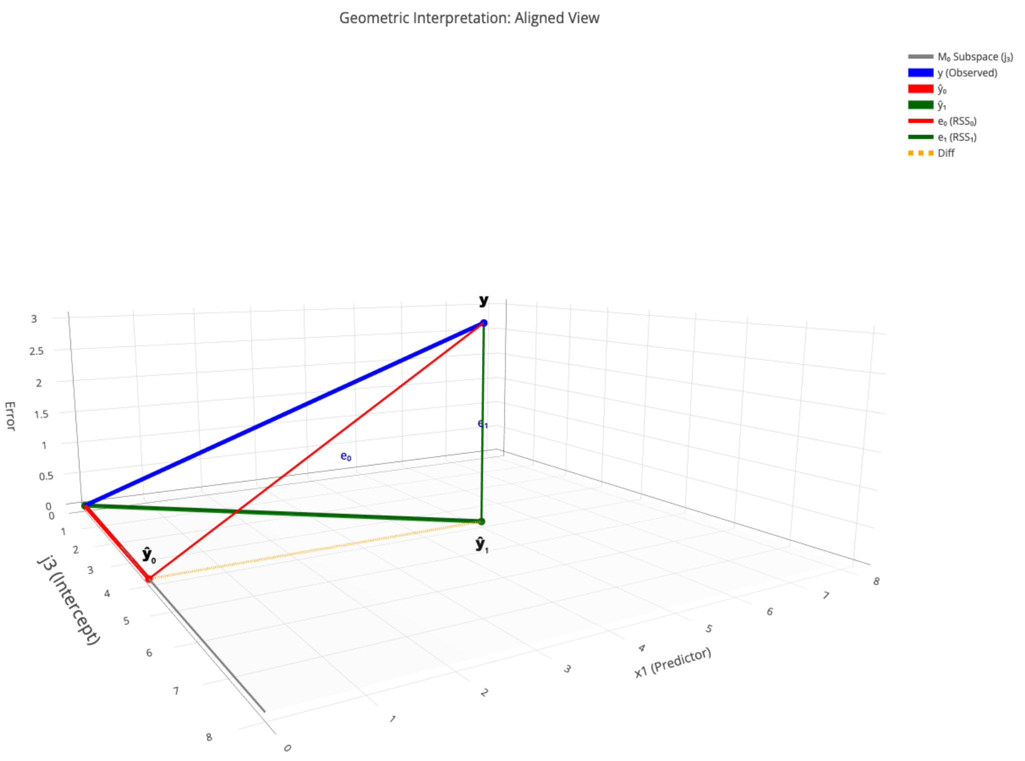

Figure 1.1: Geometric Interpretation: Projection onto Axis (M0) vs Plane (M1)

2 Projection in Vector Space

2.1 Vector and Projection onto a Line

Vectors and Operations

The concept of a vector is fundamental to linear algebra and linear models. We begin by formally defining what a vector is in the context of Euclidean space.

Definition 2.1 (Vector)**.** A **vector** x is defined as a point in n-dimensional space ($\mathbb{R}^n$). It is typically represented as a column vector containing n real-valued components:

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Vectors are not just static points; they can be combined and manipulated. The two most basic geometric operations are addition and subtraction.

Vector Arithmetic: Vectors can be manipulated geometrically:

Definition 2.2 (Vector Addition)**.** The sum of two vectors x and y creates a new vector. The operation is performed component-wise, adding corresponding elements from each vector. Geometrically, this follows the "parallelogram rule" or the "head-to-tail" method, where you place the tail of y at the head of x.

$$x + y = \begin{pmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{pmatrix}$$

Definition 2.3 (Vector Subtraction)**.** The difference $d = y - x$ is the vector that "closes the triangle" formed by x and y. It represents the displacement vector that connects the tip of x to the tip of y, such that $x + d = y$.

Scalar Multiplication and Length

In addition to combining vectors with each other, we can modify a single vector using a real number, known as a scalar.

Definition 2.4 (Scalar Multiplication)**.** Multiplying a vector by a scalar c scales its magnitude (length) without changing its line of direction. If c is positive, the direction remains the same; if c is negative, the direction is reversed.

$$cx = \begin{pmatrix} cx_1 \\ \vdots \\ cx_n \end{pmatrix}$$

We often need to quantify the "size" of a vector. This is done using the concept of length, or norm.

Definition 2.5 (Euclidean Distance (Length))**.** The length (or norm) of a vector $x = (x_1, \dots, x_n)^T$ corresponds to the straight-line distance from the origin to the point defined by x. It is defined as the square root of the sum of squared components:

$$||x||^2 = \sum_{i=1}^{n} x_i^2$$

$$||x|| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

Angle and Inner Product

To understand the relationship between two vectors x and y beyond just their lengths, we must look at the angle between them. Consider the triangle formed by the vectors x, y, and their difference $y - x$. By applying the classic **Law of Cosines** to this triangle, we can relate the geometric angle to the vector lengths.

Theorem 2.1 (Law of Cosines)**.** *For a triangle with sides a, b, c and angle θ opposite to side c:*

$$c^2 = a^2 + b^2 - 2ab\cos\theta$$

Translating this geometric theorem into vector notation where the side lengths correspond to the norms of the vectors, we get:

$$||y - x||^2 = ||x||^2 + ||y||^2 - 2||x|| \cdot ||y|| \cos\theta$$

This equation provides a critical link between the geometric angle θ and the algebraic norms of the vectors.

Derivation of Inner Product

We can express the squared distance term $||y - x||^2$ purely algebraically by expanding the components:

$$||y - x||^2 = \sum_{i=1}^{n}(x_i - y_i)^2$$

$$= \sum_{i=1}^{n}(x_i^2 + y_i^2 - 2x_i y_i)$$

$$= ||x||^2 + ||y||^2 - 2\sum_{i=1}^{n} x_i y_i$$

By comparing this expanded form with the result from the Law of Cosines derived previously, we can identify a corresponding interaction term. This term is so important that we give it a special name: the **Inner Product** (or dot product).

Definition 2.6 (Inner Product)**.** The inner product of two vectors x and y is defined as the sum of the products of their corresponding components:

$$x'y = \sum_{i=1}^{n} x_i y_i = \langle x, y \rangle$$

Thus, equating the geometric and algebraic forms yields the fundamental relationship:

$$x'y = ||x|| \cdot ||y|| \cos\theta$$

Coordinate (Scalar) Projection

The inner product allows us to calculate projections, which quantify how much of one vector "lies along" another. If we rearrange the cosine formula derived above, we can isolate the term that represents the length of the "shadow" cast by vector y onto vector x.

The length of this projection is given by:

$$||y|| \cos\theta = \frac{x'y}{||x||}$$

This expression can be interpreted as the inner product of y with the normalized (unit) vector in the direction of x:

$$\text{Scalar Projection} = \left\langle \frac{x}{||x||}, y \right\rangle$$

Vector Projection Formula

The scalar projection only gives us a magnitude (a number). To define the projection as a vector in the same space, we need to multiply this scalar magnitude by the direction of the vector we are projecting onto.

Definition 2.7 (Vector Projection)**.** The projection of vector y onto vector x, denoted $\hat{y}$, is calculated as:

$$\text{Projection Vector} = (\text{Length}) \cdot (\text{Direction})$$

$$\hat{y} = \left(\frac{x'y}{||x||} \right) \cdot \frac{x}{||x||}$$

This is often written compactly by combining the denominators:

$$\hat{y} = \frac{x'y}{||x||^2} x$$

Perpendicularity (Orthogonality)

A special case of the angle between vectors arises when $\theta = 90^\circ$. This geometric concept of perpendicularity is central to the theory of projections and least squares.

Definition 2.8 (Perpendicularity)**.** Two vectors are defined as **perpendicular** (or orthogonal) if the angle between them is 90° ($\pi/2$).

Since $\cos(90^\circ) = 0$, the condition for orthogonality simplifies to the inner product being zero:

$$x'y = 0 \iff x \perp y$$

Example 2.1 (Orthogonal Vectors)**.** Consider two vectors in $\mathbb{R}^2$: $x = (1, 1)'$ and $y = (1, -1)'$.

$$x'y = 1(1) + 1(-1) = 1 - 1 = 0$$

Since their inner product is zero, these vectors are orthogonal to each other.

Projection onto a Line (Subspace)

We can generalize the concept of projecting onto a single vector to projecting onto the entire line (a 1-dimensional subspace) defined by that vector.

Definition 2.9 (Line Spanned by a Vector)**.** The line space $L(x)$, or the space spanned by a vector x, is defined as the set of all scalar multiples of x:

$$L(x) = \{cx \mid c \in \mathbb{R}\}$$

The projection of y onto $L(x)$, denoted $\hat{y}$, is defined by the geometric property that it is the closest point on the line to y. This implies that the error vector (or residual) must be perpendicular to the line itself.

Definition 2.10 (Projection onto a Line)**.** A vector $\hat{y}$ is the projection of y onto the line $L(x)$ if:

1. $\hat{y}$ lies on the line $L(x)$ (i.e., $\hat{y} = cx$ for some scalar c).
2. The residual vector $(y - \hat{y})$ is perpendicular to the direction vector x.

Derivation: To find the value of the scalar c, we apply the orthogonality condition:

$$(y - \hat{y}) \perp x \implies x'(y - cx) = 0$$

Expanding this inner product gives:

$$x'y - c(x'x) = 0$$

Solving for c, we obtain:

$$c = \frac{x'y}{||x||^2}$$

This confirms the formula derived previously using the inner product geometry. It shows that the least squares principle (shortest distance) leads to the same result as the geometric projection.

Alternative Forms of the Projection Formula

We can express the projection vector $\hat{y}$ in several equivalent ways to highlight different geometric interpretations.

Definition 2.11 (Forms of Projection)**.** The projection of y onto the vector x is given by:

$$\hat{y} = \frac{x'y}{||x||^2}x = \left\langle y, \frac{x}{||x||} \right\rangle \frac{x}{||x||}$$

This second form separates the components into:

$$\text{Projection} = (\text{Scalar Projection}) \times (\text{Unit Direction})$$

Projection Matrix (P_x)

In linear models, it is often more convenient to view projection as a linear transformation applied to the vector y. This allows us to define a **Projection Matrix**.

We can rewrite the formula for $\hat{y}$ by factoring out y:

$$\hat{y} = \text{proj}(y|x) = x\frac{x'y}{||x||^2} = \frac{xx'}{||x||^2}y$$

This leads to the definition of the projection matrix P_x.

Definition 2.12 (Projection Matrix onto a Single Vector)**.** The matrix P_x that projects any vector y onto the line spanned by x is defined as:

$$P_x = \frac{xx'}{||x||^2}$$

Using this matrix, the projection is simply:

$$\hat{y} = P_x y$$

If $x \in \mathbb{R}^p$, then P_x is a $p \times p$ symmetric matrix.

Example: Projection in $\mathbb{R}^2$

Let's apply these concepts to a concrete example.

Example 2.2 (Numerical Projection)**.** Let $y = (1, 3)'$ and $x = (1, 1)'$. We want to find the projection of y onto x.

Method 1: Using the Vector Formula First, calculate the inner products:

$$x'y = 1(1) + 1(3) = 4$$

$$||x||^2 = 1^2 + 1^2 = 2$$

Now, apply the formula:

$$\hat{y} = \frac{4}{2}\begin{pmatrix}1\\1\end{pmatrix} = 2\begin{pmatrix}1\\1\end{pmatrix} = \begin{pmatrix}2\\2\end{pmatrix}$$

Method 2: Using the Projection Matrix Construct the matrix P_x:

$$P_x = \frac{1}{2}\begin{pmatrix}1\\1\end{pmatrix}\begin{pmatrix}1 & 1\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1 & 1\\1 & 1\end{pmatrix} = \begin{pmatrix}0.5 & 0.5\\0.5 & 0.5\end{pmatrix}$$

Multiply by y:

$$\hat{y} = P_x y = \begin{pmatrix}0.5 & 0.5\\0.5 & 0.5\end{pmatrix}\begin{pmatrix}1\\3\end{pmatrix} = \begin{pmatrix}0.5(1) + 0.5(3)\\0.5(1) + 0.5(3)\end{pmatrix} = \begin{pmatrix}2\\2\end{pmatrix}$$

Example: Projection onto the Mean Vector

A very common operation in statistics is calculating the sample mean. This can be viewed geometrically as a projection onto a specific vector.

Example 2.3 (Projection onto the "One" Vector)**.** Let $y = (y_1, \dots, y_n)'$ be a data vector. Let $j_n = (1, 1, \dots, 1)'$ be a vector of all ones.

The projection of y onto j_n is:

$$\text{proj}(y|j_n) = \frac{j_n' y}{||j_n||^2} j_n$$

Calculating the components:

$$j_n' y = \sum_{i=1}^{n} y_i \quad \text{(Sum of observations)}$$

$$||j_n||^2 = \sum_{i=1}^{n} 1^2 = n$$

Substituting these back:

$$\hat{y} = \frac{\sum y_i}{n} j_n = \bar{y} j_n = \begin{pmatrix} \bar{y} \\ \vdots \\ \bar{y} \end{pmatrix}$$

Thus, replacing a data vector with its mean vector is geometrically equivalent to projecting the data onto the line spanned by the vector of ones.

Pythagorean Theorem

The Pythagorean theorem generalizes from simple geometry to vector spaces using the concept of orthogonality defined by the inner product.

