SPECTRAL ANALYSIS OF MULTISCALE SAMPLE COVARIANCE MATRICES

AN HONORS THESIS

SUBMITTED ON THE SECOND DAY OF MAY, 2025

TO THE DEPARTMENT OF MATHEMATICS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

OF THE HONORS PROGRAM

OF NEWCOMB-TULANE COLLEGE

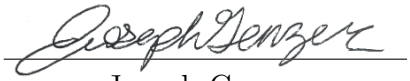
TULANE UNIVERSITY

FOR THE DEGREE OF

BACHELOR OF SCIENCE

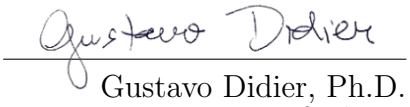
WITH HONORS IN MATHEMATICS

BY



Joseph Genzer

APPROVED:



Gustavo Didier, Ph.D.
Director of Thesis

Ken McLaughlin, Ph.D.
Second Reader

Abstract

Random matrix theory, originally developed in the 1950s, provides a powerful framework for modeling high-dimensional systems, revealing universal behaviors with applications in physics, probability, and statistics. This thesis examines the spectral properties of sample covariance matrices, fundamental objects in multivariate analysis, across both low- and high-dimensional regimes. We first revisit foundational results from classical probability and random matrix theory, including Gaussian and GOE-type asymptotics in low dimensions, and the emergence of the Marchenko-Pastur and Tracy-Widom laws in high dimensions. We then extend the analysis to multiscale sample covariance matrices, introducing dependence through processes such as fractional Brownian motion. Our results show that dependence fundamentally reshapes the spectral behavior of sample covariance matrices. In certain regimes, the top eigenvalues become asymptotically independent and Gaussian; in others, they exhibit level repulsion characteristic of β-Hermite ensembles. Through a combination of theoretical results and Monte Carlo simulations, we characterize new universality classes driven by the interplay between dimensionality and dependence.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Professor Didier, for his invaluable mentorship throughout the course of this thesis. His guidance deepened my understanding of probability theory, stochastic processes, and random matrix theory, while challenging me to integrate rigorous analysis with computational simulation. I am especially grateful for his patience and insights on the writing process of mathematical research. I also wish to thank Professor McLaughlin for serving as my second reader and for his insightful suggestions on the β-Hermite ensemble analysis. Their encouragement, high standards, and deep engagement with the material have left a lasting impact, for which I am deeply thankful.

Contents

1 Introduction

The spectral analysis of random matrices is a central theme in probability theory, with broad applications in physics, statistics, and data science. A fundamental example is the sample covariance matrix, whose eigenvalues describe the distribution of variance along principal components. The classical sample covariance matrix, denoted $\hat{\Sigma}_n$, is constructed from n independent samples of a p-dimensional random vector using centered outer products.

In the low-dimensional regime, where p is fixed and $n \to \infty$, the eigenvalues of $\hat{\Sigma}_n$ exhibit two key limiting behaviors: the top eigenvalue is asymptotically Gaussian, and after appropriate scaling, $\hat{\Sigma}_n$ converges in distribution to a Gaussian Orthogonal Ensemble (GOE), exhibiting characteristic eigenvalue repulsion. By contrast, if we instead consider the two-way limit $n, p \to \infty$ with $\frac{p}{n} \to c > 0$, we enter the high-dimensional regime, where the bulk eigenvalue distribution converges to the Marchenko–Pastur law, while the top eigenvalue distribution converges to the Tracy–Widom law.

However, in real-world applications, true independence of measurements is rare and often an idealization. To introduce dependence into our analysis, we consider fractional Brownian motion (fBm), a Gaussian process whose covariance structure is controlled by the Hurst parameter $H \in (0, 1)$. For $H = 0.5$, the process reduces to standard Brownian motion with independent increments, while for $H \neq 0.5$, either anti-persistence or long-range dependence emerges. To extend the traditional $\hat{\Sigma}_n$ framework to settings with dependence, such as multivariate fBm, we introduce the multiscale sample covariance matrix, denoted $M(h)$, constructed from outer products of scaled, non-overlapping increments at a fixed lag $h > 0$. The multiscale sample covariance matrix enables the analysis of variance structures across scales.

The main results of this thesis concern the spectral properties of $M(h)$ in both low and high dimensions. In the low-dimensional, multiscale regime, where $n, h \to \infty$ with $\frac{h}{n} = O(1)$ and a fixed p, the growth of both the diagonal entries and eigenvalues follows a power law in h. Using log-log regression across lag values, we find that while both diagonal entries scale as h^{2H_2}, the eigenvalues exhibit Hurst-based scaling; specifically, $\lambda_1 \overset{\mathbb{P}}{\sim} h^{2H_1}$ and $\lambda_2 \overset{\mathbb{P}}{\sim} h^{2H_2}$. Thus, while the diagonal entries of $M(h)$ alone cannot recover the full underlying dependence structure, the eigendomain is able to do so.

In the high-dimensional, multiscale regime, we consider the three-way limit $n, p, h \to \infty$ with $\frac{ph}{n} = O(1)$. In this setting, we model the observations as $Y(t) = PX(t) + Z(t)$, where $X(t)$ is a bivariate fBm process with independent components, P is a projection matrix, and $Z(t)$ is Gaussian noise. Depending on the Hurst parameters, the top two eigenvalues of $M(h)$ exhibit distinct asymptotic behaviors. When $0 < H_1 < H_2 < 0.75$, they are asymptotically independent and jointly Gaussian, and when $0 < H_1 = H_2 < 0.75$, they follow a 2×2 GOE distribution, exhibiting eigenvalue repulsion. A critical threshold at $H = 0.75$, arising from the squared summability constraint on the autocovariance function, marks a phase transition in the spectral behavior of $M(h)$, beyond which asymptotic Gaussianity no longer holds. When $0.75 < H_1 < H_2 < 1$, the top two eigenvalues are asymptotically independent, and when $0.75 < H_1 = H_2 < 1$, they exhibit non-GOE-like repulsion, best modeled by a β-Hermite ensemble with $\beta \approx 0.4$.

Thus, while fixed-scale analysis exhibits Gaussian and GOE-like asymptotics in low dimensions and Marchenko–Pastur and Tracy–Widom laws in high dimensions, the introduction of multiscale dependence fundamentally reshapes spectral behavior. This thesis reveals new universality classes for the eigenvalues of multiscale sample covariance matrices, driven by changes in dimensionality and dependence.

2 Foundations of Probability Theory

To lay the mathematical foundations for our analysis, we begin by reviewing key concepts in probability theory. Let Ω denote the sample space, representing all possible outcomes of a random experiment.

Definition 2.1. A *random variable* X is a function $X : \Omega \to \mathbb{R}$ that assigns a real number to each outcome $\omega \in \Omega$.

Definition 2.2. The *cumulative distribution function* (c.d.f.) of a random variable X, denoted by $F_X(x)$, is defined by $F_X(x) = \mathbb{P}(X \leqslant x)$ for all $x \in \mathbb{R}$.

Proposition 2.1. *Let $F_X(x)$ be a c.d.f. of a random variable X. Then, it can be shown that*

(F1) $\lim_{x\to-\infty} F_X(x) = 0, \lim_{x\to+\infty} F_X(x) = 1,$

(F2) $F_X(x)$ *is right-continuous:* $\lim_{x\to a^+} F_X(x) = F_X(a),$

(F3) $F_X(x)$ *is monotonic non-decreasing: If* $a < b$*, then* $F_X(a) \leqslant F_X(b)$.

Conversely, if there exists a function F that satisfies criteria (F1)-(F3), then it can be show that there exists a random variable X such that $F(x) = \mathbb{P}(X \leqslant x), x \in \mathbb{R}$ (Casella and Berger, 2002, p.31). In other words, properties (F1)-(F3) mathematically *characterize* c.d.f.s.

2.1 Types of Random Variables

Random variables can be classified based on their behavior. This is broadly established in the so-called *Lebesgue decomposition theorem* (Folland, 1999, p.91). Hereinafter, we focus on the main types, namely, discrete and absolutely continuous random variables.

Definition 2.3. A random variable X is called *discrete* if there exists a countable set $\{x_i\}_{i\in\mathbb{R}}$ such that $\mathbb{P}(X \in \{x_i\}_{i\in\mathbb{R}}) = 1$.

It is easy to see that the c.d.f. of a discrete random variable X can be expressed by $F(x) = \sum_{x_i \leqslant x} \mathbb{P}(X = x_i)$ for all $x \in \mathbb{R}$. Also, the set $S = \{x_i\}_{i\in\mathbb{R}}$ is called its *support*.

The following definition introduces a concept that is useful in the study of discrete random variables.

Definition 2.4. Let X be a discrete random variable with support $S = \{x_i\}_{i\in\mathbb{N}}$. The *probability mass function* (p.m.f.) of X is given by $f_X(x_i) = \mathbb{P}(X = x_i)$, $i \in \mathbb{N}$.

We now turn to the other main type of random variable.

Definition 2.5. Let X be a random variable with c.d.f. F_X. We say X is *absolutely continuous* (a.c.) when there exists a nonnegative function f_X such that

$$F_X(x) = \int_{-\infty}^{x} f_X(t)dt, \quad x \in \mathbb{R}.$$

In this case, the function f_X is called the *probability density function* (p.d.f.) of X.

Note that, in the framework of a.c. random variables, the p.d.f. plays a role that is analogous to that of the p.m.f. for discrete random variables. Also, for an a.c. random variable X, the support is given by $S = \overline{\{x \in \mathbb{R} \mid f(x) > 0\}}$, where the bar denotes the topological closure.

Some examples of probability distributions and their p.m.f.s or p.d.f.s include the following.

Example 2.1. A random variable X has a *binomial distribution* with parameters $n \in \mathbb{N}$ and $0 \leqslant p \leqslant 1$ (denoted $X \sim \text{Bin}(n, p)$) if its p.m.f. is

$$f_X(x) = \binom{n}{x} p^x (1-p)^{n-x}, \quad x = 0, 1, \ldots, n.$$

Example 2.2. A random variable X has a *Poisson distribution* with parameter $\lambda \in \mathbb{R}^+$ (denoted $X \sim \text{Poisson}(\lambda)$) if its p.m.f. is

$$f_X(x) = \frac{e^{-\lambda}\lambda^x}{x!}, \quad x = 0, 1, \ldots$$

Example 2.3. A random variable X has a *normal distribution* with parameters $\mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}^+$ (denoted $X \sim \mathcal{N}(\mu, \sigma^2)$) if its p.d.f. is

$$f_X(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \quad x \in \mathbb{R}.$$

Example 2.4. A random variable X has a *Gamma distribution* with parameters $a, \lambda \in \mathbb{R}^+$ (denoted $X \sim \Gamma(a, \lambda)$) if its p.d.f. is

$$f_X(x) = \frac{\lambda e^{-\lambda x}(\lambda x)^{a-1}}{\Gamma(a)}, \quad x \in (0, \infty),$$

where $\Gamma(a) = \int_0^\infty x^{a-1}e^{-x}dx$ is the gamma function.

2.2 Law of Large Numbers

The two most basic universality results in probability theory are the law of large numbers and the central limit theorem. Hereinafter, we focus on the former. For this purpose, we first need to cover two basic probabilistic notions of convergence.

Definition 2.6. A sequence of random variables X_n is said to *converge in probability* to a random variable X if, $\forall \epsilon > 0$, $\lim_{n\to\infty} \mathbb{P}(|X_n - X| \geqslant \epsilon) = 0$.

Definition 2.7. A sequence of random variables X_n is said to *converge almost surely* (a.s.) to a random variable X if $\mathbb{P}\big(\lim_{n\to\infty} X_n = X\big) = 1$.

For a sequence of random variables $\{X_n\}_{n\in\mathbb{N}}$ and a limiting random variable X, convergence a.s. implies convergence in probability (Casella and Berger, 2002, p.235). Nevertheless, the converse does not hold, as illustrated in the following example.

Example 2.5. Let $\{X_n\}_{n\in\mathbb{N}}$ be a sequence of random variables that can only take on values of 0 or 1. In addition, suppose that

$$\mathbb{P}(X_n = x) = \begin{cases} 1 - \frac{1}{n} & \text{if } x = 0, \\ \frac{1}{n} & \text{if } x = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Convergence in probability: For any $\epsilon > 0$, as $n \to \infty$, $\mathbb{P}(|X_n - 0| \geqslant \epsilon) = \mathbb{P}(X_n = 1) = \frac{1}{n} \to 0$. Thus, $\{X_n\}$ converges in probability to 0.