Theorem 2.2 (Pythagorean Theorem)**.** *If two vectors x and y are orthogonal (i.e., $x \perp y$ or $x'y = 0$), then the squared length of their sum is equal to the sum of their squared lengths:*

$$||x + y||^2 = ||x||^2 + ||y||^2$$

Proof: We expand the squared norm using the inner product:

$$\begin{aligned} ||x + y||^2 &= (x + y)'(x + y) \\ &= x'x + x'y + y'x + y'y \\ &= ||x||^2 + 2x'y + ||y||^2 \end{aligned}$$

Since $x \perp y$, the inner product $x'y = 0$. Thus, the term $2x'y$ vanishes, leaving:

$$||x + y||^2 = ||x||^2 + ||y||^2$$

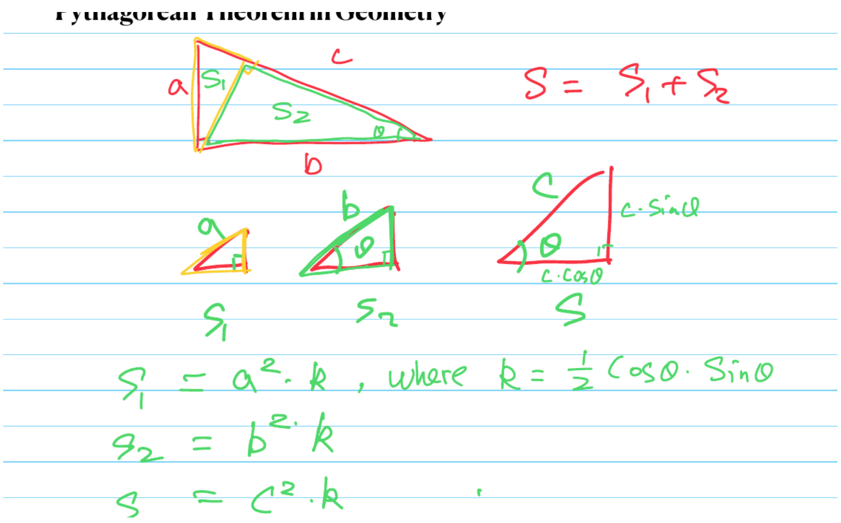
Pythagorean Theorem in Geometry
a
S1
c
S2
b
θ
S = S1 + S2
a
b
c
θ
c·Sinθ
c·cosθ
S1
S2
S
S1 = a²·k , where k = ½ cosθ·Sinθ
S2 = b²·k
S = c²·k

Figure 2.1: Pythagorean Theorem in Vector Space

Least Square Property

One of the most important properties of the orthogonal projection is that it minimizes the distance between the vector y and the subspace (or line) onto which it is projected.

Theorem 2.3 (Least Square Property)**.** *Let $\hat{y}$ be the projection of y onto the line $L(x)$. For any other vector y^* on the line $L(x)$, the distance from y to y^* is always greater than or equal to the distance from y to $\hat{y}$.*

$$||y - y^*|| \geq ||y - \hat{y}||$$

Proof: Since both $\hat{y}$ and y^* lie on the line $L(x)$, their difference $(\hat{y} - y^*)$ also lies on $L(x)$. From the definition of projection, the residual $(y - \hat{y})$ is orthogonal to the line $L(x)$. Therefore:

$$(y - \hat{y}) \perp (\hat{y} - y^*)$$

We can write the vector $(y - y^*)$ as:

$$y - y^* = (y - \hat{y}) + (\hat{y} - y^*)$$

Applying the Pythagorean Theorem:

$$||y - y^*||^2 = ||y - \hat{y}||^2 + ||\hat{y} - y^*||^2$$

Since $||\hat{y} - y^*||^2 \geq 0$, it follows that:

$$||y - y^*||^2 \geq ||y - \hat{y}||^2$$

2.2 General Vector Space

We now generalize our discussion from lines to broader spaces.

Definition 2.13 (Vector Space)**.** A set $V \subseteq \mathbb{R}^n$ is called a **Vector Space** if it is closed under vector addition and scalar multiplication:

1. **Closed under Addition:** If $x_1 \in V$ and $x_2 \in V$, then $x_1 + x_2 \in V$.
2. **Closed under Scalar Multiplication:** If $x \in V$, then $cx \in V$ for any scalar $c \in \mathbb{R}$.

It follows that the zero vector 0 must belong to any subspace (by choosing $c = 0$).

Spanned Vector Space

The most common way to construct a vector space in linear models is by spanning it with a set of vectors.

Definition 2.14 (Spanned Vector Space)**.** Let $x_1, \ldots, x_p$ be a set of vectors in $\mathbb{R}^n$. The space spanned by these vectors, denoted $L(x_1, \ldots, x_p)$, is the set of all possible linear combinations of them:

$$L(x_1, \ldots, x_p) = \{r \mid r = c_1 x_1 + \cdots + c_p x_p, \text{ for } c_i \in \mathbb{R}\}$$

Column Space and Row Space

When vectors are arranged into a matrix, we define specific spaces based on their columns and rows.

Definition 2.15 (Column Space)**.** For a matrix $X = (x_1, \ldots, x_p)$, the **Column Space**, denoted $Col(X)$, is the vector space spanned by its columns:

$$Col(X) = L(x_1, \ldots, x_p)$$

Definition 2.16 (Row Space)**.** The **Row Space**, denoted $Row(X)$, is the vector space spanned by the rows of the matrix X.

Linear Independence and Rank

Not all vectors in a spanning set contribute new dimensions to the space. This concept is captured by linear independence.

Definition 2.17 (Linear Independence)**.** A set of vectors $x_1, \ldots, x_p$ is said to be **Linearly Independent** if the only solution to the linear combination equation equal to zero is the trivial solution:

$$\sum_{i=1}^{p} c_i x_i = 0 \implies c_1 = c_2 = \cdots = c_p = 0$$

If there exist non-zero c_i's such that sum is zero, the vectors are **Linearly Dependent**.

2.3 Rank of Matrices and Dim of Vector Space

Definition 2.18 (Rank)**.** The **Rank** of a matrix X, denoted Rank(X), is the maximum number of linearly independent columns in X. This is equivalent to the dimension of the column space:

$$\text{Rank}(X) = \text{Dim}(Col(X))$$

2.3.0.1 Properties of Rank

There are several fundamental properties regarding the rank of a matrix.

Theorem 2.4 (Properties of Rank)**.**

1. ***Row Rank equals Column Rank:*** *The dimension of the column space is equal to the dimension of the row space.*

$$\textit{Dim}(Col(X)) = \textit{Dim}(Row(X)) \implies \textit{Rank}(X) = \textit{Rank}(X')$$

2. ***Bounds:*** *For an $n \times p$ matrix X:*

$$\textit{Rank}(X) \leq \min(n, p)$$

Proof. Let X be an $n \times p$ matrix. Let r be the row rank of X. This means the dimension of the row space is r. Let $u_1, \dots, u_r$ be a basis for the row space of X (these are row vectors). Since every row of X is in the row space, each row $x_{i.}$ can be written as a linear combination of the basis vectors:

$$x_{i.} = c_{i1}u_1 + c_{i2}u_2 + \dots + c_{ir}u_r \quad \text{for } i = 1, \dots, n$$

We can write this in matrix notation as:

$$X = CU$$

where C is an $n \times r$ matrix of coefficients c_{ij}, and U is an $r \times p$ matrix with rows $u_1, \dots, u_r$.

Now consider the columns of X. Since $X = CU$, the columns of X are linear combinations of the columns of C. Let $c^{(j)}$ be the j-th column of C. The columns of X lie in the space spanned by $\{c^{(1)}, \dots, c^{(r)}\}$. Thus, the column space of X, $Col(X)$, is a subspace of the column space of C.

$$\text{Dim}(Col(X)) \leq \text{Dim}(Col(C)) \leq r$$

The dimension of the column space of C is at most r (since C has only r columns). Therefore, Column Rank $\leq$ Row Rank.

Applying the same logic to X', we get Row Rank $\leq$ Column Rank. Combining these inequalities gives: **Row Rank = Column Rank**. □

Example: 2x3 Matrix

Consider the following 2×3 matrix:

$$X = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Row Rank: The rows are $r_1 = (1, 0, 1)$ and $r_2 = (0, 1, 1)$. Neither is a multiple of the other, so they are linearly independent.

$$\text{Row Rank} = 2$$

Column Rank: The columns are $c_1 = \binom{1}{0}$, $c_2 = \binom{0}{1}$, and $c_3 = \binom{1}{1}$. Notice that $c_3 = c_1 + c_2$. The third column is dependent on the first two. However, c_1 and c_2 are independent (standard basis vectors).

$$\text{Column Rank} = 2$$

Thus, Rank(Row) = Rank(Col) = 2.

Orthogonality to a Subspace

We can extend the concept of orthogonality from single vectors to entire subspaces.

Definition 2.19 (Orthogonality to a Subspace)**.** A vector y is orthogonal to a subspace V (denoted $y \perp V$) if y is orthogonal to **every** vector x in V.

$$y \perp V \iff y'x = 0 \quad \forall x \in V$$

Definition 2.20 (Orthogonal Complement)**.** The set of all vectors that are orthogonal to a subspace V is called the **Orthogonal Complement** of V, denoted $V^{\perp}$.

$$V^{\perp} = \{y \in \mathbb{R}^n \mid y \perp V\}$$

Kernel (Null Space) and Image

For a matrix transformation defined by X, we define two key spaces: the Image (Column Space) and the Kernel (Null Space).

Definition 2.21 (Image and Kernel)**.**

1. **Image (Column Space):** The set of all possible outputs.

$$\text{Im}(X) = Col(X) = \{X\beta \mid \beta \in \mathbb{R}^p\}$$

2. **Kernel (Null Space):** The set of all inputs mapped to the zero vector.

$$\text{Ker}(X) = \{\beta \in \mathbb{R}^p \mid X\beta = 0\}$$

Theorem 2.5 (Relationship between Kernel and Row Space)**.** *The kernel of X is the orthogonal complement of the row space of X:*

$$\mathit{Ker}(X) = [\mathit{Row}(X)]^{\perp}$$

Nullity Theorem

There is a fundamental relationship between the dimensions of these spaces.

Theorem 2.6 (Rank-Nullity Theorem)**.** *For an $n \times p$ matrix X:*

$$\mathit{Rank}(X) + \mathit{Nullity}(X) = p$$

Where $\mathit{Nullity}(X) = \mathit{Dim}(\mathit{Ker}(X))$.

-

Rank Inequalities

Understanding the bounds of the rank of matrix products is crucial for deriving properties of linear estimators.

Theorem 2.7 (Rank of a Matrix Product)**.** *Let X be an $n \times p$ matrix and Z be a $p \times k$ matrix. The rank of their product XZ is bounded by the rank of the individual matrices:*

$$\mathit{Rank}(XZ) \leq \min(\mathit{Rank}(X), \mathit{Rank}(Z))$$

Proof: The columns of XZ are linear combinations of the columns of X. Thus, the column space of XZ is a subspace of the column space of X:

$$Col(XZ) \subseteq Col(X) \implies \text{Rank}(XZ) \leq \text{Rank}(X)$$

Similarly, the rows of XZ are linear combinations of the rows of Z. Thus, the row space of XZ is a subspace of the row space of Z:

$$Row(XZ) \subseteq Row(Z) \implies \text{Rank}(XZ) \leq \text{Rank}(Z)$$

Rank and Invertible Matrices

Multiplying by an invertible (non-singular) matrix preserves the rank. This is a very useful property when manipulating linear equations.

Theorem 2.8 (Rank with Non-Singular Multiplication)**.** *Let A be an $n \times n$ invertible matrix (i.e., Rank$(A) = n$) and X be an $n \times p$ matrix. Then:*

$$\textit{Rank}(AX) = \textit{Rank}(X)$$

Similarly, if B is a $p \times p$ invertible matrix, then:

$$\textit{Rank}(XB) = \textit{Rank}(X)$$

Proof: From the previous theorem, we know Rank$(AX) \leq$ Rank(X). Since A is invertible, we can write $X = A^{-1}(AX)$. Applying the theorem again:

$$\text{Rank}(X) = \text{Rank}(A^{-1}(AX)) \leq \text{Rank}(AX)$$

Thus, Rank$(AX) =$ Rank(X).

Rank of $X'X$ and XX'

The matrix $X'X$ (the Gram matrix) appears in the normal equations for least squares ($X'X\beta = X'y$). Its properties are closely tied to X.

Theorem 2.9 (Rank of Gram Matrix)**.** *For any real matrix X, the rank of $X'X$ and XX' is the same as the rank of X itself:*

$$\textit{Rank}(X'X) = \textit{Rank}(X)$$
$$\textit{Rank}(XX') = \textit{Rank}(X)$$

Proof: We first show that the null space (kernel) of X is the same as the null space of $X'X$. If $v \in$ Ker(X), then $Xv = 0 \implies X'Xv = 0 \implies v \in$ Ker$(X'X)$. Conversely, if $v \in$ Ker$(X'X)$, then $X'Xv = 0$. Multiply by v':

$$v'X'Xv = 0 \implies (Xv)'(Xv) = 0 \implies ||Xv||^2 = 0 \implies Xv = 0$$

So Ker$(X) =$ Ker$(X'X)$. By the Rank-Nullity Theorem, since they have the same number of columns and same nullity, they must have the same rank.

Column Space of XX'

Beyond just the rank, the column spaces themselves are related.

Theorem 2.10 (Column Space Equivalence)**.** *The column space of XX' is identical to the column space of X:*

$$Col(XX') = Col(X)$$

Implication: This property ensures that for any y, the projection of y onto $Col(X)$ lies in the same space as the projection onto $Col(XX')$. This is vital for the existence of solutions in generalized least squares.

2.4 Projection via Orthonormal Basis (Q)

2.4.1 Orthonomal Basis

Before discussing projections onto general subspaces, we must formally define the coordinate system of a subspace, known as a basis.

Definition 2.22 (Basis)**.** A set of vectors $\{x_1, \ldots, x_k\}$ is a **Basis** for a vector space V if:

1. The vectors span the space: $V = L(x_1, \ldots, x_k)$.
2. The vectors are linearly independent.

The number of vectors in a basis is unique and is defined as the **Dimension** of V.

Calculations become significantly simpler if we choose a basis with special geometric properties.

Definition 2.23 (Orthonormal Basis)**.** A basis $\{q_1, \ldots, q_k\}$ is called an **Orthonormal Basis** if:

1. **Orthogonal:** Each pair of vectors is perpendicular.
$$q_i' q_j = 0 \quad \text{for } i \neq j$$
2. **Normalized:** Each vector has unit length.
$$||q_i||^2 = q_i' q_i = 1$$

Combining these, we write $q_i' q_j = \delta_{ij}$ (Kronecker delta).

We now generalize the projection problem. Instead of projecting y onto a single line, we project it onto a subspace V of dimension k.

If we have an orthonormal basis $\{q_1, \ldots, q_k\}$ for V, the projection $\hat{y}$ is simply the sum of the projections onto the individual basis vectors.

Definition 2.24 (Projection Defined with Orthonormal Basis)**.** The projection of y onto the subspace $V = L(q_1, \ldots, q_k)$ is:

$$\hat{y} = \sum_{i=1}^{k} \text{proj}(y|q_i) = \sum_{i=1}^{k} (q_i' y) q_i$$

Since the basis vectors are normalized, we do not need to divide by $||q_i||^2$.

Definition 2.25 (Definition of Projection onto a Subspace V)**.** Let V be a subspace of $\mathbb{R}^n$. For any vector $y \in \mathbb{R}^n$, there exists a **unique** vector $\hat{y} \in V$ such that the residual is orthogonal to the subspace:

$$(y - \hat{y}) \perp V$$

Equivalently:

$$\langle y - \hat{y}, v \rangle = 0 \quad \forall v \in V$$

Theorem 2.11 (Projection via Orthonormal Basis)**.** *Let $\{q_1, \dots, q_k\}$ be an orthonormal basis for the subspace $V \subseteq \mathbb{R}^n$. The vector defined by the sum of individual projections:*

$$\hat{y} = \sum_{i=1}^{k} \langle y, q_i \rangle q_i$$

is indeed the orthogonal projection of y onto V. That is, it satisfies $(y - \hat{y}) \perp V$.

Proof. To prove this, we must check two conditions:

1. $\hat{y} \in V$: This is immediate because $\hat{y}$ is a linear combination of the basis vectors $\{q_1, \dots, q_k\}$.
2. $(y - \hat{y}) \perp V$: It suffices to show that the error vector $e = y - \hat{y}$ is orthogonal to every basis vector q_j (for $j = 1, \dots, k$).

   Let's calculate the inner product $\langle y - \hat{y}, q_j \rangle$:

$$\begin{aligned} \langle y - \hat{y}, q_j \rangle &= \langle y, q_j \rangle - \langle \hat{y}, q_j \rangle \\ &= \langle y, q_j \rangle - \left\langle \sum_{i=1}^{k} \langle y, q_i \rangle q_i, q_j \right\rangle \\ &= \langle y, q_j \rangle - \sum_{i=1}^{k} \langle y, q_i \rangle \underbrace{\langle q_i, q_j \rangle}_{\delta_{ij}} \end{aligned}$$

   Since the basis is orthonormal, $\langle q_i, q_j \rangle$ is 1 if $i = j$ and 0 otherwise. Thus, the summation collapses to a single term where $i = j$:

$$\begin{aligned} \langle y - \hat{y}, q_j \rangle &= \langle y, q_j \rangle - \langle y, q_j \rangle \cdot 1 \\ &= 0 \end{aligned}$$

   Since $(y - \hat{y})$ is orthogonal to every basis vector q_j, it is orthogonal to the entire subspace V. Thus, $\hat{y}$ is the unique orthogonal projection.

□

2.4.2 Projection Matrix via Orthonomal Basis (Q)

Matrix Form with Orthonormal Basis

We can express the summation formula for $\hat{y}$ compactly using matrix notation.

Let Q be an $n \times k$ matrix whose columns are the orthonormal basis vectors $q_1, \dots, q_k$.

$$Q = \begin{pmatrix} q_1 & q_2 & \dots & q_k \end{pmatrix}$$

Properties of Q: * $Q'Q = I_k$ (Identity matrix of size $k \times k$). * QQ' is **not** necessarily I_n (unless $k = n$).

Definition 2.26 (Projection Matrix in Terms of Q)**.** The projection $\hat{y}$ can be written as:

$$\hat{y} = \begin{pmatrix} q_1 & \dots & q_k \end{pmatrix} \begin{pmatrix} q_1' y \\ \vdots \\ q_k' y \end{pmatrix} = Q(Q'y) = (QQ')y$$

Thus, the projection matrix P onto the subspace V is:

$$P = QQ'$$

Properties of Projection Matrices

We have defined the projection matrix as $P = X(X'X)^{-1}X'$ (or $P = QQ'$ for orthonormal bases). All orthogonal projection matrices share two fundamental algebraic properties.

Theorem 2.12 (Symmeticity and Idempotence)**.** *A square matrix P represents an orthogonal projection onto some subspace if and only if it satisfies:*

1. ***Idempotence:*** *$P^2 = P$ (Applying the projection twice is the same as applying it once).*
2. ***Symmetry:*** *$P' = P$.*

Proof. If $\hat{y} = Py$ is already in the subspace $Col(X)$, then projecting it again should not change it.

$$P(Py) = Py \implies P^2 y = Py \quad \forall y$$

Thus, $P^2 = P$. □

Example: ANOVA (Analysis of Variance)

One of the most common applications of projection is in Analysis of Variance (ANOVA). We can view the calculation of group means as a projection onto a subspace defined by group indicator variables.

Example 2.4 (Finding Projection for One-way ANOVA)**.** Consider a one-way ANOVA model with k groups:

$$y_{ij} = \mu_i + \epsilon_{ij}$$

where $i \in \{1, \dots, k\}$ represents the group and $j \in \{1, \dots, n_i\}$ represents the observation within the group. Let $N = \sum_{i=1}^{k} n_i$ be the total number of observations.

1. Matrix Definitions We define the data vector y and the design matrix X as follows:

- **Data Vector (y):** An $N \times 1$ vector containing all observations stacked by group:

$$y = \begin{pmatrix} y_{11} \\ \vdots \\ y_{1n_1} \\ y_{21} \\ \vdots \\ y_{kn_k} \end{pmatrix}$$

- **Design Matrix (X):** An $N \times k$ matrix constructed from k column vectors, $X = (x_1, x_2, \dots, x_k)$. Each vector x_g is an **indicator variable** (dummy variable) for group g:

$$x_g = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \quad \leftarrow \text{Entries are 1 if observation belongs to group } g$$

2. Orthogonality These column vectors $x_1, \dots, x_k$ are mutually orthogonal because no observation can belong to two groups at once. The dot product of any two distinct columns is zero:

$$\langle x_g, x_h \rangle = 0 \quad \text{for } g \neq h$$

This allows us to find the projection onto the column space of X by simply summing the projections onto each column individually.

3. Calculating Individual Projections For a specific group vector x_g, the projection is:

$$\text{proj}(y|x_g) = \frac{\langle y, x_g \rangle}{\langle x_g, x_g \rangle} x_g$$

We calculate the two scalar terms:

- **Denominator ($\langle x_g, x_g \rangle$):** The sum of squared elements of x_g. Since x_g contains n_g ones and zeros elsewhere:

$$\langle x_g, x_g \rangle = \sum \mathbb{1}_{\{i=g\}}^2 = n_g$$

- **Numerator ($\langle y, x_g \rangle$):** The dot product sums only the y values belonging to group g:

$$\langle y, x_g \rangle = \sum_{i,j} y_{ij} \cdot \mathbb{1}_{\{i=g\}} = \sum_{j=1}^{n_g} y_{gj} = y_{g.} \quad \text{(Group Total)}$$

4. The Resulting Projection Substituting these back into the formula gives the coefficient for the vector x_g:

$$\text{proj}(y|x_g) = \frac{y_{g.}}{n_g} x_g = \bar{y}_{g.} x_g$$

The total projection $\hat{y}$ is the sum over all groups:

$$\hat{y} = \sum_{g=1}^{k} \bar{y}_{g.} x_g$$

This confirms that the fitted value for any specific observation y_{ij} is simply its group mean $\bar{y}_{i.}$.

2.4.3 Gram-Schmidt Process

To use the simplified formula $P = QQ'$, we need an orthonormal basis. The Gram-Schmidt process provides a method to construct such a basis from any set of linearly independent vectors.

Gram-Schmidt Process Given linearly independent vectors $x_1, \dots, x_p$:

1. **Step 1:** Normalize the first vector.

$$q_1 = \frac{x_1}{||x_1||}$$

2. **Step 2:** Project x_2 onto q_1 and subtract it to find the orthogonal component.

$$v_2 = x_2 - (x_2' q_1) q_1$$

   Then normalize:

$$q_2 = \frac{v_2}{||v_2||}$$

3. **Step k:** Subtract the projections onto all previous q vectors.

$$v_k = x_k - \sum_{j=1}^{k-1} (x_k' q_j) q_j$$

$$q_k = \frac{v_k}{||v_k||}$$

This process leads to the **QR Decomposition** of a matrix: $X = QR$, where Q is orthogonal and R is upper triangular.

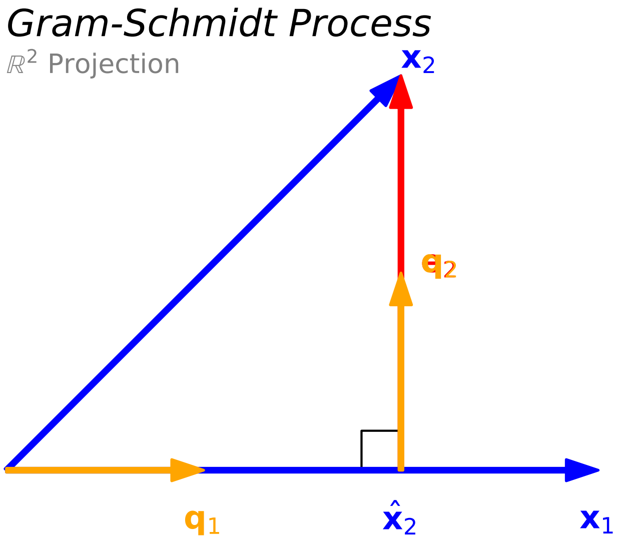

Figure 2.2: Gram-Schmidt Process: Projecting x_2 onto x_1

2.5 Hat Matrix (Projection Matrix via X)

2.5.1 Norm Equations

Let $X = (x_1, \dots, x_p)$ be an $n \times p$ matrix, where each column x_j is a predictor vector.

We want to project the target vector y onto the column space $C(X)$. This is equivalent to finding a coefficient vector $\beta \in \mathbb{R}^p$ such that the error vector (residual) is orthogonal to the entire subspace $C(X)$.

$$y - X\beta \perp C(X)$$

Since the columns of X span the subspace, the residual must be orthogonal to **every** column vector x_j individually:

$$y - X\beta \perp x_j \quad \text{for } j = 1, \dots, p$$

Writing this geometric condition as an algebraic dot product (where x_j' denotes the transpose):

$$x_j'(y - X\beta) = 0 \quad \text{for each } j$$

We can stack these p separate linear equations into a single matrix equation. Since the rows of X' are the columns of X, this becomes:

$$\begin{pmatrix} x_1' \\ \vdots \\ x_p' \end{pmatrix} (y - X\beta) = \mathbf{0} \implies X'(y - X\beta) = 0$$

Finally, we distribute the matrix transpose and rearrange terms to solve for β:

$$\begin{aligned} X'y - X'X\beta &= 0 \\ X'X\beta &= X'y \end{aligned}$$

This system is known as the **Normal Equations**.

Theorem 2.13 (Least Squares Estimator)**.** *If $X'X$ is invertible (i.e., X has full column rank), the unique solution for β is:*

$$\hat{\beta} = (X'X)^{-1}X'y$$

2.5.2 Hat Matrix

Substituting the estimator $\hat{\beta}$ back into the equation for $\hat{y}$ gives us the projection matrix.

Definition 2.27 (Hat Matrix)**.** The projection of y onto $Col(X)$ is given by:

$$\hat{y} = X\hat{\beta} = X(X'X)^{-1}X'y$$

Thus, the hat matrix P is defined as:

$$P = X(X'X)^{-1}X'$$

2.5.3 Equivalence of Hat Matrix and QQ'

If we use the QR decomposition such that $X = QR$, where the columns of Q form an orthonormal basis for $Col(X)$, the formula simplifies significantly.

Recall that for orthonormal columns, $Q'Q = I$. Substituting $X = QR$ into the general formula:

$$\begin{aligned}
P &= QR((QR)'(QR))^{-1}(QR)' \\
&= QR(R'Q'QR)^{-1}R'Q' \\
&= QR(R'\underbrace{Q'Q}_{I}R)^{-1}R'Q' \\
&= QR(R'R)^{-1}R'Q' \\
&= QRR^{-1}(R')^{-1}R'Q' \\
&= Q\underbrace{RR^{-1}}_{I}\underbrace{(R')^{-1}R'}_{I}Q' \\
&= QQ'
\end{aligned}$$

This confirms that $P = QQ'$ is consistent with the general formula $P = X(X'X)^{-1}X'$.

2.5.4 Properties of Hat Matrix

We revisit the properties of projection matrices in this general context.

Theorem 2.14 (Properties of Hat Matrix)**.** *The matrix $P = X(X'X)^{-1}X'$ satisfies:*

1. ***Symmetric:*** $P' = P$
2. ***Idempotent:*** $P^2 = P$
3. ***Trace:*** *The trace of a projection matrix equals the dimension of the subspace it projects onto.*

$$tr(P) = tr(X(X'X)^{-1}X') = tr((X'X)^{-1}X'X) = tr(I_p) = p$$

2.5.5 Projection onto Complement

Definition 2.28 (Residual Maker Matrix M)**.**

$$M = I - X(X^\top X)^{-1}X^\top$$

This matrix projects y onto the null space of $X^\top$ (i.e., Col$(X)^\perp$).