Failure to converge a.s: Since $\sum_{n=1}^\infty \mathbb{P}(X_n = 1) = \sum_{n=1}^\infty \frac{1}{n} = \infty$, the Borel-Cantelli Lemma, as given in A.1, implies that $\mathbb{P}(X_n = 1 \text{ i.o.}) = 1$. This violates the condition for a.s. convergence, which requires that the probability of the sequence not converging to the limit to be 0. Thus, $\{X_n\}$ does not converge a.s. to 0.

Proposition 2.2. *Moreover, if $f : \mathbb{R} \to \mathbb{R}$ is a continuous function, then $X_n \xrightarrow{a.s.} X$ implies $f(X_n) \xrightarrow{a.s.} f(X)$. Likewise , $X_n \xrightarrow{\mathbb{P}} X$ implies $f(X_n) \xrightarrow{\mathbb{P}} f(X)$*

Proof. We will first prove that $X_n \xrightarrow{a.s.} X$ implies $f(X_n) \xrightarrow{a.s.} f(X)$. Let $X_n \xrightarrow{a.s.} X$. This means that there exists $\Omega_0 \subseteq \Omega$, $\mathbb{P}(\Omega_0) = 1$, such that for every $\omega \in \Omega_0$, $X_n(\omega) \to X(\omega)$ as $n \to \infty$. Since f is continuous, for every $\omega \in \Omega_0$,

$$X_n(\omega) \to X(\omega) \implies f(X_n(\omega)) \to f(X(\omega)).$$

Therefore, $f(X_n(\omega)) \to f(X(\omega))$ for every $\omega \in \Omega_0$. This implies that $f(X_n) \xrightarrow{a.s.} f(X)$.
For the proof that $X_n \xrightarrow{\mathbb{P}} X$ implies $f(X_n) \xrightarrow{\mathbb{P}} f(X)$, see (Casella and Berger, 2002, p.233). □

We next lay the groundwork for the law of large numbers by establishing key probabilistic inequalities.

Theorem 2.1. *Markov's Inequality: For a non-negative random variable X and any $a > 0$,*

$$\mathbb{P}(X \geqslant a) \leqslant \frac{\mathbb{E}X}{a}. \tag{2.1}$$

Proof. (Casella and Berger, 2002, p.136) □

The following classical inequality is an immediate consequence of Theorem 2.1.

Theorem 2.2. *Chebyshev's Inequality: Let X be a random variable with finite second moment. Let μ and σ^2 be its mean and variance, respectively. Then, for any $k > 0$,*

$$\mathbb{P}(|X - \mu| \geqslant k\sigma) \leqslant \frac{1}{k^2}. \tag{2.2}$$

Proof. For X as in the statement, define a new random variable $Y = (X - \mu)^2$. Note that Y is non-negative, and that $\mathbb{E}Y = \mathbb{E}[(X - \mu)^2] = \sigma^2$, the variance of X.
Then fix $k > 0$. By applying Markov's Inequality to Y for $a = k^2\sigma^2$,

$$\mathbb{P}(Y \geqslant k^2\sigma^2) \leqslant \frac{\mathbb{E}[Y]}{k^2\sigma^2}$$

Substituting $Y = (X - \mu)^2$ and $\mathbb{E}[Y] = \sigma^2$, we get:

$$\mathbb{P}((X - \mu)^2 \geqslant k^2\sigma^2) \leqslant \frac{\sigma^2}{k^2\sigma^2}$$

Simplifying, we obtain Chebyshev's Inequality. □

We are now in a position to state and prove a weak law of large numbers.

Theorem 2.3. *Chebyshev's Weak Law of Large Numbers: Let $X_1, X_2, \ldots, X_n$ be i.i.d. random variables with mean μ and finite variance σ^2. Then, $\bar{X}_n \xrightarrow{\mathbb{P}} \mu$ as $n \to \infty$; that is,*

$$\forall \epsilon > 0, \lim_{n\to\infty} \mathbb{P}(|\bar{X}_n - \mu| \geqslant \epsilon) = 0 \tag{2.3}$$

Proof. First, note the properties of expectation and variance for the sample mean of i.i.d. random variables, i.e., $\mathbb{E}[\overline{X}_n] = \mu$ and $\mathrm{Var}(\overline{X}_n) = \frac{\sigma^2}{n}$. Applying Chebyshev's Inequality to $\bar{X}_n$, for any $\epsilon > 0$, we have

$$\mathbb{P}(|\bar{X}_n - \mu| \geqslant \epsilon) \leqslant \frac{\mathrm{Var}(\bar{X}_n)}{\epsilon^2} = \frac{\sigma^2}{n\epsilon^2}$$

As $n \to \infty$, $\frac{\sigma^2}{n\epsilon^2} \to 0$, which implies $\lim_{n\to\infty} \mathbb{P}(|\bar{X}_n - \mu| \geqslant \epsilon) = 0$. □

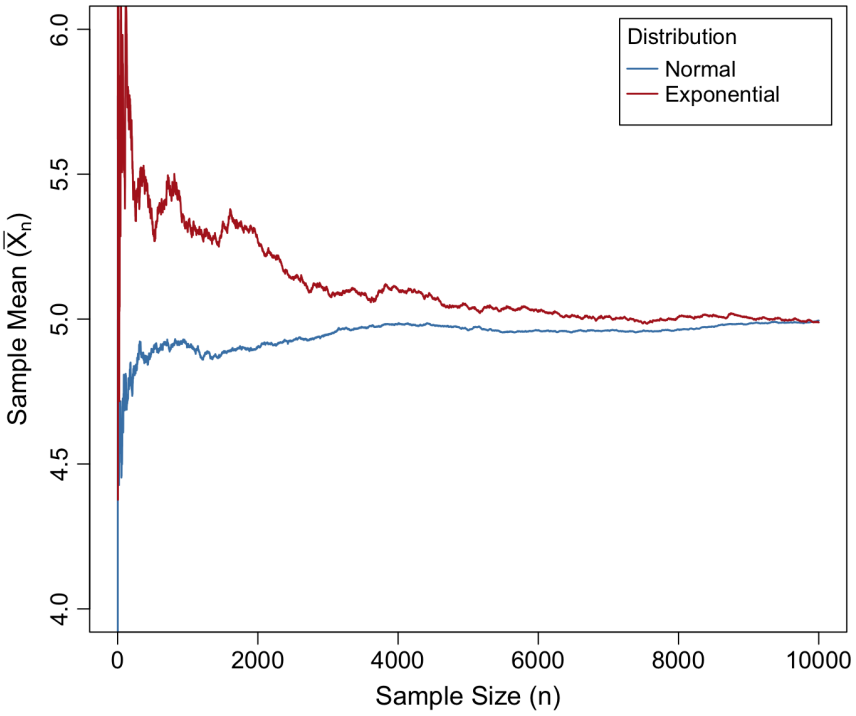


Figure 1: **Chebyshev's Weak Law of Large Numbers**, generated using the code in listing 1, with $n = 10,000$ samples from normal and exponential distributions, both with mean 5 and variance 4. As the sample size increases, the sample means converge to the theoretical mean for both distributions.

The statement of Theorem 2.3 can be significantly strengthened, though at the cost of some heavy technicalities. The so-called *Kolmogorov's Strong Law of Large Numbers* states that under the same assumptions (as in Chebychev's WLLN), $\bar{X}_n$ converges a.s. to $\mathbb{E}X_1$.

2.3 Moments, Central Moments, and Generating Functions

Moments are fundamental in characterizing the shape and properties of distributions, quantifying key statistics such as mean, variance, skewness, and kurtosis. This section outlines the mathematical foundations of moments, central moments, and moment generating functions.

Definition 2.8. For each integer j, the j-th *moment* of X is given by $\mu_j = \mathbb{E}X^j$.

Definition 2.9. The n-th *central moment* of X is given by $\mathbb{E}(X-\mu)^j$, where $\mu = \mathbb{E}X$. The second central moment is also called the *variance of* X, denoted $\mathrm{Var}(X)$.

Definition 2.10. Let X be a random variable with c.d.f. F_X. The *moment generating function* (m.g.f.) of X, denoted by $M_X(t)$, is given by $M_X(t) = \mathbb{E}e^{tX}$ for all $t \in \mathbb{R}$ where this expression exists.

The j-th derivative of $M_X(t)$ evaluated at $t = 0$ yields the j-th moment of X. That is,

$$\frac{d^j}{dt^j} M_X(t)\Big|_{t=0} = \mathbb{E}[X^j].$$

We now present the key moments and moment generating functions for each of the distributions introduced above.

Example 2.6. In the case of Example 2.1,

$$M_X(t) = \sum_{x=0}^{n} e^{tx} \binom{n}{x} p^x (1-p)^{(n-x)} = (pe^t + (1-p))^n, \quad t \in \mathbb{R}.$$

In particular, $\mathbb{E}X = np$ and $\text{Var}(X) = np(1-p)$.

Example 2.7. In the case of Example 2.2,

$$M_X(t) = \sum_{x=0}^{n} e^{tx} \frac{e^{-\lambda}\lambda^x}{x!} = e^{\lambda(e^t-1)}, \quad t \in \mathbb{R}.$$

In particular, $\mathbb{E}X = \lambda$ and $\text{Var}(X) = \lambda$.

Example 2.8. In the case of Example 2.3,

$$M_X(t) = \int_{-\infty}^{\infty} e^{tx} \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)} dx = e^{\mu t + \frac{\sigma^2 t^2}{2}}, \quad t \in \mathbb{R}.$$

In particular, $\mathbb{E}X = \mu$ and $\text{Var}(X) = \sigma^2$.

Example 2.9. In the case of Example 2.4,

$$M_X(t) = \int_0^{\infty} e^{tx} \frac{\lambda e^{-\lambda x}(\lambda x)^{a-1}}{\Gamma(a)} dx = (\frac{1}{1-t/\lambda})^a, \quad t \in (-\infty, \lambda).$$

In particular, $\mathbb{E}X = \frac{a}{\lambda}$ and $\text{Var}(X) = \frac{a}{\lambda^2}$.

2.4 Multivariate Distributions and Random Vectors

Random vectors are central to the analysis of multivariate distributions. This section introduces their formal definition, distinguishes between discrete and continuous cases, and presents the multivariate normal distribution, with particular emphasis on the bivariate case.

Definition 2.11. A *random vector* $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ is a vector valued function that maps outcomes in a sample space to n-dimensional vectors in $\mathbb{R}^n$, where each X_i is a random variable.

Definition 2.12. A random vector $\mathbf{X}$ is discrete if each component $\mathbf{X}_i$ takes values from a countable set.

Definition 2.13. An random vector $\mathbf{X}$ is absolutely continuous if there exists a non-negative function $f(\mathbf{x})$, known as the *density function*, such that $\mathbb{P}(\mathbf{X} \in A) = \int_A f(\mathbf{x})d\mathbf{x}$ for any Borel set $A \subseteq \mathbb{R}^n$.

Definition 2.14. A random vector $\mathbf{X} \in \mathbb{R}^n$ follows a *multivariate normal distribution* if its p.d.f. is given by

$$f(\mathbf{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right),$$

where $\boldsymbol{\mu} \in \mathbb{R}^n$ is the mean vector and Σ is a positive definite covariance matrix.

Example 2.10. The *bivariate normal distribution*, defined for $\mathbf{X} = (X_1, X_2)^\top \in \mathbb{R}^2$, describes the joint distribution of two random variables, and has a joint pdf given by

$$f(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} - \frac{2\rho(x_1-\mu_1)(x_2-\mu_2)}{\sigma_1\sigma_2}\right]\right),$$

where μ_1 and μ_2 are the means, σ_1 and σ_2 are the standard deviations, and $\rho = \frac{\text{Cov}(X_1,X_2)}{\sigma_1\sigma_2}$ is the correlation coefficient.