2.6 General Projection Matrices onto Nested Subspaces

2.6.1 Nested Models and Subspaces

In hypothesis testing (like comparing a null model to an alternative model), we often deal with nested subspaces.

Definition 2.29 (Nested Models)**.** Consider two models: 1. **Reduced Model (M_0):** $y \in Col(X_0)$ 2. **Full Model (M_1):** $y \in Col(X_1)$

We say the models are nested if the column space of the reduced model is contained entirely within the column space of the full model:

$$Col(X_0) \subseteq Col(X_1)$$

Usually, X_1 is constructed by adding columns to X_0: $X_1 = [X_0, X_{new}]$.

2.6.2 Projections onto Nested Subspaces

Let P_0 be the projection matrix onto $Col(X_0)$ and P_1 be the projection matrix onto $Col(X_1)$. Since $Col(X_0) \subseteq Col(X_1)$, we have important relationships between these matrices.

Theorem 2.15 (Composition of Projections)**.** *If $Col(P_0) \subseteq Col(P_1)$, then:*

1. $P_1P_0 = P_0$ *(Projecting onto the small space, then the large space, keeps you in the small space).*
2. $P_0P_1 = P_0$ *(Projecting onto the large space, then the small space, is the same as just projecting onto the small space).*

Difference of Projections

The difference between the two projection matrices, $P_1 - P_0$, is itself a projection matrix.

Theorem 2.16 (Difference Projection)**.** *The matrix $P_\Delta = P_1 - P_0$ is an orthogonal projection matrix onto the subspace $Col(X_1) \cap Col(X_0)^\perp$. This subspace represents the "extra" information in the full model that is orthogonal to the reduced model.*

Properties:

1. ***Symmetric:*** $(P_1 - P_0)' = P_1 - P_0$.
2. ***Idempotent:*** $(P_1 - P_0)(P_1 - P_0) = P_1 - P_0P_1 - P_1P_0 + P_0 = P_1 - P_0 - P_0 + P_0 = P_1 - P_0$.
3. ***Orthogonality:*** $(P_1 - P_0)P_0 = P_1P_0 - P_0 = P_0 - P_0 = 0$.

2.6.3 Decomposition of Projections and their Sum Squares

Theorem 2.17 (Orthogonal Decomposition)**.** *Let $M_0 \subset M_1$ be two nested linear models with corresponding design matrices X_0 and X_1 such that $C(X_0) \subset C(X_1)$. Let P_0 and P_1 be the orthogonal projection matrices onto $C(X_0)$ and $C(X_1)$ respectively.*

For any observation vector y, we have the decomposition:

$$y = \underbrace{P_0 y}_{\hat{y}_0} + \underbrace{(P_1 - P_0)y}_{\hat{y}_1 - \hat{y}_0} + \underbrace{(I - P_1)y}_{y - \hat{y}_1}$$

Geometric Interpretation:

1. *$\hat{y}_0 \in C(X_0)$: The fit of the reduced model.*
2. *$(\hat{y}_1 - \hat{y}_0) \in C(X_0)^\perp \cap C(X_1)$: The additional fit provided by M_1 over M_0.*
3. *$(y - \hat{y}_1) \in C(X_1)^\perp$: The projection of y onto the* ***orthogonal complement*** *of $C(X_1)$.*

The three component vectors are mutually orthogonal. Consequently, their squared norms sum to the total squared norm:

$$\|y\|^2 = \|\hat{y}_0\|^2 + \|\hat{y}_1 - \hat{y}_0\|^2 + \|y - \hat{y}_1\|^2$$

Proof. **1. Definitions** We define the three components as vectors v_1, v_2, v_3:

- $v_1 = \hat{y}_0 = P_0 y$.
- $v_2 = \hat{y}_1 - \hat{y}_0 = (P_1 - P_0)y$.
- $v_3 = y - \hat{y}_1 = (I - P_1)y$.
  - **Note:** Since P_1 projects onto $C(X_1)$, the matrix $(I - P_1)$ projects onto the **orthogonal complement** $C(X_1)^\perp$. Thus, $v_3 \in \text{Col}(I - P_1)$.

Note that since $C(X_0) \subset C(X_1)$, we have the property $P_1P_0 = P_0P_1 = P_0$. (Projecting onto the smaller subspace M_0 is unchanged if we first project onto the enclosing subspace M_1).

2. Orthogonality of v_1 and v_2 We check the inner product $\langle v_1, v_2 \rangle = v_1' v_2$:

$$\begin{aligned}
v_1' v_2 &= (P_0 y)'(P_1 - P_0)y \\
&= y' P_0' (P_1 - P_0)y \\
&= y'(P_0P_1 - P_0^2)y \quad \text{(Since } P_0 \text{ is symmetric)} \\
&= y'(P_0 - P_0)y \quad \text{(Since } P_0P_1 = P_0 \text{ and } P_0^2 = P_0\text{)} \\
&= 0
\end{aligned}$$

3. Orthogonality of $(v_1 + v_2)$ and v_3 Note that $v_1 + v_2 = P_1 y = \hat{y}_1$. We check if the total fit $\hat{y}_1$ is orthogonal to the residual v_3:

$$\begin{aligned}
\hat{y}_1' v_3 &= (P_1 y)'(I - P_1)y \\
&= y' P_1 (I - P_1) y \\
&= y'(P_1 - P_1^2) y \\
&= y'(P_1 - P_1) y \\
&= 0
\end{aligned}$$

Since $\hat{y}_1$ is orthogonal to v_3, and $\hat{y}_0$ is a component of $\hat{y}_1$, it follows that all three pieces are mutually orthogonal.

4. Sum of Squares By the Pythagorean theorem applied twice to these orthogonal vectors, the equality of squared norms follows immediately. □

Example 2.5 (Geometric Interpretation of One-way ANOVA)**.** We apply the **Nested Model Theorem** ($M_0 \subset M_1$) to the One-way ANOVA setting with k groups and n_i observations per group.

1. The Data Vector We stack all observations into a single $N \times 1$ vector ($N = \sum n_i$):

$$y = \left(\begin{array}{c} y_{11} \\ \vdots \\ y_{1n_1} \\ \hline \vdots \\ \hline y_{k1} \\ \vdots \\ y_{kn_k} \end{array} \right)$$

2. The Projections (as Vectors)

- $\hat{y}_0$ **(Null Model Projection):** Projecting onto the intercept vector **1** replaces every observation with the **Grand Mean** $\bar{y}_{..}$.

$$\hat{y}_0 = P_0 y = \left(\begin{array}{c} \bar{y}_{..} \\ \vdots \\ \bar{y}_{..} \\ \hline \vdots \\ \hline \bar{y}_{..} \\ \vdots \\ \bar{y}_{..} \end{array} \right)$$

- $\hat{y}_1$ **(Full Model Projection):** Projecting onto the group indicators replaces observations in group i

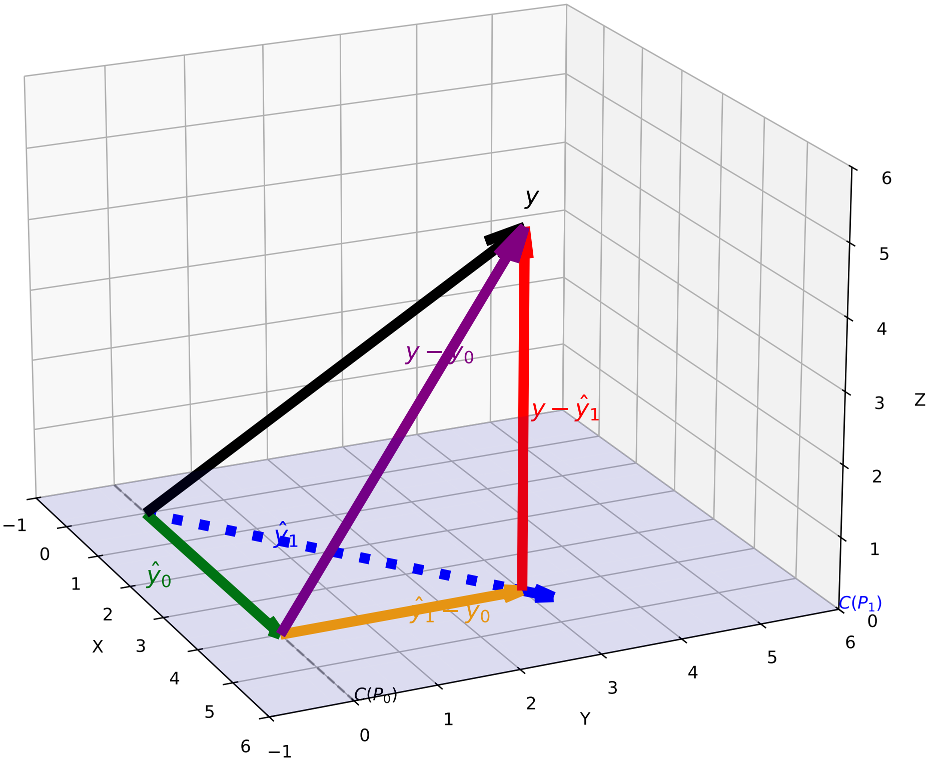

Figure 2.3: Illustration of Projections onto Nested Subspaces

with the **Group Mean** $\bar{y}_{i.}$.

$$\hat{y}_1 = P_1 y = \begin{pmatrix} \bar{y}_{1.} \\ \vdots \\ \bar{y}_{1.} \\ \hline \vdots \\ \hline \bar{y}_{k.} \\ \vdots \\ \bar{y}_{k.} \end{pmatrix}$$

3. The Decomposition The total deviation vector $y - \hat{y}_0$ splits into two orthogonal vector components:

- **Model Improvement Vector ($\hat{y}_1 - \hat{y}_0$):** The difference between group means and the grand mean.

$$\hat{y}_1 - \hat{y}_0 = \begin{pmatrix} \bar{y}_{1.} - \bar{y}_{..} \\ \vdots \\ \bar{y}_{1.} - \bar{y}_{..} \\ \hline \vdots \\ \hline \bar{y}_{k.} - \bar{y}_{..} \\ \vdots \\ \bar{y}_{k.} - \bar{y}_{..} \end{pmatrix}$$

Squared Norm: $RSS_0 - RSS_1 = \sum_i n_i(\bar{y}_{i.} - \bar{y}_{..})^2$ $(SS_{between})$

- **Residual Vector ($y - \hat{y}_1$):** The difference between individual observations and their specific group mean.

$$y - \hat{y}_1 = \begin{pmatrix} y_{11} - \bar{y}_{1.} \\ \vdots \\ y_{1n_1} - \bar{y}_{1.} \\ \hline \vdots \\ \hline y_{k1} - \bar{y}_{k.} \\ \vdots \\ y_{kn_k} - \bar{y}_{k.} \end{pmatrix}$$

Squared Norm: $RSS_1 = \sum_i \sum_j (y_{ij} - \bar{y}_{i.})^2$ (SS_{within})

Conclusion: By orthogonality, the squared length of the total deviation equals the sum of the squared lengths of these components:

$$\underbrace{||\hat{y}_1 - \hat{y}_0||^2}_{SS_{between}} + \underbrace{||y - \hat{y}_1||^2}_{SS_{within}} = \underbrace{||y - \hat{y}_0||^2}_{TSS}$$

2.6.4 Projections in General Orthogonal Subspaces

Finally, we consider the case where the entire space $\mathbb{R}^n$ is decomposed into mutually orthogonal subspaces.

Theorem 2.18 (General Orthogonal Projects)**.** *If $\mathbb{R}^n$ is the direct sum of orthogonal subspaces $V_1, V_2, \dots, V_k$:*

$$\mathbb{R}^n = V_1 \oplus V_2 \oplus \cdots \oplus V_k$$

where $V_i \perp V_j$ for all $i \neq j$.

Then any vector y can be uniquely written as:

$$y = x_1 + x_2 + \cdots + x_k$$

where $x_i \in V_i$.

Furthermore, each component x_i is simply the projection of y onto the subspace V_i:

$$x_i = P_i y$$

This implies that the identity matrix can be decomposed into a sum of projection matrices:

$$I_n = P_1 + P_2 + \cdots + P_k$$

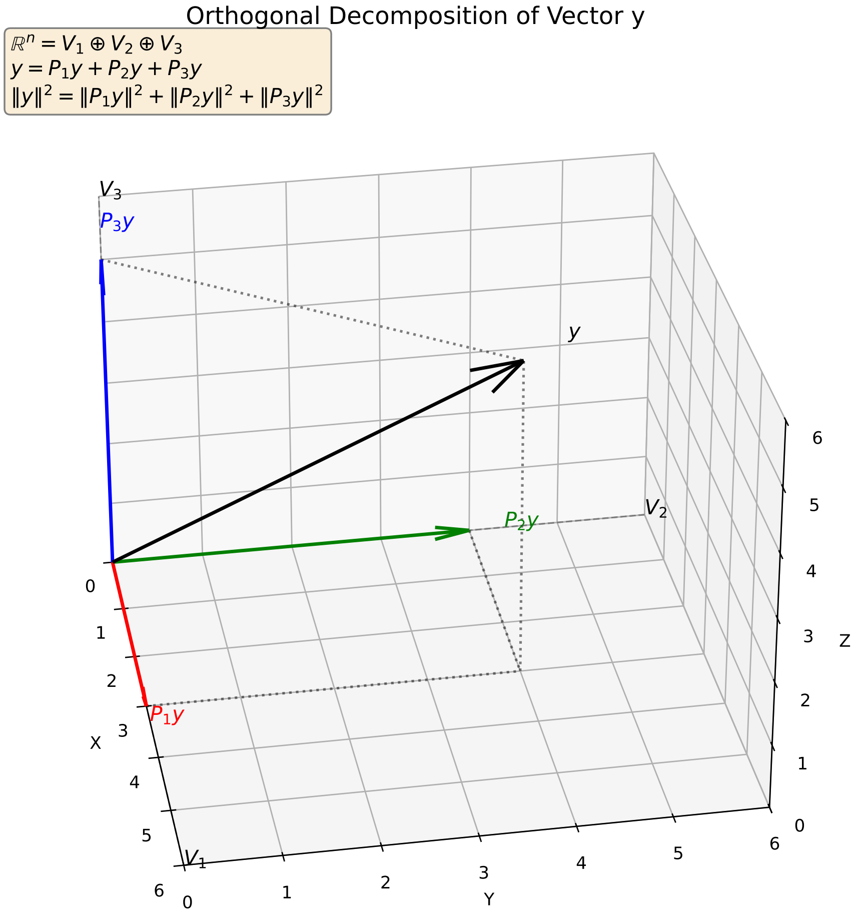
Orthogonal Decomposition of Vector y
$\mathbb{R}^n = V_1 \oplus V_2 \oplus V_3$
$y = P_1y + P_2y + P_3y$
$\|y\|^2 = \|P_1y\|^2 + \|P_2y\|^2 + \|P_3y\|^2$
V_3
P_3y
y
P_2y
V_2
P_1y
V_1
X
0
1
2
3
4
5
6
Y
0
1
2
3
4
5
6
Z
0
1
2
3
4
5
6

Figure 2.4: Orthogonal decomposition of vector y into subspaces

3 Spectral Theory and Generalized Inverse

This chapter covers a review of matrix algebra concepts essential for linear models, including eigenvalues, spectral decomposition, and generalized inverses.

3.1 Spectral Theory

3.1.1 Eigenvalues and Eigenvectors

Definition 3.1 (Eigenvalues and Eigenvectors)**.** For a square matrix A $(n \times n)$, a scalar λ is an **eigenvalue** and a non-zero vector x is the corresponding **eigenvector** if:

$$Ax = \lambda x \iff (A - \lambda I_n)x = 0$$

The eigenvalues are found by solving the characteristic equation:

$$|A - \lambda I_n| = 0$$

3.1.2 Quadratic Form

Definition 3.2. A **quadratic form** in n variables $x_1, x_2, \dots, x_n$ is a scalar function defined by a symmetric matrix A:

$$Q(x) = x'Ax = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j$$

3.1.3 Positive and Non-Negative Definite Matrices

Definition 3.3 (Positive and Non-Negative Definite Matrices)**.** A symmetric matrix A is **positive definite (p.d.)** if:

$$x'Ax > 0 \quad \forall x \neq 0$$

It is **non-negative definite (n.n.d.)** if:

$$x'Ax \geq 0 \quad \forall x$$

Theorem 3.1 (Properties of Definite Matrices)**.** *Let A be a symmetric $n \times n$ matrix with eigenvalues $\lambda_1, \dots, \lambda_n$.*

1. ***Eigenvalue Characterization:***

   - *A is p.d. $\iff$ all $\lambda_i > 0$.*
   - *A is n.n.d. $\iff$ all $\lambda_i \geq 0$.*

2. ***Determinant and Inverse:***

   - *If A is p.d., then $|A| > 0$ and A^{-1} exists.*
   - *If A is n.n.d. and singular, then $|A| = 0$ (at least one $\lambda_i = 0$).*

3. ***Gram Matrices ($B'B$):*** *Let B be an $n \times p$ matrix.*

   - *If rank$(B) = p$, then $B'B$ is p.d.*
   - *If rank$(B) < p$, then $B'B$ is n.n.d.*

3.1.4 Properties of Symmetric Matrices

Theorem 3.2 (Properties of Symmetric Matrices)**.** *Let A be a symmetric matrix with spectral decomposition $A = Q\Lambda Q'$. The following properties hold:*

1. ***Trace:*** *tr$(A) = \sum \lambda_i$.*
2. ***Determinant:*** *$|A| = \prod \lambda_i$.*
3. ***Singularity:*** *A is singular if and only if at least one $\lambda_i = 0$.*
4. ***Inverse:*** *If A is non-singular ($\lambda_i \neq 0$), then $A^{-1} = Q\Lambda^{-1}Q'$.*
5. ***Powers:*** *$A^k = Q\Lambda^k Q'$.*

   - Square Root: *$A^{1/2} = Q\Lambda^{1/2}Q'$ (if $\lambda_i \geq 0$).*

6. ***Spectral Representation of Quadratic Forms:*** *The quadratic form $x'Ax$ can be diagonalized using the eigenvectors of A:*

$$x'Ax = x'Q\Lambda Q'x = y'\Lambda y = \sum_{i=1}^{n} \lambda_i y_i^2$$

   where $y = Q'x$ represents a rotation of the coordinate system.

3.1.5 Spectral Representation of Projection Matrices

We revisit projection matrices in the context of eigenvalues.

Theorem 3.3 (Eigenvalues of Projection Matrices)**.** *A symmetric matrix P is a projection matrix (idempotent, $P^2 = P$) if and only if its eigenvalues are either 0 or 1.*

$$P^2x = \lambda^2 x \quad \text{and} \quad Px = \lambda x \implies \lambda^2 = \lambda \implies \lambda \in \{0, 1\}$$

For a projection matrix P:

- If $x \in Col(P)$, $Px = x$ (Eigenvalue 1).

- If $x \perp Col(P)$, $Px = 0$ (Eigenvalue 0).
- rank$(P) =$ tr$(P) = \sum \lambda_i$ (Count of 1s).

Example 3.1. For $P = \frac{1}{n} J_n J_n'$, the rank is tr$(P) = 1$.

3.1.6 Singular Value Decomposition (SVD)

Theorem 3.4 (Singular Value Decomposition (SVD))**.** *Let X be an $n \times p$ matrix with rank $r \leq \min(n, p)$. X can be decomposed into the product of three matrices:*

$$X = U\mathbf{D}V'$$

1. Partitioned Matrix Form

$$X = \underset{n\times n}{(U_1, U_2)} \begin{pmatrix} \Lambda_r & O_{r\times(p-r)} \\ O_{(n-r)\times r} & O_{(n-r)\times(p-r)} \end{pmatrix} \underset{p\times p}{\begin{pmatrix} V_1' \\ V_2' \end{pmatrix}}$$

2. Detailed Matrix Form

Expanding the diagonal matrix explicitly:

$$X = \underset{n\times n}{(u_1, \dots, u_n)} \left(\begin{array}{cccc|c} \lambda_1 & 0 & \dots & 0 & \\ 0 & \lambda_2 & \dots & 0 & O_{12} \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \dots & \lambda_r & \\ \hline \multicolumn{4}{c|}{O_{21}} & O_{22} \end{array}\right) \underset{p\times p}{\begin{pmatrix} v_1' \\ \vdots \\ v_p' \end{pmatrix}}$$

3. Reduced Form

$$X = U_1 \Lambda_r V_1' = \sum_{i=1}^{r} \lambda_i u_i v_i'$$

Properties:

1. ***Singular Values (Λ_r):*** *$\Lambda_r =$ diag$(\lambda_1, \dots, \lambda_r)$ contains the singular values ($\lambda_i > 0$), which are the square roots of the non-zero eigenvalues of $X'X$.*
2. ***Orthogonality:***
    - *U is $n \times n$ orthogonal ($U'U = I_n$).*
    - *V is $p \times p$ orthogonal ($V'V = I_p$).*

3.1.6.1 Connection to Gram Matrices

The matrices U and V provide the basis vectors (eigenvectors) for the Gram matrices of X.

1. **Right Singular Vectors (V):** The columns of V are the eigenvectors of the Gram matrix $X'X$.
$$X'X = (U\Lambda V')'(U\Lambda V') = V\Lambda U'U\Lambda V' = V\Lambda^2 V'$$
   - The eigenvalues of $X'X$ are the squared singular values λ_i^2.
2. **Left Singular Vectors (U):** The columns of U are the eigenvectors of the Gram matrix XX'.
$$XX' = (U\Lambda V')(U\Lambda V')' = U\Lambda V'V\Lambda U' = U\Lambda^2 U'$$
   - The eigenvalues of XX' are also λ_i^2 (for non-zero values).

3.1.6.2 Numerical Example

Consider the matrix $X = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}$.

1. **Compute $X'X$ and find V:**
$$X'X = \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix} = \begin{pmatrix} 5 & 5 \\ 5 & 5 \end{pmatrix}$$
   - Eigenvalues of $X'X$: Trace is 10, Determinant is 0. Thus, $\mu_1 = 10, \mu_2 = 0$.
   - **Singular Values:** $\lambda_1 = \sqrt{10}, \lambda_2 = 0$.
   - Eigenvector for $\mu_1 = 10$: Normalized $v_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.
   - Eigenvector for $\mu_2 = 0$: Normalized $v_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$.
   - Therefore, $V = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.
2. **Compute XX' and find U:**
$$XX' = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 4 \\ 4 & 8 \end{pmatrix}$$
   - Eigenvalues are again 10 and 0.
   - Eigenvector for $\mu_1 = 10$: Normalized $u_1 = \frac{1}{\sqrt{5}} \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.
   - Eigenvector for $\mu_2 = 0$: Normalized $u_2 = \frac{1}{\sqrt{5}} \begin{pmatrix} 2 \\ -1 \end{pmatrix}$.
   - Therefore, $U = \frac{1}{\sqrt{5}} \begin{pmatrix} 1 & 2 \\ 2 & -1 \end{pmatrix}$.
3. **Verification:**
$$X = \sqrt{10} u_1 v_1' = \sqrt{10} \begin{pmatrix} \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 2 \end{pmatrix}$$

3.2 Cholesky Decomposition

A symmetric matrix A has a Cholesky decomposition if and only if it is **non-negative definite** (i.e., $x'Ax \geq 0$ for all x).

$$A = B'B$$

where B is an **upper triangular** matrix with non-negative diagonal entries.

3.2.1 Matrix Representation of the Algorithm

To derive the algorithm, we equate the elements of A with the product of the lower triangular matrix B' and the upper triangular matrix B.

For a 3×3 matrix, this looks like:

$$\underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}}_{A} = \underbrace{\begin{pmatrix} b_{11} & 0 & 0 \\ b_{12} & b_{22} & 0 \\ b_{13} & b_{23} & b_{33} \end{pmatrix}}_{B'} \underbrace{\begin{pmatrix} b_{11} & b_{12} & b_{13} \\ 0 & b_{22} & b_{23} \\ 0 & 0 & b_{33} \end{pmatrix}}_{B}$$

Multiplying the matrices on the right yields the system of equations:

$$A = \begin{pmatrix} \mathbf{b_{11}^2} & b_{11}b_{12} & b_{11}b_{13} \\ b_{12}b_{11} & \mathbf{b_{12}^2} + \mathbf{b_{22}^2} & b_{12}b_{13} + b_{22}b_{23} \\ b_{13}b_{11} & b_{13}b_{12} + b_{23}b_{22} & \mathbf{b_{13}^2} + \mathbf{b_{23}^2} + \mathbf{b_{33}^2} \end{pmatrix}$$

By solving for the bolded diagonal terms and substituting known values from previous rows, we get the recursive algorithm.

3.2.2 The Algorithm

1. **Row 1:** Solve for b_{11} using a_{11}, then solve the rest of the row (b_{1j}) by division.
   - $b_{11} = \sqrt{a_{11}}$
   - $b_{1j} = a_{1j}/b_{11}$
2. **Row 2:** Solve for b_{22} using a_{22} and the known b_{12}, then solve b_{2j}.
   - $b_{22} = \sqrt{a_{22} - b_{12}^2}$
   - $b_{2j} = (a_{2j} - b_{12}b_{1j})/b_{22}$
3. **Row 3:** Solve for b_{33} using a_{33} and the known b_{13}, b_{23}.
   - $b_{33} = \sqrt{a_{33} - b_{13}^2 - b_{23}^2}$

3.2.3 Numerical Example

Consider the positive definite matrix A:

$$A = \begin{pmatrix} 4 & 2 & -2 \\ 2 & 10 & 2 \\ -2 & 2 & 6 \end{pmatrix}$$

We find B such that $A = B'B$:

1. **First Row of B (b_{11}, b_{12}, b_{13}):**
   - $b_{11} = \sqrt{4} = 2$
   - $b_{12} = 2/2 = 1$
   - $b_{13} = -2/2 = -1$
2. **Second Row of B (b_{22}, b_{23}):**
   - $b_{22} = \sqrt{10 - (1)^2} = \sqrt{9} = 3$
   - $b_{23} = (2 - (1)(-1))/3 = 3/3 = 1$
3. **Third Row of B (b_{33}):**
   - $b_{33} = \sqrt{6 - (-1)^2 - (1)^2} = \sqrt{4} = 2$

Result:

$$B = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 3 & 1 \\ 0 & 0 & 2 \end{pmatrix}$$

3.3 Generalized Inverses

3.3.1 Motivation

Consider the linear system $X\beta = y$. In $\mathbb{R}^2$, if $X = [x_1, x_2]$ is invertible, the solution is unique: $\beta = X^{-1}y$. This satisfies $X(X^{-1}y) = y$.However, if X is not square or not invertible (e.g., X is 2×3), $X\beta = y$ does not have a unique solution. We seek a matrix G such that $\beta = Gy$ provides a solution whenever $y \in C(X)$ (the column space of X). Substituting $\beta = Gy$ into the equation $X\beta = y$:

$$X(Gy) = y \quad \forall y \in C(X)$$

Since any $y \in C(X)$ can be written as Xw for some vector w:

$$XGXw = Xw \quad \forall w$$

This implies the defining condition:

$$XGX = X$$

3.3.2 Definition of Generalized Inverse

Definition 3.4 (Generalized Inverse)**.** Let X be an $n \times p$ matrix. A matrix X^- of size $p \times n$ is called a **generalized inverse** of X if it satisfies:

$$XX^-X = X$$

Example 3.2 (Examples of Generalized Inverse)**.**

- **Example 1: Diagonal Matrix** If $X = \text{diag}(\lambda_1, \lambda_2, 0, 0)$, we can write it in matrix form as:

$$X = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

  A generalized inverse is obtained by inverting the non-zero elements:

$$X^- = \begin{pmatrix} \lambda_1^{-1} & 0 & 0 & 0 \\ 0 & \lambda_2^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

- **Example 2: Row Vector** Let $X = (1, 2, 3)$. One possible generalized inverse is a column vector where the first element is the reciprocal of the first non-zero element of X (which is 1), and others are zero:

$$X^- = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

  Verification:

$$XX^-X = (1, 2, 3) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} (1, 2, 3) = (1) \cdot (1, 2, 3) = (1, 2, 3) = X$$

  Other valid generalized inverses include $\begin{pmatrix} 0 \\ 1/2 \\ 0 \end{pmatrix}$ or $\begin{pmatrix} 0 \\ 0 \\ 1/3 \end{pmatrix}$.