Proposition 2.3. *Let $F_{\mathbf{X}}$ be the c.d.f. of a random vector. Fix $i \in \{1, \ldots, n\}$. Then it can be shown that:*

(F1)

$$\lim_{x_i \to -\infty} F_{\mathbf{X}}(x_1, \ldots, x_i, \ldots, x_n) = 0, \quad \lim_{x_i \to \infty} F_{\mathbf{X}}(x_1, \ldots, x_i, \ldots, x_n) = F_{\mathbf{X}}(x_1, \ldots, \overline{x_i}, \ldots, x_n).$$

(F2) $F_{\mathbf{X}}$ *is entry-wise right-continuous:*

$$y_i \downarrow x_i \Rightarrow F_{\mathbf{X}}(x_1, \ldots, y_i, \ldots, x_n) \leqslant F_{\mathbf{X}}(x_1, \ldots, x_i, \ldots, x_n).$$

(F3) $F_{\mathbf{X}}$ *is entry-wise non-decreasing:*

$$x_i \leqslant y_i \Rightarrow F_{\mathbf{X}}(x_1, \ldots, x_i, \ldots, x_n) \leqslant F_{\mathbf{X}}(x_1, \ldots, y_i, \ldots, x_n).$$

(F4) *For any intervals* $I_k = (a_k, b_k]$, $k = 1, \ldots, n$,

$$\Delta_{1,I_1} \ldots \Delta_{n,I_n} F_{\mathbf{X}}(x_1, \ldots, x_n) \geqslant 0,$$

where $\Delta_{k,I_k} g(x) = g(x_1, \ldots, b_k, \ldots, x_n) - g(x_1, \ldots, a_k, \ldots, x_n)$.

Conversely, for any $n \in \mathbb{N}$, if there exists a function $F : \mathbb{R}^n \to \mathbb{R}$ satisfying properties (F1) - (F4), it is called an n-variate c.d.f. Properties (F1) - (F3) are analogous to the respective properties in the univariate case, but (F4) is an additional property that ensures non-negativity in $\mathbb{R}^n$.

3 Sample Covariance Matrices

Covariance matrices are fundamental in understanding the relationships between multiple random variables within a joint distribution. This section outlines the theoretical foundations for both population and sample covariance matrices, exploring properties such as symmetry and definiteness.

Definition 3.1. Given a distribution with p variables (not necessarily i.i.d.) with finite second moments, the *covariance matrix* Σ is defined as

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_p^2 \end{bmatrix}, \tag{3.1}$$

where σ_i^2 is the variance of the ith variable, and σ_{ij} is the covariance between the ith and jth variables, defined as $\sigma_{ij} = \mathrm{Cov}(X_i, X_j) = \mathbb{E}[(X_i - \mu_i)(X_j - \mu_j)]$.

Definition 3.2. Let $\mathbf{X_1}, \ldots, \mathbf{X_n}$ be a sample of random vectors. The *sample covariance matrix* $\hat{\Sigma}$ is defined as

$$\hat{\Sigma}_n = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^T \tag{3.2}$$

At this point, it is useful to recap the Leibniz formula for the determinant, which is given by

$$\det(A) = \begin{vmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{vmatrix} = \sum_{\sigma \in S_n} \left(\mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{i,\sigma(i)} \right). \tag{3.3}$$

Note that, by (3.3), $\det(A)$ is a continuous function of the matrix argument A.

Proposition 3.1. *The following propositions establish the convergence of the roots of the characteristic polynomial in the deterministic and random cases, respectively.*

(*i*) *If* $\Sigma_n \to \Sigma_0$ *elementwise, then the roots of the characteristic polynomial converge.*

(*ii*) *If* $\Sigma_n \xrightarrow{a.s.} \Sigma_0$ *in norm, then the roots of the characteristic polynomial converge almost surely.*

(*iii*) *If* $\Sigma_n \xrightarrow{\mathbb{P}} \Sigma_0$ *in norm, then the roots of the characteristic polynomial converge in probability.*

Proof. Let $p(\Sigma, \lambda) = \det(\Sigma - \lambda I)$, $\lambda \in \mathbb{R}$, denote the characteristic polynomial of a matrix Σ, where the roots of $p(\Sigma, \lambda)$ are the eigenvalues of Σ.

(*i*) Let Σ_n and Σ_0 be $p \times p$ matrices. Now note that $\Sigma_n \to \Sigma_0$ elementwise if and only if $\lim_{n\to\infty} \|\Sigma_n - \Sigma_0\|_F = 0$, where $\|\cdot\|_F$ denotes the Frobenius norm. Therefore, elementwise convergence of Σ_n to Σ_0 is equivalent to $\lim_{n\to\infty} ||\Sigma_n - \Sigma_0|| = 0$, where $\|\cdot\|$ denotes any matrix norm. By (3.3), the coefficients of the characteristic polynomial $p(\Sigma, \cdot)$ depend continuously on the entries of Σ. Also, the roots of any polynomial are a continuous function of the coefficients of the polynomial in question. Therefore, if $\Sigma_n \to \Sigma_0$ elementwise as $n \to \infty$, then the p roots of $p(\Sigma_n, \cdot)$ also converge to those of $p(\Sigma_0, \cdot)$.

(*ii*) First recall that all norms on a finite-dimensional vector space are equivalent. In other words, there exist constants $c, C > 0$ such that, for any matrix M in the space of $p \times p$ matrices $\mathcal{M}(p, \mathbb{R})$, we have

$$c\|M\|_1 \leqslant \|M\|_2 \leqslant C\|M\|_1,$$

where $\|\cdot\|_1$ and $\|\cdot\|_2$ are two different matrix norms. Consequently, convergence in one matrix norm implies convergence in any other matrix norm. Assume $\Sigma_n \xrightarrow{a.s.} \Sigma_0$ in norm, where Σ_n and Σ_0 are $p \times p$ random matrices. This almost sure convergence implies that there exists $\Omega_0 \subseteq \Omega$ such that $\mathbb{P}(\Omega_0) = 1$ and for every $\omega \in \Omega_0$ we have $\Sigma_n(\omega) \to \Sigma_0(\omega)$ elementwise, by the equivalence of matrix norms. Thus, as a consequence of part (*i*) in this proposition, $\Sigma_n \xrightarrow{a.s.} \Sigma_0$ implies $p(\Sigma_n(\omega), \cdot) \xrightarrow{a.s.} p(\Sigma_0(\omega), \cdot)$, where $p(\Sigma, \cdot)$ denotes the characteristic polynomial whose roots are the eigenvalues.

(*iii*) Assume $\Sigma_n \xrightarrow{\mathbb{P}} \Sigma_0$ in norm, where Σ_n and Σ_0 are $p \times p$ random matrices. By the definition of convergence in probability, for any $\epsilon > 0$ and any $\delta > 0$, there exists $N \in \mathbb{N}$ such that for all $n \geqslant N$,

$$\mathbb{P}\left(\|\Sigma_n - \Sigma_0\| \geqslant \epsilon\right) < \delta.$$

Because the roots of the characteristic polynomial are continuous functions of the matrix entries, there exists $\epsilon' > 0$ such that

$$\mathbb{P}\left(\max_{1 \leqslant i \leqslant p} |\lambda_i(\Sigma_n) - \lambda_i(\Sigma_0)| \geqslant \epsilon'\right) < \delta.$$

Therefore, the probability that the roots of the characteristic polynomial differ by at least ϵ' converges to zero as $n \to \infty$. Hence,

$$\Sigma_n \xrightarrow{\mathbb{P}} \Sigma_0 \implies \lambda_i(\Sigma_n) \xrightarrow{\mathbb{P}} \lambda_i(\Sigma_0) \text{ for } i = 1, \ldots, p,$$

which shows that the roots of the characteristic polynomial converge in probability. □

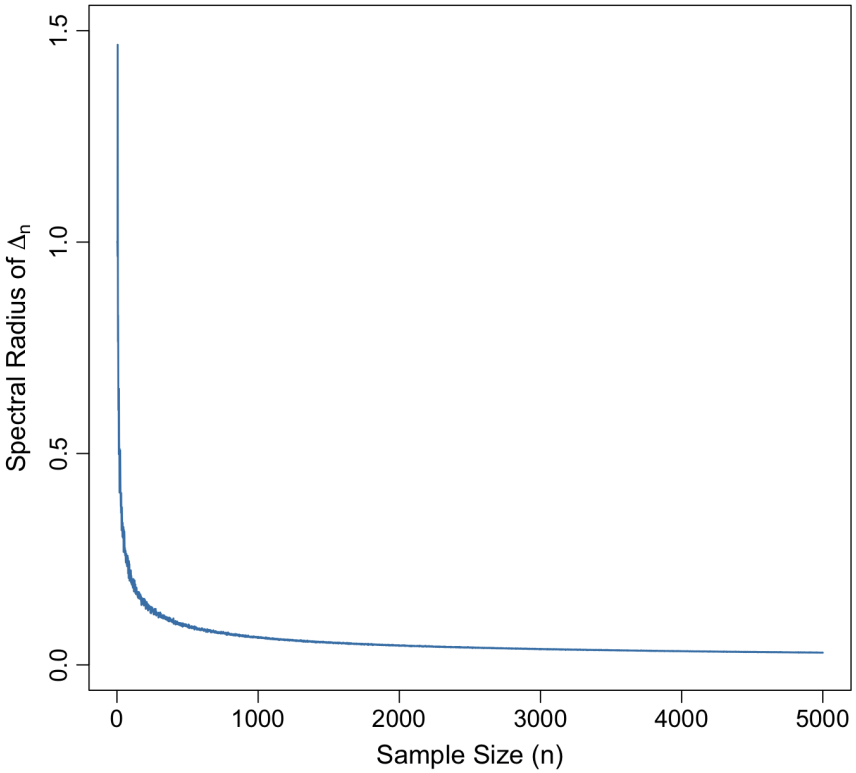


Figure 2: **Convergence of** $\hat{\Sigma}_n$, generated using the code in listing 2, with $p = 10$ and independent standard normal samples. As n increases, the spectral radius of $\Delta_n = \hat{\Sigma}_n - \Sigma_n$ vanishes, confirming that $\hat{\Sigma}_n$ converges entry-wise to $\Sigma_n = I_p$ and that its eigenvalues converge to 1.

In the next proposition, we lay out a few basic properties of sample covariance matrices.

Proposition 3.2. *For $n \in \mathbb{N}$, let $\hat{\Sigma}_n$ be as in* (3.2)*. Then the following claims hold:*

(*i*) *$\hat{\Sigma}_n$ is a symmetric matrix a.s.*

(*ii*) *$\hat{\Sigma}_n$ is positive semidefinite a.s., meaning all eigenvalues of $\hat{\Sigma}_n$ are non-negative.*

(*iii*) *Assuming $\mathbf{X_1}, \ldots, \mathbf{X_n}$ as in* (3.2) *are i.i.d. and $\mathbf{X_i} \sim N(\mathbf{0}, I_p)$, then $\hat{\Sigma}_n$ is positive definite a.s.*

Proof. (i) The transpose of $\hat{\Sigma}_n$ is given by

$$\hat{\Sigma}_n^\top = \left(\frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{X_i} - \bar{\mathbf{X}})(\mathbf{X_i} - \bar{\mathbf{X}})^\top \right)^\top = \frac{1}{n-1} \sum_{i=1}^{n} ((\mathbf{X_i} - \bar{\mathbf{X}})^\top)^\top (\mathbf{X_i} - \bar{\mathbf{X}})^\top = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{X_i} - \bar{\mathbf{X}})(\mathbf{X_i} - \bar{\mathbf{X}})^\top.$$

Thus, $\hat{\Sigma}_n^\top = \hat{\Sigma}_n$, proving symmetry.

(ii) A symmetric matrix A is positive semidefinite if and only if, for any non-zero vector $\mathbf{z}$, $\mathbf{z}^\top A \mathbf{z} \geqslant 0$. So, for any non-zero $\mathbf{z}$, consider the quadratic form

$$z^\top \hat{\Sigma}_n \mathbf{z} = \mathbf{z}^\top \left(\frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{X_i} - \bar{\mathbf{X}})(\mathbf{X_i} - \bar{\mathbf{X}})^\top \right) \mathbf{z}$$

$$= \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{z}^\top (\mathbf{X_i} - \bar{\mathbf{X}})(\mathbf{X_i} - \bar{\mathbf{X}})^\top \mathbf{z}$$

$$= \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{z}^\top(\mathbf{X_i}-\bar{\mathbf{X}}))^2.$$

Since a square of a real number is always non-negative, $(\mathbf{z}^\top(\mathbf{X_i}-\bar{\mathbf{X}}))^2 \geqslant 0$. Thus, $\mathbf{z}^\top\hat{\Sigma}_n\mathbf{z} \geqslant 0$, proving $\hat{\Sigma}_n$ is positive semidefinite.

(iii) For the sake of clarity, consider the case of $p = 2$, where the mean is known to be the zero vector. In this case, we can assume the sample covariance matrix is given by

$$\hat{\Sigma}_n = \frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix}X_i\\Y_i\end{bmatrix}\begin{bmatrix}X_i & Y_i\end{bmatrix} = \frac{1}{n}\begin{bmatrix}\sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_iY_i\\ \sum_{i=1}^n X_iY_i & \sum_{i=1}^n Y_i^2\end{bmatrix} = \frac{1}{n}\mathbf{W}^\top\mathbf{W},$$

where $\mathbf{W} = [\mathbf{X}, \mathbf{Y}]$.
Now consider the probability that any two random vectors, say $\mathbf{X}$ and $\mathbf{Y}$, are colinear. This probability is given by

$$\begin{aligned}\mathbb{P}(\mathbf{X},\mathbf{Y}\text{ colinear}) &= \int_{\mathbb{R}^n}\mathbb{P}(\mathbf{X},\mathbf{Y}\text{ colinear}\mid\mathbf{Y}=\mathbf{y})\,F_{\mathbf{Y}}(d\mathbf{y})\\ &= \int_{\mathbb{R}^n}\mathbb{P}(\mathbf{X},\mathbf{y}\text{ colinear})\,F_{\mathbf{Y}}(d\mathbf{y})\\ &= \int_{\mathbb{R}^n}\mathbb{P}\left(\mathbf{X},\frac{\mathbf{y}}{\|\mathbf{y}\|}\text{ colinear}\right)F_{\mathbf{Y}}(d\mathbf{y}).\end{aligned}$$

In the simplification above, the second equality follows from the substitution principle, and the third one, from the independence of $\mathbf{X}$ and $\mathbf{Y}$. As shown in A.2, there exists an orthogonal matrix $O \in O(n)$ such that $O\left(\frac{\mathbf{y}}{\|\mathbf{y}\|}\right) = \mathbf{e}_1 = (1, 0, \ldots, 0)^T$, so the expression above can be re-expressed as

$$\begin{aligned}\int_{\mathbb{R}^n}\mathbb{P}\left(O\mathbf{X}, O\left(\frac{\mathbf{y}}{\|\mathbf{y}\|}\right)\text{ colinear}\right)F_{\mathbf{Y}}(d\mathbf{y}) &= \int_{\mathbb{R}^n}\mathbb{P}(\mathbf{X},\mathbf{e}_1\text{ colinear})\,F_{\mathbf{Y}}(d\mathbf{y})\\ &= \int_{\mathbb{R}^n}\mathbb{P}(\exists\lambda\in\mathbb{R}\text{ such that }\lambda\mathbf{X}=\mathbf{e}_1)\,F_{\mathbf{Y}}(d\mathbf{y})\\ &= \int_{\mathbb{R}^n}\mathbb{P}(\exists\lambda\text{ such that }\lambda X_1=1)\cdot\prod_{i=2}^{n}\mathbb{P}(X_i=0)\,F_{\mathbf{Y}}(d\mathbf{y})\\ &= 0.\end{aligned}$$

Thus, $\mathbb{P}(\mathbf{X}, \mathbf{Y}\text{ colinear}) = 0$, implying that $\mathbf{W}$ has full rank and, consequently, that $\hat{\Sigma}_n$ is also full rank, with no zero eigenvalues. Hence, all eigenvalues are positive, and $\hat{\Sigma}_n$ is positive definite. This result extends to general $p \in \mathbb{R}$ (Abry et al., 2025, p.35). □

We can now establish the almost sure convergence of sample covariance matrices in low dimensions.

Proposition 3.3. *Let $\hat{\Sigma}_n$ be as defined in* (3.2) *for $n \in \mathbb{N}$. Then, as $n \to \infty$, $\hat{\Sigma}_n \xrightarrow{a.s.} \Sigma$.*

Proof. First, consider the case $p = 1$. Let $Y_i = X_i^2 \geqslant 0$. By Kolmogorov's Strong Law of Large Numbers, as $n \to \infty$,

$$\frac{1}{n}\sum_{i=1}^{n}Y_i \xrightarrow{a.s.} \mathbb{E}[Y_1] = \mathbb{E}[X_1^2]. \tag{3.4}$$

Also, for two i.i.d. sequences $\{Z_i\}, \{W_i\}$ with finite second moments and independent across i, the Strong Law yields

$$\frac{1}{n}\sum_{i=1}^{n}Z_iW_i \xrightarrow{a.s.} \mathbb{E}[Z_1W_1]. \tag{3.5}$$

Now let $p \in \mathbb{N}$ and $\mathbf{X}_i \in \mathbb{R}^p$. By applying (3.4) and (3.5) entry-wise, we obtain

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i \mathbf{X}_i^\top \xrightarrow{a.s.} \mathbb{E}[\mathbf{X}_1 \mathbf{X}_1^\top] = \Sigma.$$

Now consider the general case, without the mean zero assumption. The sample covariance matrix is

$$\hat{\Sigma}_n = \frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \bar{\mathbf{X}})(\mathbf{X}_i - \bar{\mathbf{X}})^\top,$$

where $\bar{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{X}_i$. Rewriting, we have

$$\hat{\Sigma}_n = \frac{1}{n-1}\sum_{i=1}^{n} \left[(\mathbf{X}_i - \boldsymbol{\mu}) + (\boldsymbol{\mu} - \bar{\mathbf{X}})\right]\left[(\mathbf{X}_i - \boldsymbol{\mu}) + (\boldsymbol{\mu} - \bar{\mathbf{X}})\right]^\top.$$

Expanding, we obtain four terms:

$$\hat{\Sigma}_n = \frac{1}{n-1}\sum_{i=1}^{n} \left[(\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^\top + (\mathbf{X}_i - \boldsymbol{\mu})(\boldsymbol{\mu} - \bar{\mathbf{X}})^\top + (\boldsymbol{\mu} - \bar{\mathbf{X}})(\mathbf{X}_i - \boldsymbol{\mu})^\top + (\boldsymbol{\mu} - \bar{\mathbf{X}})(\boldsymbol{\mu} - \bar{\mathbf{X}})^\top\right].$$

The first term is

$$\frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^\top.$$

The second term is

$$\frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \boldsymbol{\mu})(\boldsymbol{\mu} - \bar{\mathbf{X}})^\top = \frac{1}{n-1}(\boldsymbol{\mu} - \bar{\mathbf{X}})\sum_{i=1}^{n} (\mathbf{X}_i - \boldsymbol{\mu}).$$

Since $\sum_{i=1}^{n}(\mathbf{X}_i - \boldsymbol{\mu}) = n(\bar{\mathbf{X}} - \boldsymbol{\mu})$, this term vanishes as $n \to \infty$ because $\bar{\mathbf{X}} \xrightarrow{a.s.} \boldsymbol{\mu}$.
The third term is

$$\frac{1}{n-1}\sum_{i=1}^{n} (\boldsymbol{\mu} - \bar{\mathbf{X}})(\mathbf{X}_i - \boldsymbol{\mu})^\top,$$

which also vanishes as $n \to \infty$ by the same reasoning.
The fourth term is

$$\frac{n}{n-1}(\boldsymbol{\mu} - \bar{\mathbf{X}})(\boldsymbol{\mu} - \bar{\mathbf{X}})^\top,$$

which satisfies

$$\frac{n}{n-1}(\boldsymbol{\mu} - \bar{\mathbf{X}})(\boldsymbol{\mu} - \bar{\mathbf{X}})^\top \xrightarrow{a.s.} 0$$

since $\bar{\mathbf{X}} \xrightarrow{a.s.} \boldsymbol{\mu}$. Thus, as $n \to \infty$, only the first term remains, and

$$\hat{\Sigma}_n \xrightarrow{a.s.} \frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^\top.$$

By the Strong Law of Large Numbers,

$$\frac{1}{n-1}\sum_{i=1}^{n} (\mathbf{X}_i - \boldsymbol{\mu})(\mathbf{X}_i - \boldsymbol{\mu})^\top \xrightarrow{a.s.} \mathbb{E}\left[(\mathbf{X}_1 - \boldsymbol{\mu})(\mathbf{X}_1 - \boldsymbol{\mu})^\top\right] = \Sigma.$$

Finally, since $\frac{n}{n-1} \to 1$, we conclude that

$$\hat{\Sigma}_n \xrightarrow{a.s.} \Sigma.$$

□

Definition 3.3. If $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ are i.i.d. p-dimensional random vectors with a multivariate normal distribution $N_p(\mathbf{0}, \mathbf{\Sigma})$, then the matrix $\mathbf{S} = \sum_{i=1}^{n} \mathbf{X}_i \mathbf{X}_i^\top$ follows a *Wishart distribution* with n degrees of freedom and covariance matrix $\mathbf{\Sigma}$, denoted as $\mathbf{S} \sim \mathcal{W}_p(n, \mathbf{\Sigma})$. The p.d.f. of $\mathbf{S}$ is given by

$$f_{\mathbf{S}}(\mathbf{S}) = \frac{|\mathbf{S}|^{(n-p-1)/2} \exp\left(-\frac{1}{2}\mathrm{tr}(\mathbf{\Sigma}^{-1}\mathbf{S})\right)}{2^{np/2}|\mathbf{\Sigma}|^{n/2}\Gamma_p\left(\frac{n}{2}\right)},$$

where Γ_p is the multivariate gamma function.

Proposition 3.4. *Assuming $p \leqslant n$, the ordered eigenvalues $\lambda_1 \leqslant \ldots \leqslant \lambda_p$ of a standard Wishart matrix for the case $\beta = 1$ (real-valued Gaussian) follow a distribution whose joint p.d.f. is given by*

$$f(\lambda_1, \ldots, \lambda_p) = C \cdot \mathbf{1}_{\{0 \leqslant \lambda_1 \leqslant \ldots \leqslant \lambda_p\}} \cdot \prod_{\ell=1}^{p} \lambda_\ell^{(n-p-1)/2} \cdot \exp\left\{-\frac{1}{2}\sum_{\ell=1}^{p} \lambda_\ell\right\} \cdot \prod_{i<\ell} (\lambda_\ell - \lambda_i),$$

where C is a normalization constant, and $\mathbf{1}_{\{0 \leqslant \lambda_1 \leqslant \ldots \leqslant \lambda_p\}}$ is an indicator function that ensures the eigenvalues are ordered and non-negative.

The term $\prod_{i<\ell}(\lambda_\ell - \lambda_i)$, known as the *Vandermonde determinant*, reduces the probability of any two eigenvalues being close together, causing the characteristic eigenvalue repulsion seen in the Wishart distribution.

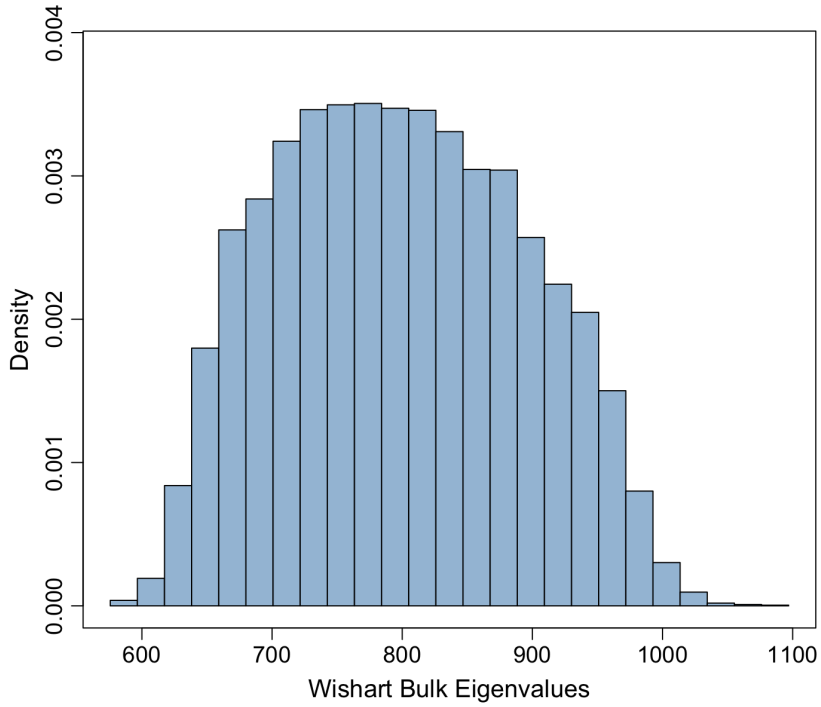


Figure 3: **Wishart Eigenvalue Repulsion**, generated using the code in listing 3, with $p = 10$ and $n = 250$ observations. The spread of bulk eigenvalues reflects eigenvalue repulsion in the Wishart distribution, induced by the Vandermonde determinant.