- **Example 3: Rank Deficient Matrix** Let $A = \begin{pmatrix} 2 & 2 & 3 \\ 1 & 0 & 1 \\ 3 & 2 & 4 \end{pmatrix}$. Note that Row 3 = Row 1 + Row 2, so Rank$(A) = 2$.

  Solution: A generalized inverse can be found by locating a non-singular 2 × 2 submatrix, inverting it, and padding the rest with zeros. Let's take the top-left minor $M = \begin{pmatrix} 2 & 2 \\ 1 & 0 \end{pmatrix}$. The inverse is $M^{-1} = \frac{1}{-2} \begin{pmatrix} 0 & -2 \\ -1 & 2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0.5 & -1 \end{pmatrix}$.

Placing this in the corresponding position in A^- and setting the rest to 0:

$$A^- = \begin{pmatrix} 0 & 1 & 0 \\ 0.5 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Verification ($AA^-A = A$): First, compute AA^-:

$$AA^- = \begin{pmatrix} 2 & 2 & 3 \\ 1 & 0 & 1 \\ 3 & 2 & 4 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 0.5 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

Then multiply by A:

$$(AA^-)A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 2 & 3 \\ 1 & 0 & 1 \\ 3 & 2 & 4 \end{pmatrix} = \begin{pmatrix} 2 & 2 & 3 \\ 1 & 0 & 1 \\ 3 & 2 & 4 \end{pmatrix} = A$$

3.3.3 A General Procedure to Find a Generalized Inverse

If we can partition X (possibly after permuting rows/columns) such that R_{11} is a non-singular rank r submatrix:

$$X = \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix}$$

Then a generalized inverse is:

$$X^- = \begin{pmatrix} R_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix}$$

Verification:

$$\begin{aligned} XX^-X &= \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} R_{11}^{-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \\ &= \begin{pmatrix} I_r & 0 \\ R_{21}R_{11}^{-1} & 0 \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \\ &= \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{21}R_{11}^{-1}R_{12} \end{pmatrix} \end{aligned}$$

Note that since $\text{rank}(X) = \text{rank}(R_{11})$, the rows of $[R_{21}, R_{22}]$ are linear combinations of $[R_{11}, R_{12}]$, implying $R_{22} = R_{21}R_{11}^{-1}R_{12}$. Thus, $XX^-X = X$.

An Algorithm for Finding a Generalized Inverse

A systematic procedure to find a generalized inverse A^- for any matrix A:

1. Find any non-singular $r \times r$ submatrix C, where r is the rank of A. It is not necessary for the elements of C to occupy adjacent rows and columns in A.
2. Find C^{-1} and $(C^{-1})'$.
3. Replace the elements of C in A with the elements of $(C^{-1})'$.
4. Replace all other elements in A with zeros.
5. Transpose the resulting matrix.

Matrix Visual Representation

$$\underset{\text{Original } A}{\begin{pmatrix} \times & \otimes & \times & \otimes \\ \times & \otimes & \times & \otimes \\ \times & \times & \times & \times \end{pmatrix}} \xrightarrow[\text{with } (C^{-1})']{\text{Replace } C} \underset{\text{Intermediate}}{\begin{pmatrix} \times & \triangle & \times & \triangle \\ \times & \triangle & \times & \triangle \\ \times & \times & \times & \times \end{pmatrix}} \xrightarrow[\text{Result}]{\text{Transpose}} \underset{\text{Final } A^{-}}{\begin{pmatrix} \times & \times & \times \\ \square & \square & \times \\ \times & \times & \times \\ \square & \square & \times \end{pmatrix}}$$

Legend: * $\otimes$: Elements of submatrix C * $\triangle$: Elements of $(C^{-1})'$ * $\square$: Elements of C^{-1} (after transposition) * $\times$: Other elements (replaced by 0 in the final calculation)

3.3.4 Moore-Penrose Inverse

The Moore-Penrose inverse (denoted X^{+}) is a unique generalized inverse defined via Singular Value Decomposition (SVD).

If X has SVD:

$$X = U \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} V'$$

Then the Moore-Penrose inverse is:

$$X^{+} = V \begin{pmatrix} \Lambda_r^{-1} & 0 \\ 0 & 0 \end{pmatrix} U'$$

where $\Lambda_r = \text{diag}(\lambda_1, \dots, \lambda_r)$ contains the singular values. Unlike standard generalized inverses, X^{+} is unique.

Verification:

We verify that X^{+} satisfies the condition $XX^{+}X = X$.

1. **Substitute definitions:**

$$XX^{+}X = \left[U \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} V'\right] \left[V \begin{pmatrix} \Lambda_r^{-1} & 0 \\ 0 & 0 \end{pmatrix} U'\right] \left[U \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} V'\right]$$

2. **Apply orthogonality:** Recall that $V'V = I$ and $U'U = I$.

$$= U \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} \underbrace{(V'V)}_{I} \begin{pmatrix} \Lambda_r^{-1} & 0 \\ 0 & 0 \end{pmatrix} \underbrace{(U'U)}_{I} \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} V'$$

3. **Multiply diagonal matrices:**

$$= U \left[\begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Lambda_r^{-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} \right] V'$$

Since $\Lambda_r \Lambda_r^{-1} \Lambda_r = I \cdot \Lambda_r = \Lambda_r$:

$$= U \begin{pmatrix} \Lambda_r & 0 \\ 0 & 0 \end{pmatrix} V' = X$$

3.3.5 Solving Linear Systems with Generalized Inverse

We apply generalized inverses to solve systems of linear equations $X\beta = c$ where X is $n \times p$.

Definition 3.5 (Consistency and Solution)**.** The system $X\beta = c$ is consistent if and only if $c \in \mathcal{C}(X)$ (the column space of X). If consistent, $\beta = X^- c$ is a solution.

Proof: If the system is consistent, there exists some b such that $Xb = c$. Using the definition $XX^-X = X$:

$$X(X^-c) = X(X^-Xb) = (XX^-X)b = Xb = c$$

Thus, X^-c is a solution. Note that the solution is not unique if X is not full rank.

Example 3.3 (Examples of Solutions of Linear System with Generalized Inverse)**.**

- **Example 1: Underdetermined System**

  Let $X = \begin{pmatrix} 1 & 2 & 3 \end{pmatrix}$ and we want to solve $X\beta = 4$.

  Solution 1: Using the generalized inverse $X^- = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$:

$$\beta = X^- \cdot 4 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} 4 = \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}$$

  Verification:

$$X\beta = \begin{pmatrix} 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix} = 1(4) + 2(0) + 3(0) = 4 \quad \checkmark$$

  Solution 2: Using another generalized inverse $X^- = \begin{pmatrix} 0 \\ 0 \\ 1/3 \end{pmatrix}$:

$$\beta = X^- \cdot 4 = \begin{pmatrix} 0 \\ 0 \\ 1/3 \end{pmatrix} 4 = \begin{pmatrix} 0 \\ 0 \\ 4/3 \end{pmatrix}$$

Verification:

$$X\beta = (1 \quad 2 \quad 3)\begin{pmatrix} 0 \\ 0 \\ 4/3 \end{pmatrix} = 0 + 0 + 3(4/3) = 4 \quad \checkmark$$

- **Example 2: Overdetermined System**

Let $X = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$. Solve $X\beta = \begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix} = c$. Here $c = 2X$, so the system is consistent. Since X is a column vector, β is a scalar.

Solution: Using the generalized inverse $X^{-} = (1 \quad 0 \quad 0)$:

$$\beta = X^{-}c = (1 \quad 0 \quad 0)\begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix} = 1(2) + 0(4) + 0(6) = 2$$

Verification:

$$X\beta = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}(2) = \begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix} = c \quad \checkmark$$

3.4 Least Squares with Generalized Inverse

3.4.1 Projection Matrix with Generalized Inverse of $X'X$

For the normal equations $(X'X)\beta = X'y$, a solution is given by:

$$\hat{\beta} = (X'X)^{-}X'y$$

The fitted values are

$$\hat{y} = X\hat{\beta} = X(X'X)^{-}X'y.$$

This $\hat{y}$ represents the unique orthogonal projection of y onto $Col(X)$.

3.4.2 Invariance and Uniqueness of "the" Projection Matrix

Theorem 3.5 (Transpose Property of Generalized Inverses)**.** *$(X^{-})'$ is a version of $(X')^{-}$. That is, $(X^{-})'$ is a generalized inverse of X'.*

Proof. By definition, a generalized inverse X^{-} satisfies the property:

$$XX^{-}X = X$$

To verify that $(X^{-})'$ is a generalized inverse of X', we need to show that it satisfies the condition $AGA = A$ where $A = X'$ and $G = (X^{-})'$.

1. Start with the fundamental definition:
$$XX^{-}X = X$$
2. Take the transpose of both sides of the equation:
$$(XX^{-}X)' = X'$$
3. Apply the reverse order law for transposes, $(ABC)' = C'B'A'$:
$$X'(X^{-})'X' = X'$$

Since substituting $(X^{-})'$ into the generalized inverse equation for X' yields X', $(X^{-})'$ is a valid generalized inverse of X'. □

Lemma 3.1 (Invariance of Generalized Least Squares)**.** *For any version of the generalized inverse $(X'X)^{-}$, the matrix $X'(X'X)^{-}X'$ is invariant and equals X'.*

$$X'X(X'X)^{-}X' = X'$$

Proof (using Projection): Let $P = X(X'X)^{-}X'$. This is the projection matrix onto $\mathcal{C}(X)$. By definition of projection, $Px = x$ for any $x \in Col(X)$. Since columns of X are in $Col(X)$, $PX = X$. Taking the transpose: $(PX)' = X' \implies X'P' = X'$. Since projection matrices are symmetric ($P = P'$), $X'P = X'$. Substituting P: $X'X(X'X)^{-}X' = X'$.

Proof (Direct Matrix Manipulation): Decompose $y = X\beta + e$ where $e \perp Col(X)$ (i.e., $X'e = 0$).

$$\begin{aligned} X'X(X'X)^{-}X'y &= X'X(X'X)^{-}X'(X\beta + e) \\ &= X'X(X'X)^{-}X'X\beta + X'X(X'X)^{-}X'e \end{aligned}$$

Using the property $AA^{-}A = A$ (where $A = X'X$), the first term becomes $X'X\beta$. The second term is 0 because $X'e = 0$. Thus, the expression simplifies to $X'X\beta = X'(X\beta) = X'\hat{y}_{proj}$. This implies the operator acts as X'.

Theorem 3.6 (Properties of Projection Matrix P)**.** *Let $P = X(X'X)^{-}X'$. This matrix has the following properties:*

1. ***Symmetry:*** $P = P'$.

2. ***Idempotence:*** $P^2 = P$.

$$P^2 = X(X'X)^{-}X'X(X'X)^{-}X' = X(X'X)^{-}(X'X(X'X)^{-}X')$$

Using the identity from Lemma 3.1 ($X'X(X'X)^{-}X' = X'$), this simplifies to:

$$X(X'X)^{-}X' = P$$

3. ***Uniqueness:*** *P is unique and invariant to the choice of the generalized inverse $(X'X)^{-}$.*

Proof. **Proof of Uniqueness:**

Let A and B be two different generalized inverses of $X'X$. Define $P_A = XAX'$ and $P_B = XBX'$. From Lemma 3.1, we know that $X'P_A = X'$ and $X'P_B = X'$.

Subtracting these two equations:

$$X'(P_A - P_B) = 0$$

Taking the transpose, we get $(P_A - P_B)X = 0$. This implies that the columns of the difference matrix $D = P_A - P_B$ are orthogonal to the columns of X (i.e., $D \perp Col(X)$).

However, by definition, the columns of P_A and P_B (and thus D) are linear combinations of the columns of X (i.e., $D \in Col(X)$).

The only matrix that lies *in* the column space of X but is also *orthogonal* to the column space of X is the zero matrix. Therefore:

$$P_A - P_B = 0 \implies P_A = P_B$$

□

3.4.3 An Explicit Expression

When X has rank $r < p$ (where X is $n \times p$), we can derive the least squares estimator using partitioned matrices.

Assume the first r columns of X are linearly independent. We can partition X as:

$$X = Q(R_1, R_2)$$

where Q is an $n \times r$ matrix with orthogonal columns ($Q'Q = I_r$), R_1 is an $r \times r$ non-singular matrix, and R_2 is $r \times (p - r)$.