It is interesting to note the connection between sample covariance matrices and the renowned Gaussian ensembles. This relationship is more accurately stated in the following proposition.

Proposition 3.5. *Let S be as in Definition 3.3, where $\Sigma = I$. Then,*

$$\sqrt{n}\left(\frac{S}{n} - I\right) \xrightarrow{d} GOE, \quad n \to \infty,$$

where GOE denotes the Gaussian Orthogonal Ensemble: a symmetric matrix with independent Gaussian entries of variance 2 and 1 on the diagonal and off-diagonal entries, respectively. This result links the Wishart matrix to the GOE (Bai and Silverstein, 2010, p.15).

3.1 Low Dimensions

In low dimensions, p is fixed while $n \to \infty$. For the simplest example of $\hat{\Sigma}_n$, consider the bivariate case ($p = 2$), where

$$\hat{\Sigma}_n = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix}.$$

Naturally, we are interested in the eigenvalue asymptotics of $\hat{\Sigma}_n$ in this regime.

Proposition 3.6. *For independent $X_1 \sim \mathcal{N}(0,1)$, $X_2 \sim \mathcal{N}(0,2)$, the largest eigenvalue of $\hat{\Sigma}_n$ is asymptotically Gaussian. That is, if we consider the asymptotic fluctuations of the top eigenvalue as $\widetilde{\lambda}_2 = \sqrt{n}\left(\lambda_2(\hat{\Sigma}_n) - 2\right)$ then*

$$\widetilde{\lambda}_2 \xrightarrow{d} \mathcal{N}(0,8), \quad n \to \infty$$

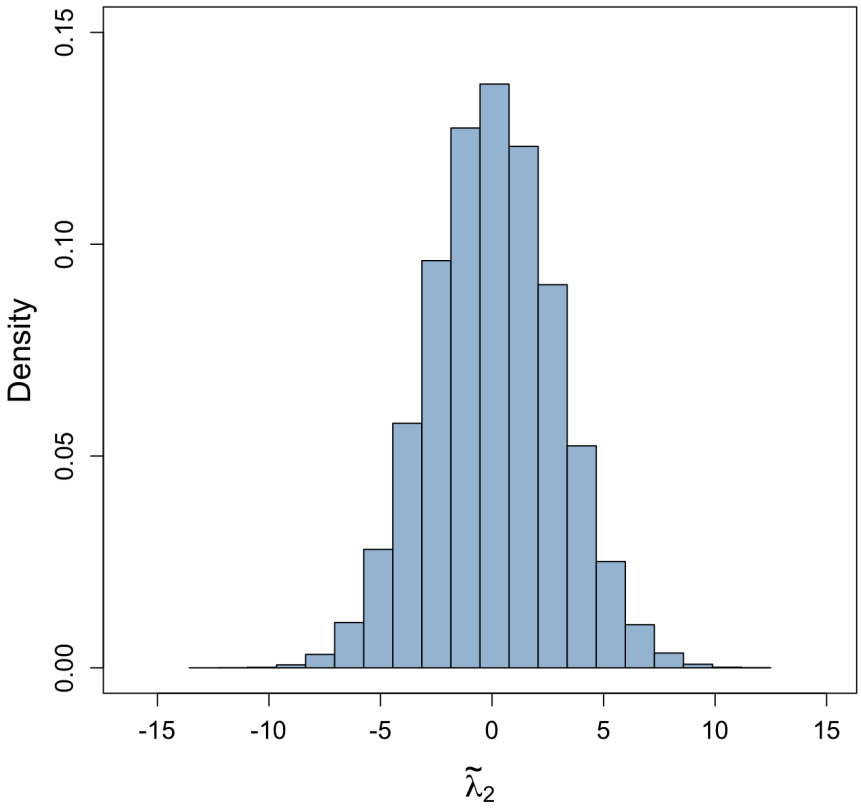


Figure 4: **Asymptotic Gaussianity in Low Dimensions**, generated using the code in listing 4, with $n = 5000$, $p = 2$, and $R = 2000$ iterations. As n increases, the empirical distribution of the top eigenvalue of $\hat{\Sigma}_n$ closely matches $\mathcal{N}(0,8)$, confirming asymptotic Gaussianity in the low-dimensional regime.

Proposition 3.7. *For independent $X_1, X_2 \sim \mathcal{N}(0,1)$, the eigenvalues of $\hat{\Sigma}_n$ are asymptotically GOE-like. That is,*

$$\sqrt{n}(\hat{\Sigma}_n - I_2) \xrightarrow{d} \mathrm{GOE}_2(\eta), \quad n \to \infty$$

where $\mathrm{GOE}_2(\eta)$ denotes a 2×2 Gaussian Orthogonal Ensemble matrix scaled by a variance factor $\eta > 0$.

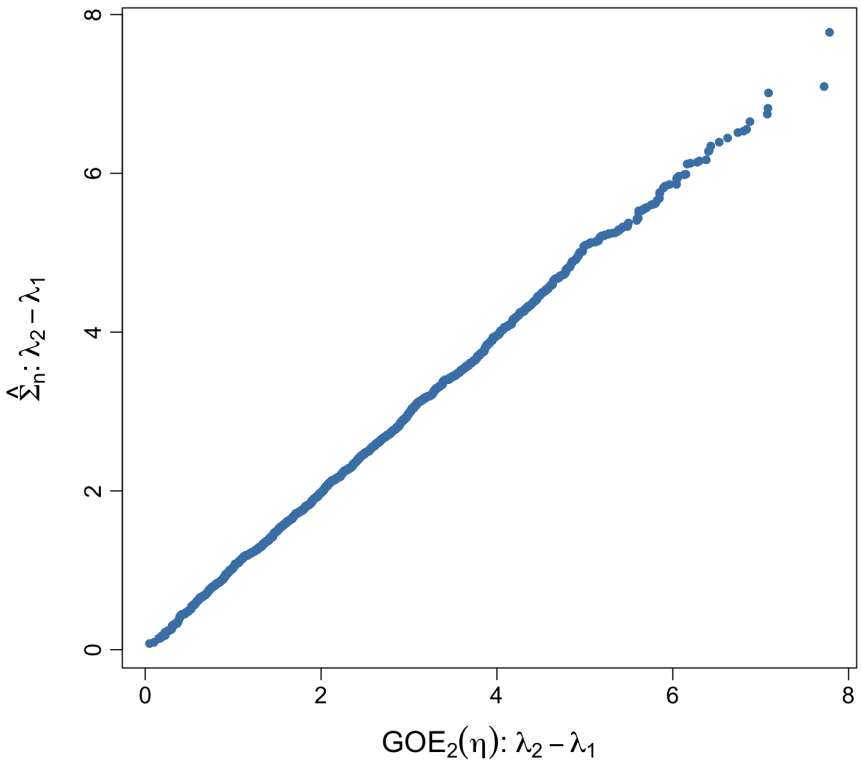


Figure 5: **GOE-like Repulsion in Low Dimensions**, generated using the code in listing 5, with $n = 5000$, $p = 2$, and $R = 2000$ iterations. The straight-line fit of the QQ plot comparing $\lambda_2 - \lambda_1$ from $\hat{\Sigma}_n$ to that of $\mathrm{GOE}_2(\eta)$ confirms GOE-like eigenvalue repulsion in $\hat{\Sigma}_n$.

3.2 High Dimensions

In high dimensions, both n and p grow to infinity while maintaining a fixed ratio. That is,

$$n, p \to \infty, \quad \frac{p}{n} \to c \in (0, \infty)$$

Applications and motivations include bioinformatics, where n denotes the sample size and p the number of genes, and neurology, where n represents time and p the number of sensors.

Proposition 3.8. *In the high-dimensional regime, the bulk eigenvalue distribution of $\hat{\Sigma}_n$ converges in distribution to the Marchenko–Pastur law, whose density is given by*

$$\rho_c(\lambda) = \begin{cases} (c-1)\delta(\lambda) + \frac{1}{2\pi c\lambda}\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)} & \text{if } c > 1, \\ \frac{1}{2\pi c\lambda}\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)} & \text{if } 0 < c \leqslant 1, \end{cases}$$

for $\lambda \in [\lambda_-, \lambda_+]$, where $\lambda_\pm = (1 \pm \sqrt{c})^2$, and $\delta(\lambda)$ is the Dirac delta function centered at zero.

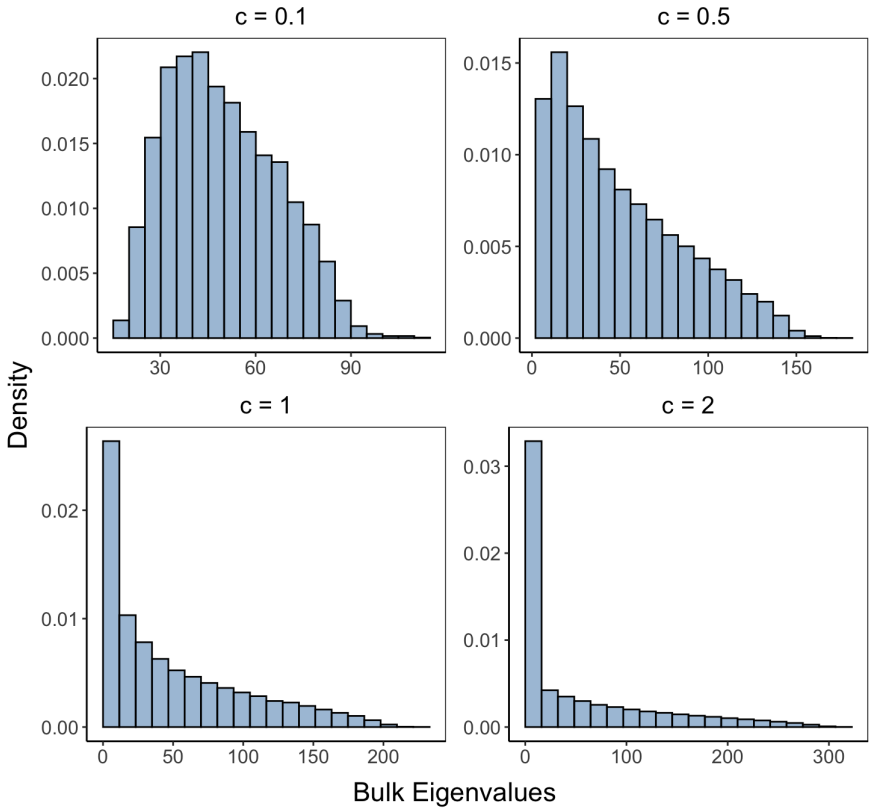


Figure 6: **Marchenko-Pastur Distribution**, generated using the code in listing 6, with $n = 1000$ and varying values of $c = p/n$. As c increases, the empirical bulk eigenvalue distribution transitions from being more uniformly spread ($c < 1$), to peaking at zero with a heavier, extended tail ($c \geqslant 1$), consistent with the Marchenko–Pastur law.

Unlike in low dimensions, where the top eigenvalue of $\hat{\Sigma}_n$ is asymptotically normal, in high dimensions it is governed by the Tracy–Widom law.