The normal equations are:

$$X'X\beta = X'y \implies \begin{pmatrix} R_1' \\ R_2' \end{pmatrix} Q'Q(R_1, R_2)\beta = \begin{pmatrix} R_1' \\ R_2' \end{pmatrix} Q'y$$

Simplifying ($Q'Q = I_r$):

$$\begin{pmatrix} R_1'R_1 & R_1'R_2 \\ R_2'R_1 & R_2'R_2 \end{pmatrix} \beta = \begin{pmatrix} R_1'Q'y \\ R_2'Q'y \end{pmatrix}$$

3.4.3.1 Constructing a Solution by Solving Normal Equations

Proof. One specific generalized inverse of $X'X$ can be found by focusing on the non-singular block $R_1'R_1$:

$$(X'X)^- = \begin{pmatrix} (R_1'R_1)^{-1} & 0 \\ 0 & 0 \end{pmatrix}$$

Using this generalized inverse, the estimator $\widehat{\beta}$ becomes:

$$\widehat{\beta} = (X'X)^- X'y = \begin{pmatrix} (R_1'R_1)^{-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} R_1'Q'y \\ R_2'Q'y \end{pmatrix}$$

$$\widehat{\beta} = \begin{pmatrix} (R_1'R_1)^{-1}R_1'Q'y \\ 0 \end{pmatrix} = \begin{pmatrix} R_1^{-1}Q'y \\ 0 \end{pmatrix}$$

The fitted values are:

$$\widehat{y} = X\widehat{\beta} = Q(R_1, R_2) \begin{pmatrix} R_1^{-1}Q'y \\ 0 \end{pmatrix} = QR_1R_1^{-1}Q'y = QQ'y$$

This confirms that $\widehat{y}$ is the projection of y onto the column space of Q (which is the same as the column space of X). □

3.4.3.2 Constructing a Solution by Solving Reparametrized β

We can view the model as:

$$y = Q(R_1, R_2)\beta + \epsilon = Qb + \epsilon$$

where $b = R_1\beta_1 + R_2\beta_2$.

Since the columns of Q are orthogonal, the least squares estimate for b is simply:

$$\widehat{b} = (Q'Q)^{-1}Q'y = Q'y$$

To find β, we solve the underdetermined system:

$$R_1\beta_1 + R_2\beta_2 = \widehat{b} = Q'y$$

Proof. **Solution Strategy 1:** Set $\beta_2 = 0$. Then:

$$R_1\beta_1 = Q'y \implies \widehat{\beta}_1 = R_1^{-1}Q'y$$

This yields the same result as the generalized inverse method above: $\widehat{\beta} = \begin{pmatrix} R_1^{-1}Q'y \\ 0 \end{pmatrix}$.

Solution Strategy 2: Using the generalized inverse of $R = (R_1, R_2)$:

$$R^- = \begin{pmatrix} R_1^{-1} \\ 0 \end{pmatrix}$$

$$\hat{\beta} = R^{-}Q'y = \begin{pmatrix} R_1^{-1}Q'y \\ 0 \end{pmatrix}$$

This demonstrates that finding a solution to the normal equations using $(X'X)^{-}$ is equivalent to solving the reparameterized system $b = R\beta$. □

---

4 Multivariate Normal Distribution

4.1 Motivation

Consider the linear model:

$$y = X\beta + \epsilon, \quad \epsilon_i \sim N(0, \sigma^2)$$

We are often interested in the distributional properties of the response vector y and the residuals. Specifically, if $y = (y_1, \dots, y_n)'$, we need to understand its multivariate distribution.

$$\hat{y} = Py, \quad e = y - \hat{y} = (I_n - P)y$$

4.2 Random Vectors and Matrices

Definition 4.1 (Random Vector and Matrix)**.** A **Random Vector** is a vector whose elements are random variables. E.g.,

$$x_{k\times 1} = (x_1, x_2, \dots, x_k)^T$$

where $x_1, \dots, x_k$ are each random variables.

A **Random Matrix** is a matrix whose elements are random variables. E.g., $X_{n\times k} = (x_{ij})$, where $x_{11}, \dots, x_{nk}$ are each random variables.

Definition 4.2 (Expected Value)**.** The expected value (population mean) of a random matrix (or vector) is the matrix (or vector) of expected values of its elements.

For $X_{n\times k}$:

$$E(X) = \begin{pmatrix} E(x_{11}) & \dots & E(x_{1k}) \\ \vdots & \ddots & \vdots \\ E(x_{n1}) & \dots & E(x_{nk}) \end{pmatrix}$$

$$E\left(\begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix}\right) = \begin{pmatrix} E(x_1) \\ \vdots \\ E(x_k) \end{pmatrix}$$

Definition 4.3 (Variance-Covariance Matrix)**.** For a random vector $x_{k\times 1} = (x_1, \dots, x_k)^T$, the matrix is:

$$\text{var}(x) = \Sigma_x = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{k1} & \sigma_{k2} & \cdots & \sigma_{kk} \end{pmatrix}$$

Where:

- $\sigma_{ij} = \text{cov}(x_i, x_j) = E[(x_i - \mu_i)(x_j - \mu_j)]$
- $\sigma_{ii} = \text{var}(x_i) = E[(x_i - \mu_i)^2]$

In matrix notation:

$$\text{var}(x) = E[(x - \mu_x)(x - \mu_x)^T]$$

Note: var(x) is symmetric.

4.2.1 Derivation of Covariance Matrix Structure

Expanding the vector multiplication for variance:

$$(x - \mu_x)(x - \mu_x)' \quad \text{where } \mu_x = (\mu_1, \dots, \mu_n)'$$
$$= \begin{pmatrix} x_1 - \mu_1 \\ \vdots \\ x_n - \mu_n \end{pmatrix} (x_1 - \mu_1, \dots, x_n - \mu_n)$$

This results in the matrix $A = (a_{ij})$ where $a_{ij} = (x_i - \mu_i)(x_j - \mu_j)$. Taking expectations yields the covariance matrix elements σ_{ij}.

Definition 4.4 (Covariance Matrix (Two Vectors))**.** For random vectors $x_{k\times 1}$ and $y_{n\times 1}$, the covariance matrix is:

$$\text{cov}(x, y) = E[(x - \mu_x)(y - \mu_y)^T] = \begin{pmatrix} \text{cov}(x_1, y_1) & \cdots & \text{cov}(x_1, y_n) \\ \vdots & \ddots & \vdots \\ \text{cov}(x_k, y_1) & \cdots & \text{cov}(x_k, y_n) \end{pmatrix}$$

Note that cov$(x, x) =$ var(x).

Definition 4.5 (Correlation Matrix)**.** The correlation matrix of a random vector x is:

$$\text{corr}(x) = \begin{pmatrix} 1 & \rho_{12} & \cdots & \rho_{1k} \\ \vdots & \ddots & \vdots & \\ \rho_{k1} & \rho_{k2} & \cdots & 1 \end{pmatrix}$$

where $\rho_{ij} = \text{corr}(x_i, x_j)$.

Relationships: Let $V_x = \text{diag}(\text{var}(x_1), \dots, \text{var}(x_k))$.

$$\Sigma_x = V_x^{1/2} \rho_x V_x^{1/2} \quad \text{and} \quad \rho_x = (V_x^{1/2})^{-1} \Sigma_x (V_x^{1/2})^{-1}$$

Similarly for two vectors:

$$\Sigma_{xy} = V_x^{1/2} \rho_{xy} V_y^{1/2}$$

4.3 Properties of Mean and Variance

We can derive several key algebraic properties for operations on random vectors.

1. $E(X+Y) = E(X) + E(Y)$
2. $E(AXB) = AE(X)B$ (In particular, $E(AX) = A\mu_x$)
3. $\text{cov}(x, y) = \text{cov}(y, x)^T$
4. $\text{cov}(x+c, y+d) = \text{cov}(x, y)$
5. $\text{cov}(Ax, By) = A\text{cov}(x, y)B^T$
    - Special case for scalars: $\text{cov}(ax, by) = ab \cdot \text{cov}(x, y)$
6. $\text{cov}(x_1 + x_2, y_1) = \text{cov}(x_1, y_1) + \text{cov}(x_2, y_1)$
7. $\text{var}(x+c) = \text{var}(x)$
8. $\text{var}(Ax) = A\text{var}(x)A^T$
9. $\text{var}(x_1 + x_2) = \text{var}(x_1) + \text{cov}(x_1, x_2) + \text{cov}(x_2, x_1) + \text{var}(x_2)$
10. $\text{var}(\sum x_i) = \sum \text{var}(x_i)$ if independent.

4.3.1 Proof of Property 5 (Covariance of Linear Transformation)

$$\begin{aligned}
\text{cov}(Ax, By) &= E[(Ax - A\mu_x)(By - B\mu_y)^T] \\
&= AE[(x - \mu_x)(y - \mu_y)^T]B^T \\
&= A\text{cov}(x, y)B^T
\end{aligned}$$

4.3.2 Proof of Property 2 (Expectation of Linear Transformation)

To prove $E(AXB) = AE(X)B$: First consider $E(Ax_j)$ where x_j is a column of X.

$$E(Ax_j) = E\begin{pmatrix} a_1' x_j \\ \vdots \\ a_n' x_j \end{pmatrix} = \begin{pmatrix} E(a_1' x_j) \\ \vdots \\ E(a_n' x_j) \end{pmatrix}$$

Since a_i are constants:

$$E(a_i' x_j) = E\left(\sum_{k=1}^{p} a_{ik} x_{kj}\right) = \sum_{k=1}^{p} a_{ik} E(x_{kj}) = a_i' E(x_j)$$

Thus $E(Ax_j) = AE(x_j)$. Applying this to all columns of X:

$$E(AX) = [E(Ax_1), \dots, E(Ax_m)] = [AE(x_1), \dots, AE(x_m)] = AE(X)$$

Similarly, $E(XB) = E(X)B$.

4.3.3 Proof of Property 9 (Variance of Sum)

$$\text{var}(x_1 + x_2) = E[(x_1 + x_2 - \mu_1 - \mu_2)(x_1 + x_2 - \mu_1 - \mu_2)^T]$$

Let centered variables be denoted by differences.

$$= E[((x_1 - \mu_1) + (x_2 - \mu_2))((x_1 - \mu_1) + (x_2 - \mu_2))^T]$$

Expanding terms:

$$= E[(x_1 - \mu_1)(x_1 - \mu_1)^T + (x_1 - \mu_1)(x_2 - \mu_2)^T + (x_2 - \mu_2)(x_1 - \mu_1)^T + (x_2 - \mu_2)(x_2 - \mu_2)^T]$$

$$= \text{var}(x_1) + \text{cov}(x_1, x_2) + \text{cov}(x_2, x_1) + \text{var}(x_2)$$

4.4 The Multivariate Normal Distribution

4.4.1 Definition and Density

Definition 4.6 (Independent Standard Normal)**.** Let $z = (z_1, \dots, z_n)'$ where $z_i \sim N(0, 1)$ are independent. We say $z \sim N_n(0, I_n)$. The joint PDF is the product of marginals:

$$f(z) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} e^{-\frac{z_i^2}{2}} = \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2} z^T z}$$

Properties: $E(z) = 0$ and $\text{var}(z) = I_n$ (Covariance is 0 for $i \neq j$, Variance is 1).

Definition 4.7 (Multivariate Normal Distribution)**.** A random vector x ($n \times 1$) has a **multivariate normal distribution** if it has the same distribution as:

$$x = A_{n \times p} z_{p \times 1} + \mu_{n \times 1}$$

where $z \sim N_p(0, I_p)$, A is a matrix of constants, and μ is a vector of constants. The moments are:

- $E(x) = \mu$
- $\text{var}(x) = AA^T = \Sigma$

4.4.2 Geometric Interpretation

Using Spectral Decomposition, $\Sigma = Q \Lambda Q'$. We can view the transformation $x = Az + \mu$ as:

1. Scaling by eigenvalues ($\Lambda^{1/2}$).
2. Rotation by eigenvectors (Q).
3. Shift by mean (μ).

An Shinely App for Visualizing Bivariate Normal

Use the controls to construct the covariance matrix Σ geometrically.

We define the transformation matrix $\mathbf{A} = \mathbf{Q}□^{1/2}$, where $\mathbf{Q}$ is a rotation matrix and $□^{1/2}$ is a diagonal scaling matrix. The resulting covariance is $\Sigma = \mathbf{A}\mathbf{A}'$.

```
#| '!! shinylive warning !!': |
#|   shinylive does not work in self-contained HTML documents.
#|   Please set `embed-resources: false` in your metadata.
#| standalone: true
#| viewerHeight: 700

library(shiny)
library(bslib)
library(shinyWidgets)
library(munsell)
library(scales)
library(tibble)
library(rlang)
library(ggplot2)
library(mvtnorm)

# --- 1. PRE-GENERATE FIXED Z POINTS ---
set.seed(123)
z_fixed <- matrix(rnorm(50 * 2), ncol = 2)

ui <- page_fillable(
  theme = bs_theme(version = 5),
  withMathJax(),

  # --- ROW 1: CONTROLS (Compact Strip) ---
  card(
    class = "p-2",
    layout_columns(
      col_widths = c(3, 2, 2, 2, 2),

      div(class = "text-center", tags$label(HTML("$$\\theta$$")),
          noUiSliderInput("theta", label = NULL, min = 0, max = 360, value = 0, step = 5,
                          orientation = "horizontal", width = "100%", height = "10px", color = "#

      div(class = "text-center", tags$label(HTML("$$\\sqrt{\\lambda_1}$$")),
          noUiSliderInput("L1", label = NULL, min = 0.5, max = 3, value = 2, step = 0.1,
                          orientation = "horizontal", width = "100%", height = "10px", color = "#

      div(class = "text-center", tags$label(HTML("$$\\sqrt{\\lambda_2}$$")),
          noUiSliderInput("L2", label = NULL, min = 0.5, max = 3, value = 1, step = 0.1,
                          orientation = "horizontal", width = "100%", height = "10px", color = "#

      div(class = "text-center", tags$label(HTML("$$\\mu_1$$")),
          noUiSliderInput("mu1", label = NULL, min = -3, max = 3, value = 0, step = 0.5,
                          orientation = "horizontal", width = "100%", height = "10px", color = "#

      div(class = "text-center", tags$label(HTML("$$\\mu_2$$")),
          noUiSliderInput("mu2", label = NULL, min = -3, max = 3, value = 0, step = 0.5,
                          orientation = "horizontal", width = "100%", height = "10px", color = "#
    )
  ),

  # --- ROW 2: SIDE-BY-SIDE (Plot & Math) ---
  layout_columns(
    col_widths = c(8, 4), # 2/3 for Plot, 1/3 for Matrix

    # Left: Visualization
    card(
      full_screen = TRUE,
      plotOutput("contourPlot", height = "500px")
    ),

    # Right: The Math (Larger Font)
    card(
      class = "p-3 d-flex justify-content-center", # Center content vertically
      h5("Algebraic Representation", class = "mb-3 text-center"),

      # Use CSS to make the font larger and monospaced
      div(
        style = "font-family: 'Courier New', monospace; font-size: 1.1rem; line-height: 1.4;",
        verbatimTextOutput("matrixSide", placeholder = TRUE)
      )
    )
  )
)

server <- function(input, output) {

  data <- reactive({
    theta_rad <- input$theta * pi / 180
    Q <- matrix(c(cos(theta_rad), sin(theta_rad), -sin(theta_rad), cos(theta_rad)), 2, 2)
    Lam_sqrt <- diag(c(input$L1, input$L2))
    A <- Q %*% Lam_sqrt
    Sigma <- A %*% t(A)
    mu_vec <- c(input$mu1, input$mu2)

    x_points <- z_fixed %*% t(A)
    x_points[,1] <- x_points[,1] + mu_vec[1]
    x_points[,2] <- x_points[,2] + mu_vec[2]

    list(Q=Q, L=c(input$L1, input$L2), mu=mu_vec, Sigma=Sigma, A=A, points=as.data.frame(x_points
  })

  output$matrixSide <- renderText({
    M <- data()
    A <- round(M$A, 2)
    S <- round(M$Sigma, 2)
    rho <- cov2cor(M$Sigma)[1,2]

    # Formatted to fill vertical space comfortably
    paste0(
      "Linear Transform:\n",
      "x = A z +  \n\n",

      "Matrix A:\n",
      sprintf("[%4.1f   %4.1f]\n", A[1,1], A[1,2]),
      sprintf("[%4.1f   %4.1f]\n", A[2,1], A[2,2]),
      "\n",

      "Covariance Σ:\n",
      "(Σ = AA')\n",
      sprintf("[%4.1f   %4.1f]\n", S[1,1], S[1,2]),
      sprintf("[%4.1f   %4.1f]\n", S[2,1], S[2,2]),
      "\n",

      "Correlation:\n",
      sprintf("  = %.3f", rho)
    )
  })

  output$contourPlot <- renderPlot({
    req(data())
    M <- data()

    grid_r <- seq(-6, 6, length.out = 60)
    df_grid <- expand.grid(x = grid_r, y = grid_r)
```

```
    df_grid$z <- dmvnorm(as.matrix(df_grid), mean = M$mu, sigma = M$Sigma)

    v1 <- M$Q[,1] * M$L[1]; v2 <- M$Q[,2] * M$L[2]
    axes <- tibble(x = M$mu[1], y = M$mu[2],
                   xend1 = M$mu[1] + v1[1], yend1 = M$mu[2] + v1[2],
                   xend2 = M$mu[1] + v2[1], yend2 = M$mu[2] + v2[2])

    ggplot() +
      geom_contour_filled(data = df_grid, aes(x, y, z = z), bins = 9, show.legend = FALSE) +
      geom_point(data = M$points, aes(V1, V2), color = "black", size = 2, alpha = 0.7) +
      geom_segment(data = axes, aes(x=x, y=y, xend=xend1, yend=yend1),
                   color = "#ffc107", linewidth = 1.5, arrow = arrow(length = unit(0.3,"cm"))) +
      geom_segment(data = axes, aes(x=x, y=y, xend=xend2, yend=yend2),
                   color = "white", linewidth = 1.5, arrow = arrow(length = unit(0.3,"cm"))) +
      coord_fixed(xlim = c(-6, 6), ylim = c(-6, 6)) +
      theme_minimal() +
      labs(x = "X", y = "Y")
  })
}

shinyApp(ui, server)
```

4.4.3 Probability Density Function

If Σ is positive definite, the PDF exists. We use the change of variable formula for $x = Az + \mu$:

$$f_x(x) = f_z(g^{-1}(x)) \cdot |J|$$

where $z = A^{-1}(x - \mu)$ and $J = \det(A^{-1}) = |A|^{-1}$.

$$f_x(x) = (2\pi)^{-p/2}|A|^{-1} \exp\left\{-\frac{1}{2}(A^{-1}(x-\mu))^T(A^{-1}(x-\mu))\right\}$$

Using $|\Sigma| = |AA^T| = |A|^2$ and $\Sigma^{-1} = (AA^T)^{-1}$, we get:

$$f_x(x) = (2\pi)^{-p/2}|\Sigma|^{-1/2} \exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$

4.4.4 Moment Generating Function

Definition 4.8 (Moment Generating Function (MGF))**.** The MGF of a random vector x is $M_x(t) = E(e^{t^Tx})$. For $x = Az + \mu$:

$$M_x(t) = E[e^{t^T(Az+\mu)}] = e^{t^T\mu}E[e^{(A^Tt)^Tz}] = e^{t^T\mu}M_z(A^Tt)$$

Since $M_z(u) = e^{u^T u/2}$:

$$M_x(t) = e^{t^T \mu} \exp\left(\frac{1}{2} t^T (AA^T) t\right) = \exp\left(t^T \mu + \frac{1}{2} t^T \Sigma t\right)$$

Key Properties:

1. **Uniqueness:** Two random vectors with the same MGF have the same distribution.
2. **Independence:** y_1 and y_2 are independent iff $M_y(t) = M_{y_1}(t_1) M_{y_2}(t_2)$.

4.5 Construction and Linear Transformations

Theorem 4.1 (Constructing MVN Random Vector)**.** *Let $\mu \in \mathbb{R}^n$ and Σ be an $n \times n$ symmetric positive semi-definite (p.s.d.) matrix. Then there exists a multivariate normal distribution with mean μ and covariance Σ.*

Proof: *Since Σ is p.s.d., there exists B such that $\Sigma = BB^T$ (e.g., via Cholesky). Let $z \sim N_n(0, I)$ and define $x = Bz + \mu$.*

Theorem 4.2 (Linear Transformation Theorem)**.** *Let $x \sim N_n(\mu, \Sigma)$. Let $y = Cx + d$ where C is $r \times n$ and d is $r \times 1$. Then:*

$$y \sim N_r(C\mu + d, C\Sigma C^T)$$

Proof: *$x = Az + \mu$ where $AA^T = \Sigma$.*

$$y = C(Az + \mu) + d = (CA)z + (C\mu + d)$$

This fits the definition of MVN with mean $C\mu + d$ and variance $C\Sigma C^T$.

4.5.1 Corollaries

Corollary 4.1 (Marginals)**.** *Any subvector of a multivariate normal vector is also multivariate normal. If we partition $x = (x_1', x_2')'$, we can use $C = (I_r, 0)$ to show $x_1 \sim N(\mu_1, \Sigma_{11})$.*

Corollary 4.2 (Univariate Combinations)**.** *Any linear combination $a^T x$ is univariate normal:*

$$a^T x \sim N(a^T \mu, a^T \Sigma a)$$

Corollary 4.3 (Orthogonal Transformations)**.** *If $x \sim N(0, I_n)$ and Q is orthogonal ($Q'Q = I$), then $y = Q'x \sim N(0, I_n)$.*

Corollary 4.4 (Standardization)**.** *If $y \sim N_n(\mu, \Sigma)$ and Σ is positive definite:*

$$\Sigma^{-1/2}(y - \mu) \sim N_n(0, I_n)$$

Proof: *Let $z = \Sigma^{-1/2}(y - \mu)$. Then var$(z) = \Sigma^{-1/2} \Sigma \Sigma^{-1/2} = I_n$.*

4.6 Independence

Theorem 4.3 (Independence in MVN)**.** *Let $y \sim N(\mu, \Sigma)$ be partitioned into y_1 and y_2.*

$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

Then y_1 and y_2 are independent if and only if $\Sigma_{12} = 0$ (zero covariance).

1. Independence $\implies$ Covariance is 0: This holds generally for any distribution.

$$\text{cov}(y_1, y_2) = E[(y_1 - \mu_1)(y_2 - \mu_2)'] = 0$$

2. Covariance is 0 $\implies$ Independence: This is specific to MVN. We use MGFs. If $\Sigma_{12} = 0$, the quadratic form in the MGF splits:

$$t^T \Sigma t = t_1^T \Sigma_{11} t_1 + t_2^T \Sigma_{22} t_2$$

The MGF becomes:

$$M_y(t) = \text{exp}(t_1^T \mu_1 + \frac{1}{2} t_1^T \Sigma_{11} t_1) \times \text{exp}(t_2^T \mu_2 + \frac{1}{2} t_2^T \Sigma_{22} t_2)$$

$$M_y(t) = M_{y_1}(t_1) M_{y_2}(t_2)$$

Thus, they are independent.

4.7 Conditional Distributions

We often wish to find the distribution of a subvector y_2 given the value of another subvector y_1.

Lemma 4.1 (Constructing Independent Vectors)**.** *Let $y \sim N_n(\mu, \Sigma)$ partitioned into y_1 and y_2. Define:*

$$y_{2|1} = y_2 - \Sigma_{21} \Sigma_{11}^{-1} y_1$$

Then y_1 and $y_{2|1}$ are independent.

Proof: *Consider the linear transformation:*

$$\begin{pmatrix} y_1 \\ y_{2|1} \end{pmatrix} = Cy = \begin{pmatrix} I & 0 \\ -\Sigma_{21}\Sigma_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

The covariance matrix is $C \Sigma C'$. The off-diagonal block is:

$$cov(y_1, y_{2|1}) = \Sigma_{11}(-\Sigma_{11}^{-1}\Sigma_{12}) + \Sigma_{12} = -\Sigma_{12} + \Sigma_{12} = 0$$

Since covariance is zero, they are independent.

$$C = \begin{pmatrix} -\Sigma_{21}\Sigma_{11}^{-1} & I_{n-p} \end{pmatrix}'$$

$$C \cdot \Sigma \cdot C'$$

$$= \begin{pmatrix} I_p & 0 \\ -\Sigma_{21}\Sigma_{11}^{-1} & I_{n-p} \end{pmatrix} \cdot \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \cdot C'$$

$$= \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ 0 & \Sigma_{22} - \Sigma_{12}\Sigma_{11}^{-1}\Sigma_{12} \end{pmatrix} \begin{pmatrix} I_p & -\Sigma_{11}^{-1}\Sigma_{1} \\ 0 & I_{n-p} \end{pmatrix}$$

$$= \begin{pmatrix} \Sigma_{11} & 0 \end{pmatrix}$$

Theorem 4.4 (Conditional Distribution Theorem). *The conditional distribution of y_2 given y_1 is multivariate normal:*

$$y_2|y_1 \sim N(\mu_{2|1}, \Sigma_{22|1})$$

where:

- ***Conditional Mean:*** $E(y_2|y_1) = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_1 - \mu_1)$
- ***Conditional Variance:*** $var(y_2|y_1) = \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$

Proof: *Write $y_2 = y_{2|1} + \Sigma_{21}\Sigma_{11}^{-1}y_1$. Conditional on y_1, the term $\Sigma_{21}\Sigma_{11}^{-1}y_1$ is constant. Since $y_{2|1}$ is independent of y_1, its conditional distribution is simply its marginal distribution. Thus, the mean shifts by the constant term, and the variance remains that of $y_{2|1}$ (the Schur complement).*

4.7.1 Numerical Example

Let $\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \sim N\left(\begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ 1 & 4 \end{pmatrix}\right)$. Find the distribution of $y_1|y_2$.

- $\mu_{1|2} = \mu_1 + \sigma_{12}\sigma_{22}^{-1}(y_2 - \mu_2) = 1 + 1(4)^{-1}(y_2 - 2) = 0.5 + 0.25y_2$
- $\sigma^2_{1|2} = \sigma_{11} - \sigma_{12}\sigma_{22}^{-1}\sigma_{21} = 2 - 1(1/4)1 = 1.75$

So $y_1|y_2 \sim N(0.5 + 0.25y_2, 1.75)$.

4.7.2 Variance Decomposition

The Law of Total Variance states:

$$\text{var}(y_2) = E[\text{var}(y_2|y_1)] + \text{var}[E(y_2|y_1)]$$

In the MVN case:

- $E[\text{var}(y_2|y_1)] = \Sigma_{22|1}$ (constant variance).
- $\text{var}[E(y_2|y_1)] = \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$ (explained variance). Summing these returns the total variance Σ_{22}.

4.8 Partial and Multiple Correlation

Definition 4.9 (Partial Correlation)**.** The partial correlation between elements y_i and y_j given a set of variables x is derived from the conditional covariance matrix $\Sigma_{y|x}$:

$$\rho_{ij|x} = \frac{\sigma_{ij|x}}{\sqrt{\sigma_{ii|x}\sigma_{jj|x}}}$$

where $\sigma_{ij|x}$ are elements of $\Sigma_{y|x} = \Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}$.

Definition 4.10 (Multiple Correlation (R^2))**.** For a scalar y and vector x, the squared multiple correlation is the proportion of variance of y explained by the conditional mean:

$$\rho_{y|x}^2 = \frac{\text{var}(E(y|x))}{\text{var}(y)} = \frac{\Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}}{\sigma_{yy}}$$