Proposition 3.9. *In the high-dimensional regime, the top eigenvalue of* $\hat{\Sigma}_n$ *converges in distribution to the Tracy-Widom Law, given by*

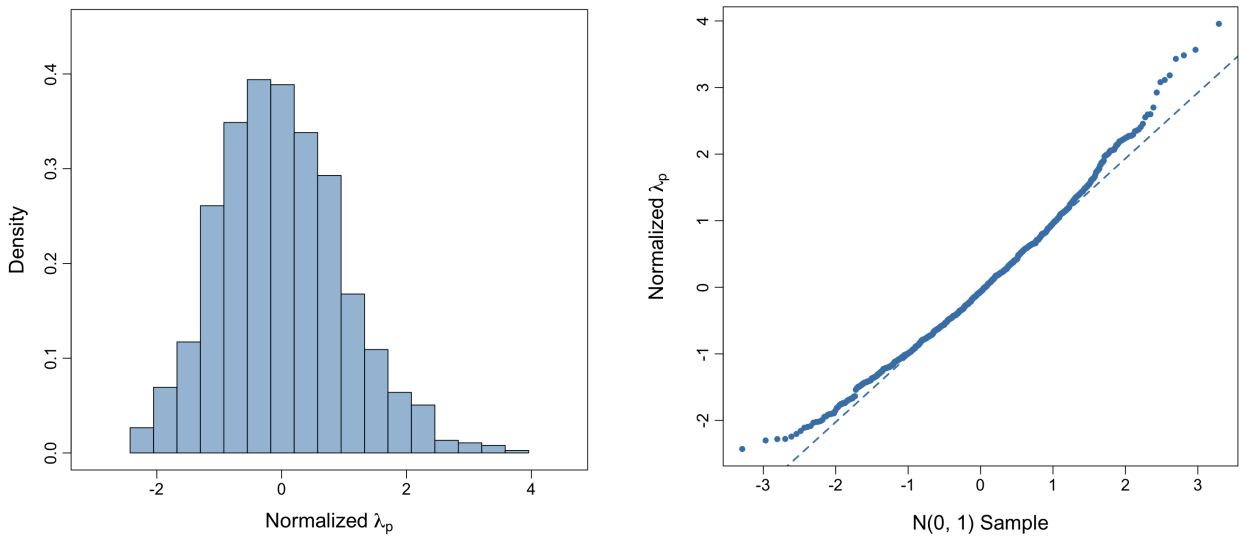
$$n^{\frac{2}{3}}\left(\frac{\lambda_p(\hat{\Sigma}_n) - \mu_{n,p}}{\sigma_{n,p}}\right) \xrightarrow{d} F_1, \quad \mu_{n,p} \to (1+\sqrt{c})^2, \quad \sigma_{n,p} \to (1+\sqrt{c})\left(1+\sqrt{\frac{1}{c}}\right)^{\frac{1}{3}},$$

where $F_1(s)$ *is given by*

$$F_1(s) = \exp\left(-\frac{1}{2}\int_s^\infty (q(x) + (x-s)q^2(x))dx\right),$$

and $q(x)$ *satisfies the Painlevé II equation*

$$q''(x) = xq(x) + 2q^3(x), \quad q(x) \sim Ai(x) \text{ as } x \to \infty.$$



(a) (b)

Figure 7: **Tracy-Widom Distribution**, generated using the code in listing 7, with $n = 3000$, $p = 1500$, $c = 0.5$, and $R = 2000$ iterations. The empirical distribution of $\lambda_p(\hat{\Sigma}_n)$ appears right-skewed, and the curved QQ plot comparing normalized $\lambda_p(\hat{\Sigma}_n)$ to a standard normal sample confirms that the top eigenvalue is not asymptotically Gaussian, in contrast to the low-dimensional case.

4 Fractional Brownian Motion

Fractional Brownian motion (fBm) is a universal model for long-range dependence, with applications in fields such as biophysics and network traffic. It is constructed by cumulatively summing fractional Gaussian noise (fGn), resulting in a process whose self-similar increments and dependence structure make it particularly useful for analyzing sample covariance matrices in settings where variation across scales is of interest.

Definition 4.1. A process $\{B_H(t)\}_{t\geqslant 0}$ is called a *fractional Brownian motion* (fBm) with Hurst parameter $H \in (0, 1)$ if:

(*i*) It is a Gaussian process.

(*ii*) It is self-similar: for any $c > 0$, $\{B_H(ct)\}_{t\geqslant 0} \overset{d}{=} \{c^H B_H(t)\}_{t\geqslant 0}$.

(*iii*) It has stationary increments: for all $t, s \geqslant 0$, the distribution of $B_H(t+s) - B_H(s)$ depends only on t.

The Hurst parameter controls the roughness of the process:

- $H = 0.5$ corresponds to standard Brownian motion $B(t)$, characterized by independent increments.
- $H < 0.5$ reflects anti-persistence, where increases are likely to be followed by decreases, and vice versa.
- $H > 0.5$ reflects long-range dependence, where increases are likely to be followed by further increases, and vice versa.

The covariance function of $B_H(t)$ is given by

$$\mathbb{E}[B_H(t)B_H(s)] = \frac{1}{2}\left(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\right), \quad t, s \in \mathbb{R}.$$

For standard Brownian motion, (4) simplifies down to

$$\mathbb{E}[B_{H=0.5}(t)B_{H=0.5}(s)] = \min(t, s), \quad t, s \in \mathbb{R}.$$

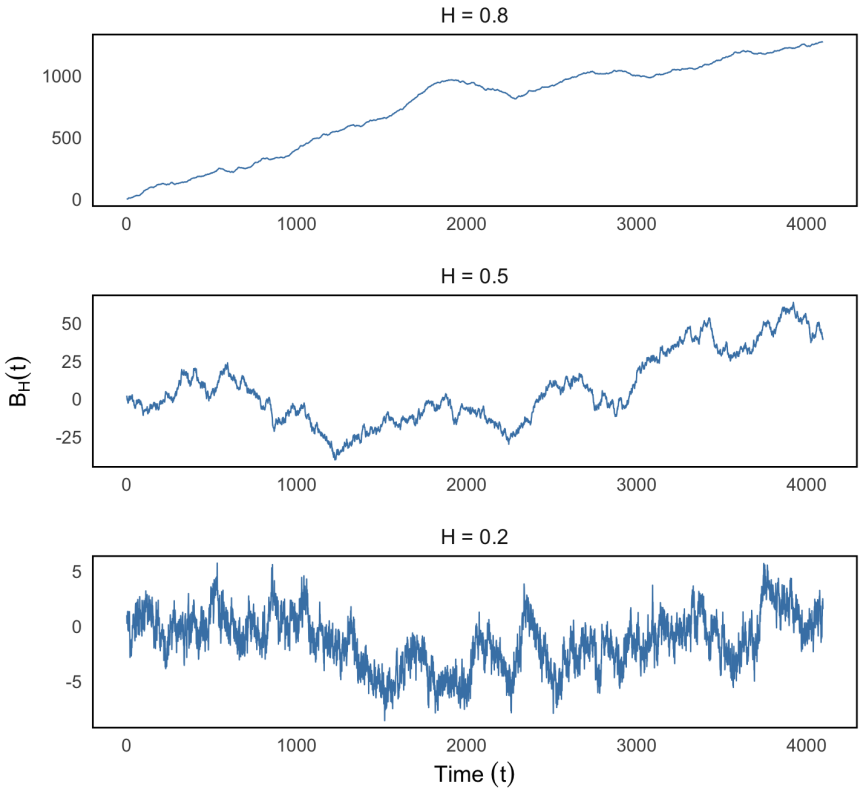


Figure 8: **fBm Paths**, generated using the code in listing 8, with 2^{12} steps and Hurst parameters $H = 0.2$, 0.5, and 0.8. For $H = 0.5$, the process reduces to standard Brownian motion with independent increments, while for $H > 0.5$ and $H < 0.5$, it exhibits long-range dependence and anti-persistence, respectively.

5 Multiscale Sample Covariance Matrices

To introduce serial dependence into the data, we model the underlying process using bivariate fractional Brownian motion with independent components. A linear transformation by a rotation matrix P then introduces dependence between components, resulting in the process $Y(t)$ defined below.

Definition 5.1. Let $\{B_{H_1}(t)\}_{t\geqslant 0}$ and $\{B_{H_2}(t)\}_{t\geqslant 0}$ be two independent fractional Brownian motion processes with Hurst parameters H_1 and H_2, respectively. Define the two-dimensional stochastic process $Y(t)$ as

$$Y(t) = P\begin{bmatrix} B_{H_1}(t) \\ B_{H_2}(t) \end{bmatrix} =: PX(t), \quad t \in \mathbb{R}.$$

In (5.1), P is a 2×2 rotation matrix that introduces dependence between the components of $Y(t)$. In particular, we can write

$$P = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \in O(2), \quad \theta \in [0, 2\pi).$$

With the dependent process $Y(t)$ defined, we now construct the associated multiscale sample covariance matrix, allowing us to capture variance structures at any lag h.

Definition 5.2. The *multiscale sample covariance matrix* $M(h)$ is defined as

$$M(h) = \frac{1}{n-h} \sum_{t=1}^{n-h} \left(Y(t+h) - Y(t)\right)\left(Y(t+h) - Y(t)\right)^\top, \quad n > h,$$

where $h \in \mathbb{N} \cup \{0\}$ is called the *lag parameter*.

5.1 Scaling Laws

In the two-way limit $n, h \to \infty$, it is expected that the growth of both the entries and of the eigenvalues of $M(h)$ or $\mathbb{E}M(h)$ follow a power law in h. In fact, the following two propositions confirm this expectation. In the first one, we establish the entry-wise scaling behavior of $\mathbb{E}M(h)$.

Proposition 5.1. *Consider measurements of the form* (5.1)*, where* $0 < H_1 < H_2 < 1$ *and* $\theta \notin \{0, 3\pi/2\}$*. Then, each entry of* $M(h)$*, denoted by* $M(h)_{1,1}$*,* $M(h)_{2,2}$ *and* $M(h)_{1,2}$*, grow at a rate dominated by* $2H_2$*. That is, as* $n, h \to \infty$ *with* $h/n = O(1)$*,*

$$\mathbb{E}M(h)_{1,1} \sim c_1|h|^{2H_2}, \quad \mathbb{E}M(h)_{2,2} \sim c_2|h|^{2H_2}, \qquad \mathbb{E}M(h)_{1,2} \sim c_{12}|h|^{2H_2},$$

for constants $c_1, c_2, c_{12} > 0$*.*

Proof. Without loss of generality, consider only the diagonal elements of $M(h)$. Relation (5.1) implies that we can write

$$Y_1(t) = B_{H_1}(t)\cos(\theta) - B_{H_2}(t)\sin(\theta), \quad Y_2(t) = B_{H_1}(t)\sin(\theta) + B_{H_2}(t)\cos(\theta).$$

For the first diagonal entry, consider

$$\mathbb{E}\big[(Y_1(t+h) - Y_1(t))^2\big] = \cos^2(\theta)\mathbb{E}\big[(B_{H_1}(t+h) - B_{H_1}(t))^2\big] + \sin^2(\theta)\mathbb{E}\big[(B_{H_2}(t+h) - B_{H_2}(t))^2\big].$$

Since fBm satisfies

$$\mathbb{E}\big[(B_H(t+h) - B_H(t))^2\big] = \sigma^2|h|^{2H},$$

we substitute this into our expression, obtaining

$$\mathbb{E}\big[(Y_1(t+h) - Y_1(t))^2\big] = \cos^2(\theta)\sigma^2|h|^{2H_1} + \sin^2(\theta)\sigma^2|h|^{2H_2},$$

where $\sin^2(\theta) \neq 0$. We can then factor out $|h|^{2H_2}$, resulting in $|h|^{2H_2}\big(\cos^2(\theta)\sigma^2|h|^{2(H_1-H_2)} + \sin^2(\theta)\sigma^2\big)$. Now, as $h \to \infty$, observe that $|h|^{2(H_1-H_2)} \to 0$. Thus, the term $\cos^2(\theta)\sigma^2|h|^{2(H_1-H_2)}$ vanishes, leaving $M(h)_{1,1} \sim \sin^2(\theta)\sigma^2|h|^{2H_2}$, where $\sin^2(\theta)\sigma^2$ is a constant. A similar argument applies to $M(h)_{2,2}$, confirming that the growth rate for both diagonal elements is $|h|^{2H_2}$. □

In the following proposition, we show that, unlike the diagonal elements, the eigenvalues of $\mathbb{E}M(h)$ exhibit distinct Hurst-based scaling behaviors.

Proposition 5.2. *Suppose the assumptions of Proposition 5.1 hold. Then, the smaller eigenvalue* $\lambda_1(\mathbb{E}M(h))$ *grows at a rate determined by* $2H_1$*, while the larger eigenvalue* $\lambda_2(\mathbb{E}M(h))$ *grows at a rate determined by* $2H_2$*. That is, as* $n, h \to \infty$ *with* $h/n = O(1)$*,*

$$\lambda_1(\mathbb{E}M(h)) \sim c_1|h|^{2H_1}, \quad \lambda_2(\mathbb{E}M(h)) \sim c_2|h|^{2H_2}$$

for constants $c_1, c_2 > 0$*, provided that* $\theta \neq 0$*.*

Proof. We begin with $M(h)$ as defined in Definition 5.2. Since the process $Y(t)$ is obtained by applying the orthogonal transformation P to the entry-wise independent fBm process $X(t)$, we can write

$$Y(j+h) = PX(j+h), \quad Y(j) = PX(j).$$

Substituting these expressions into $\mathbb{E}M(h)$, we obtain

$$\mathbb{E}M(h) = \mathbb{E}\Big(\frac{1}{n-h}\sum_{j=1}^{n-h}(PX(j+h) - PX(j))(PX(j+h) - PX(j))^\top\Big).$$

Factoring out P, we rewrite (5.1) as

$$\mathbb{E}M(h) = P\mathbb{E}\left(\frac{1}{n-h}\sum_{j=1}^{n-h}(X(j+h)-X(j))(X(j+h)-X(j))^\top\right)P^\top.$$

We obtain

$$\lambda_\ell(\mathbb{E}M(h)) = \lambda_\ell\left(\mathbb{E}\left(\frac{1}{n-h}\sum_{j=1}^{n-h}(X(j+h)-X(j))(X(j+h)-X(j))^\top\right)\right), \quad \ell = 1,2.$$

The claim is now a consequence of the large-scale behavior of the univariate mean-squared displacement (cf. Table 1 in Zhang et al. (2018)). □

In summary, Propositions 5.1 and 5.2 show that, while the diagonal entries of $M(h)$ alone cannot recover both H_1 and H_2, the eigendomain is able to do so. To confirm these results, the scaling exponent α of the diagonal elements and eigenvalues can be estimated via log-log regression; since $M(h)_{i,i} \sim |h|^{\alpha_{ii}}$ and $\lambda_i(M(h)) \sim |h|^{\alpha}_{\lambda_i}$, taking logarithms gives

$$\log M(h)_{i,i} \approx \alpha \log|h| + C, \quad \log\lambda_i(M(h)) \approx \alpha\log|h| + C. \tag{5.1}$$

Thus, α corresponds to the slope of a regression between $\log M(h)_{i,i}$ or $\log\lambda_i(M(h))$ and $\log h$. As demonstrated in Propositions 5.1 and 5.2, the estimated α values satisfy

$$\alpha_{M_{1,1}}, \alpha_{M_{2,2}} \approx 2H_2, \quad \alpha_{\lambda_1} \approx 2H_1, \quad \alpha_{\lambda_2} \approx 2H_2. \tag{5.2}$$

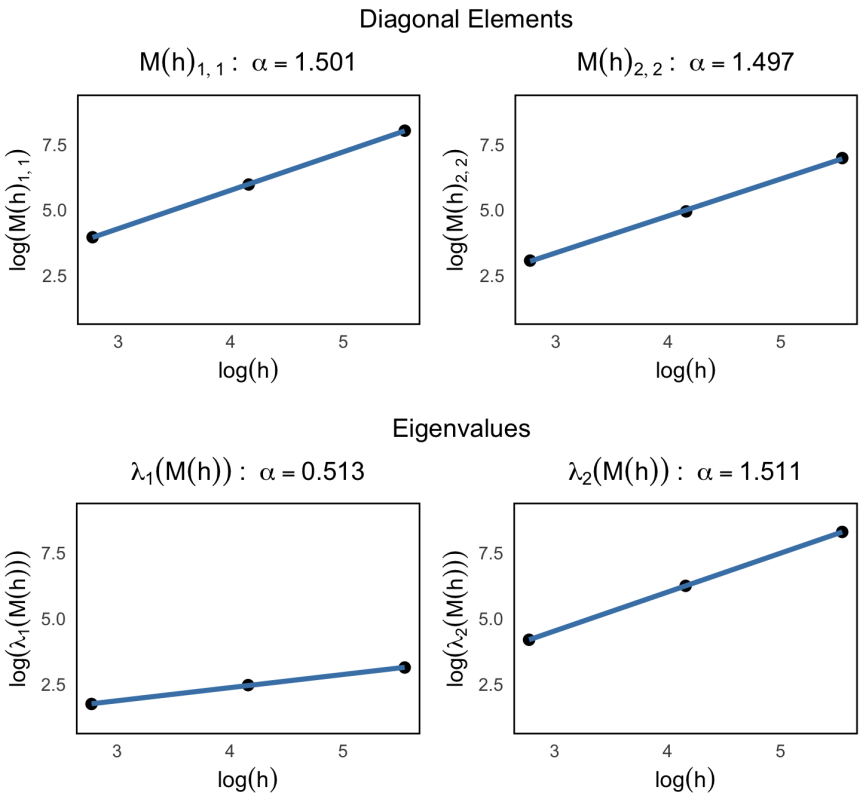


Figure 9: $M(h)$ **Scaling Exponent Estimation**, generated using the code in listing 9, with $n = 2^{16}$ steps, $h \in \{2^4, 2^6, 2^8\}$, $H_1 = 0.25$, and $H_2 = 0.75$. The slopes from the log-log regression confirm that the diagonal entries scale as $2H_2 \approx 1.5$, while the eigenvalues split as $2H_1 \approx 0.5$ and $2H_2 \approx 1.5$, demonstrating dominance by H_2 in the diagonals and distinct Hurst-based scaling in the eigenvalues.

Remark 5.1. Going beyond Propositions 5.1 and 5.2, one can now ask about the scaling behavior of the eigenvalues of $M(h)$. It is not difficult to show that the eigenvalues of $M(h)$ behave in a way that is analogous to (5.2), where asymptotic equivalence is replaced with asymptotic equivalence in probability. This topic will be explored in depth in high dimensions (see Section 5.2).

In turn, regarding the related issue of the fluctuations of eigenvalues of $M(h)$, for $0 < H_1 \leqslant H_2 < 3/4$ they are either Gaussian or non-Gaussian depending on whether or not $H_1 = H_2$. The critical value $H = 0.75$ marks a phase transition in the fluctuations of the eigenvalue asymptotics of $M(h)$. This threshold emerges from a squared summability constraint on the autocovariance function of fGn (in dimension 1). This can be briefly explained as follows. Since $\gamma_H(k) \sim Ck^{2H-2}$ as $k \to \infty$, it follows that

$$|\gamma_H(k)|^2 \sim \frac{C^2}{k^{2(2-2H)}}.$$

By applying the summability constraint, we have

$$\sum_{k=1}^{\infty} |\gamma_H(k)|^2 < \infty \quad \Longleftrightarrow \quad 2(2-2H) > 1 \quad \Longleftrightarrow \quad H < \frac{3}{4}. \tag{5.3}$$

The topic of the fluctuations of the eigenvalues of $M(h)$ will also be addressed in Section 5.2.

5.2 Multiscale Analysis in High Dimensions

Consider the high-dimensional model given by

$$Y(t) = PX(t) + Z(t)$$

where

- $P \in \mathbb{R}^{p\times 2}$ contains a 2×2 orthogonal matrix $O_2 \in \mathrm{O}(2)$ in its top two rows and zeros elsewhere,
- $X(t) \in \mathbb{R}^2$ is a two-dimensional fBm with independent components $B_{H_1}(t)$ and $B_{H_2}(t)$,
- $Z(t) \in \mathbb{R}^p$ is Gaussian noise.

In the high-dimensional, large-scale regime, we consider $n, p, h \to \infty$ with $\frac{ph}{n} = O(1)$, and to analyze the eigenvalue asymptotics of p-dimensional $M(h)$, we consider the asymptotic fluctuations of the top two eigenvalues as

$$\widetilde{\lambda}_{p-1} = \sqrt{\frac{n}{h}} \left(\log \lambda_{p-1}(M(h)) - \log \lambda_{p-1}(\mathbb{E}M(h))\right), \quad \widetilde{\lambda}_{p} = \sqrt{\frac{n}{h}} \left(\log \lambda_{p}(M(h)) - \log \lambda_{p}(\mathbb{E}M(h))\right). \tag{5.4}$$

It will also be useful to consider the decompositions

$$M_x(h) = \frac{1}{n-h} \sum_{j=1}^{n-h} \left(X(j+h) - X(j)\right)\left(X(j+h) - X(j)\right)^\top, \tag{5.5}$$

$$M_z(h) = \frac{1}{n-h} \sum_{j=1}^{n-h} \left(Z(j+h) - Z(j)\right)\left(Z(j+h) - Z(j)\right)^\top = O_{\mathbb{P}}(1), \tag{5.6}$$

and

$$M_{x,z}(h) = \frac{1}{n-h} \sum_{j=1}^{n-h} \left(X(j+h) - X(j)\right)\left(Z(j+h) - Z(j)\right)^\top. \tag{5.7}$$

In view of (5.5)–(5.7), we can conveniently rewrite

$$M(h) = PM_x(h)P^\top + M_z(h) + PM_{x,z}(h) + M_{x,z}(h)^\top P^\top. \tag{5.8}$$

So, let

$$\mathbf{H} = \begin{bmatrix} H_1 & 0 \\ 0 & H_2 \end{bmatrix} \quad \text{so that} \quad h^{-\mathbf{H}} = \begin{bmatrix} h^{-H_1} & 0 \\ 0 & h^{-H_2} \end{bmatrix}.$$

As it turns out, the asymptotic behavior of (5.4) hinges on the fluctuations of the auxiliary (low-dimensional) random matrix

$$\widehat{B}(h) = h^{-\mathbf{H}} M_x(h) h^{-\mathbf{H}^\top}, \tag{5.9}$$

where $M_x(h)$ is given as in (5.5).

We break up the remainder of this section into subsections, according to the range of the underlying Hurst exponents (see Remark 5.1).

5.2.1 Pre-Phase Transition

The following theorem is the first of the two main mathematical results of this thesis.

Theorem 5.1. *Suppose $0 < H_1 < H_2 < 0.75$. In the high-dimensional, large-scale regime where $n, p, h \to \infty$ with $\frac{ph}{n} = O(1)$, the top two eigenvalues of $M(h)$ are asymptotically independent and jointly Gaussian. That is, for $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$, as defined in* (5.4)

$$\begin{pmatrix} \widetilde{\lambda}_{p-1} \\ \widetilde{\lambda}_p \end{pmatrix} \xrightarrow{d} \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \Sigma_\lambda\right),$$

where $\Sigma_\lambda \in \mathcal{S}_{>0}(2, \mathbb{R})$ is a diagonal matrix.

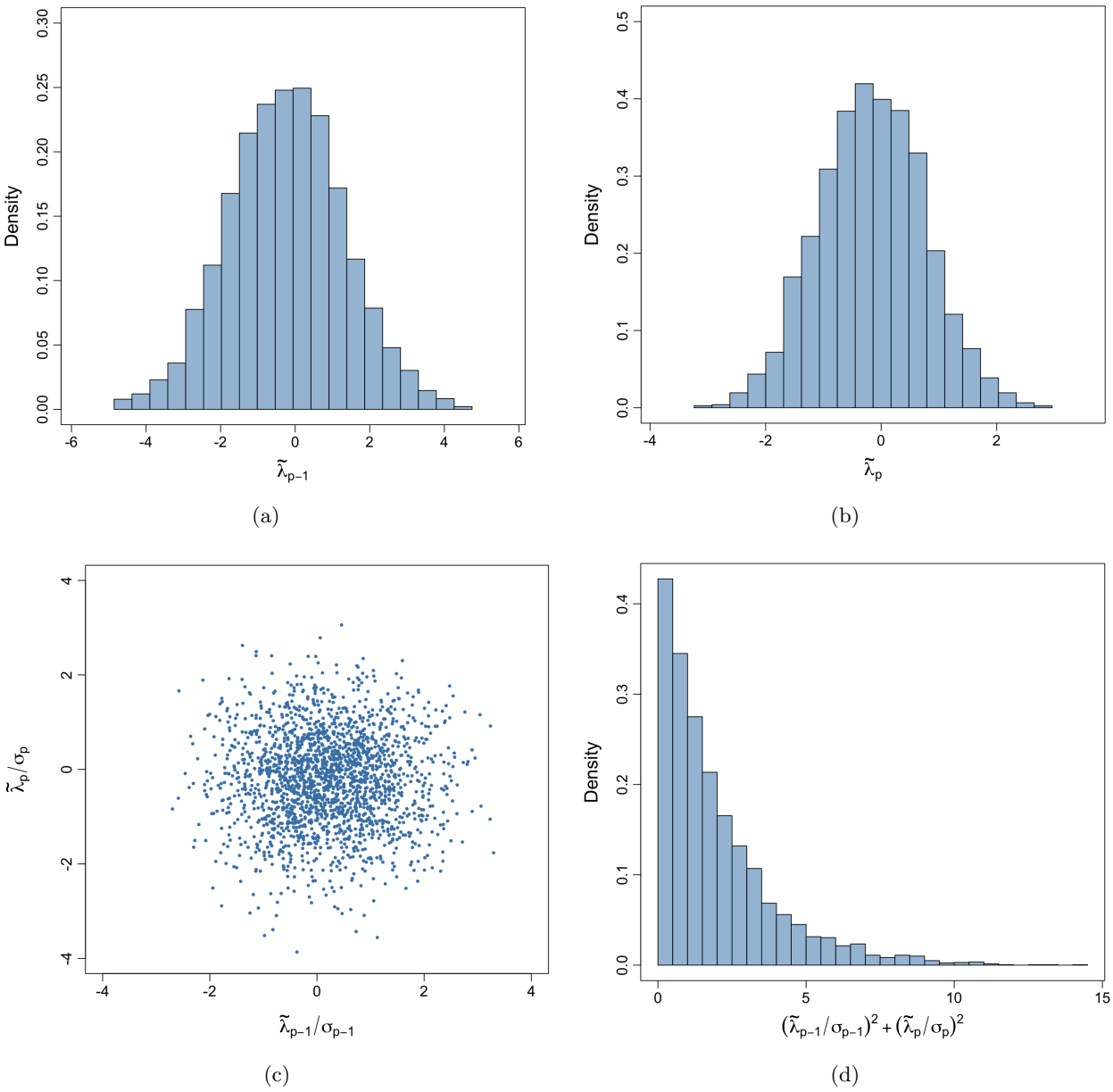


Figure 10: **Eigenvalue Asymptotics for** $0 < H_1 < H_2 < 0.75$, generated using the code in listing 10, with $n = 2^{16}$, $h = 2^{12}$, $p = 50$, $H_1 = 0.3$, $H_2 = 0.7$, and $R = 2000$ iterations. The empirical distributions of $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$ indicate asymptotic Gaussianity, while the cloud-like formation with correlation $\rho = 0.02$ suggests asymptotic independence. The sum of normalized squares $(\widetilde{\lambda}_{p-1}/\sigma_{p-1})^2 + (\widetilde{\lambda}_p/\sigma_p)^2$ closely matches the χ^2_2 distribution, confirming asymptotic independence and Gaussianity.

Proof. Let $M_x(h)$ be as in (5.5). Then write

$$M_x(h) = \begin{bmatrix} M_{x,11}(h) & M_{x,12}(h) \\ M_{x,12}(h) & M_{x,22}(h) \end{bmatrix}.$$

We then consider the compensated matrix $\widehat{B}(h)$ as in (5.9), where the normalization removes the scaling

induced by the Hurst parameters, resulting in

$$\widehat{B}(h) = \begin{bmatrix} \frac{M_{x,11}(h)}{h^{2H_1}} & \frac{M_{x,12}(h)}{h^{H_1+H_2}} \\ \frac{M_{x,12}(h)}{h^{H_1+H_2}} & \frac{M_{x,22}(h)}{h^{2H_2}} \end{bmatrix}. \tag{5.10}$$

By an extension of the analysis in Table 1 of Zhang et al. (2018) to the joint case, we know that the normalized entries of $M_x(h)$ jointly satisfy

$$\sqrt{\frac{n}{h}} \left(\begin{bmatrix} \frac{M_{x,11}(h)}{h^{2H_1}} - \mathbb{E}\left[\frac{M_{x,11}(h)}{h^{2H_1}}\right] \\ \frac{M_{x,12}(h)}{h^{H_1+H_2}} - \mathbb{E}\left[\frac{M_{x,12}(h)}{h^{H_1+H_2}}\right] \\ \frac{M_{x,22}(h)}{h^{2H_2}} - \mathbb{E}\left[\frac{M_{x,22}(h)}{h^{2H_2}}\right] \end{bmatrix} \right) \xrightarrow{d} \mathcal{N}(0, \Sigma_B), \quad n \to \infty \tag{5.11}$$

where Σ is the asymptotic covariance matrix.
Now let $B(h) = \mathbb{E}[\widehat{B}(h)]$, $h > 0$. We define the scale-compensated covariance matrix

$$\begin{aligned} \widetilde{\Sigma}_q(h, B) &= \frac{1}{h^{2H_q}} \left[Ph^{\mathbf{H}} B h^{\mathbf{H}^\top} P^\top + M_z(h) + Ph^{\mathbf{H}} h^{-\mathbf{H}} M_{x,z}(h) + M_{x,z}(h)^\top h^{-\mathbf{H}^\top} h^{\mathbf{H}^\top} P^\top \right] \\ &= \frac{1}{h^{2H_q}} \left[Ph^{\mathbf{H}} B h^{\mathbf{H}^\top} P^\top + M_z(h) + Ph^{\mathbf{H}} O_{\mathbb{P}}(1) + O_{\mathbb{P}}^\top(1) h^{\mathbf{H}^\top} P^\top \right], \end{aligned}$$

where $O_{\mathbb{P}}(1) \in \mathbb{R}^{2\times p}$ is a random matrix whose norm is bounded in probability, and $B \in \mathscr{S}_{\geqslant 0}(2, \mathbb{R})$. Additionally, let $f_{n,q}(B) = \log \lambda_q(\widetilde{\Sigma}_q(h, B))$. By equation (5.105) in Abry et al. (2024), using a Taylor expansion centered at $B(h)$, we have

$$\sqrt{\frac{n}{h}} \left(f_{n,q}(\widehat{B}(h)) - f_{n,q}(B(h)) \right) = \sum_{1 \leqslant \ell \leqslant \ell' \leqslant 2} \frac{\partial}{\partial b_{\ell\ell'}} f_{n,q}(\breve{B}(h)) \cdot \sqrt{\frac{n}{h}}\, \pi_{\ell\ell'} \left(\widehat{B}(h) - B(h) \right) + o_P(1) \tag{5.12}$$

where $\breve{B}(h)$ is a convex combination of $\widehat{B}(h)$ and $B(h)$, and $\pi_{\ell\ell'}$ projects onto the (ℓ, ℓ')-th entry of a symmetric matrix. By the same argument as in the proof of Theorem 3.2 in Abry et al. (2024), we have

$$\frac{\partial}{\partial b_{\ell\ell'}} f_{n,q}(\breve{B}(h)) = \frac{\frac{\partial}{\partial b_{\ell\ell'}} \lambda_q(\widetilde{\Sigma}_q(h, \breve{B}(h)))}{\lambda_q(\widetilde{\Sigma}_q(h, \breve{B}(h)))} \xrightarrow{\mathbb{P}} c_{\ell\ell'}, \quad n \to \infty, \tag{5.13}$$

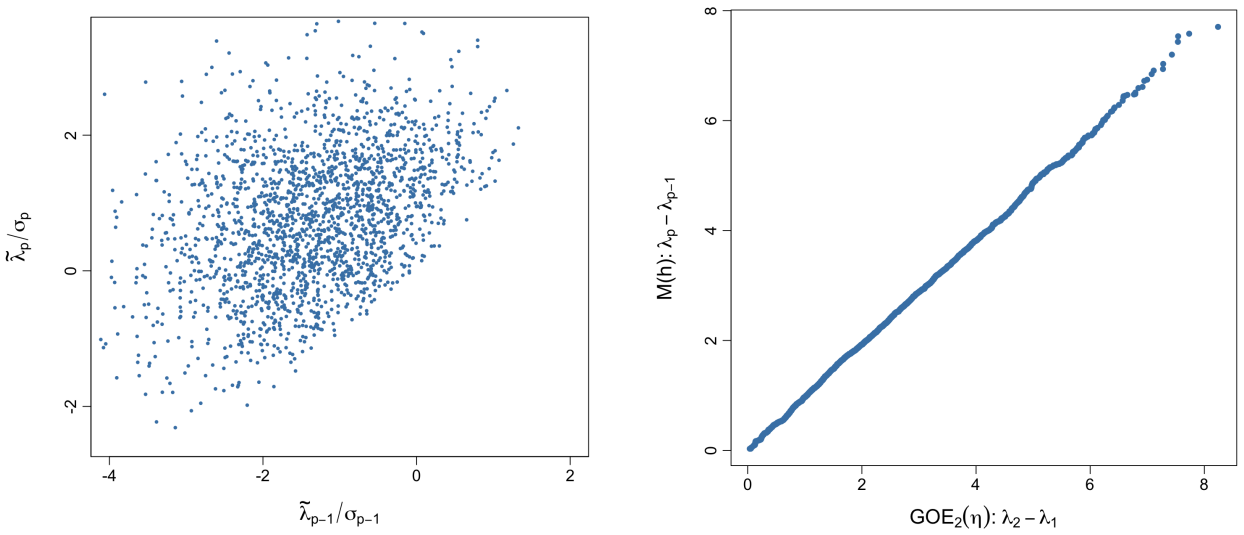
for some constants $c_{\ell\ell'}$. From Equations (5.11), (5.12), and (5.13), and applying Slutsky's theorem (see Theorem A.1), it follows that (5.1) holds. This establishes the theorem. □

The following theorem is the second of the two main mathematical results of this thesis.

Theorem 5.2. *Suppose $0 < H_1 = H_2 < 0.75$. In the high-dimensional, large-scale regime where $n, p, h \to \infty$ with $\frac{ph}{n} = O(1)$, the top two eigenvalues of $M(h)$ are asymptotically non-Gaussian and GOE-like. That is, $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$, as defined in (5.4), satisfy*

$$\begin{pmatrix} \widetilde{\lambda}_{p-1} \\ \widetilde{\lambda}_p \end{pmatrix} \xrightarrow{d} \begin{pmatrix} \lambda_1(\mathrm{GOE}_2(\eta)) \\ \lambda_2(\mathrm{GOE}_2(\eta)) \end{pmatrix}$$

where $\mathrm{GOE}_2(\eta)$ denotes a 2×2 Gaussian Orthogonal Ensemble matrix scaled by a variance factor $\eta > 0$.



(a) (b)

Figure 11: **Eigenvalue Asymptotics for** $0 < H_1 = H_2 < 0.75$, generated using the code in listing 10, with $n = 2^{16}$, $h = 2^{12}$, $p = 50$, $H_1 = H_2 = 0.6$, and $R = 2000$ iterations. The tilted, uneven cloud formation with correlation $\rho = 0.37$ indicates asymptotic dependence, and the straight-line fit of the QQ plot comparing $\widetilde{\lambda}_p - \widetilde{\lambda}_{p-1}$ to $\lambda_2 - \lambda_1$ of $\mathrm{GOE}_2(\eta)$ confirms GOE-like repulsion. The observed dependence and repulsion together suggest that $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$ are not asymptotically Gaussian.

Proof. Let $H := H_1 = H_2$. First, we show that

$$\left\{\sqrt{\frac{n}{h}} \cdot \left(\lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right)\right\}_{q=p-1,p} \xrightarrow{d} \mathcal{N}(\mathbf{0}, \Sigma), \quad n \to \infty, \tag{5.14}$$

for some $\Sigma_{>0}(2, \mathbb{R})$.

In fact, let $M_x(h), M_z(h), M_{x,z}(h)$ and $\widehat{B}(h)$ be defined as in (5.5), (5.6), (5.7), and (5.9), respectively. Also, let $\widetilde{\Sigma}_q(h, \widehat{B}(h))$ be as in (5.2.1). Using $M_z(h) = O_{\mathbb{P}}(1)$ and $h^{-\mathbf{H}} M_{x,z}(h) = O_{\mathbb{P}}(1)$, it follows that

$$\frac{1}{h^{2H}} M_z(h) = o_{\mathbb{P}}(1), \quad \frac{1}{h^{2H}} P h^{\mathbf{H}} h^{-\mathbf{H}} M_{x,z}(h) = o_{\mathbb{P}}(1).$$

Thus,

$$\widetilde{\Sigma}_q(h, \widehat{B}(h)) \overset{\mathbb{P}}{\sim} P\widehat{B}(h)P^\top.$$

In addition,

$$\frac{\mathbb{E}M(h)}{h^{2H}} = P\frac{\mathbb{E}M_x(h)}{h^{2H}}P^\top,$$

since $\mathbb{E}M_{x,z}(h) = 0$ and $\mathbb{E}M_z(h)/h^{2H} = O(1)$. Consequently, we can write

$$\left\{\sqrt{\frac{n}{h}} \cdot \left(\lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right)\right\}_{q=p-1,p} \overset{\mathbb{P}}{\sim} \sqrt{\frac{n}{h}} \left\{\lambda_q\left(P\widehat{B}(h)P^\top\right) - \lambda_q\left(P\frac{\mathbb{E}M_x(h)}{h^{2H}}P^\top\right)\right\}_{q=p-1,p}$$

$$= \sqrt{\frac{n}{h}} \left\{\lambda_q\left(\widehat{B}(h)\right) - \lambda_q\left(\frac{\mathbb{E}M_x(h)}{h^{2H}}\right)\right\}_{q=1,2} = \sqrt{\frac{n}{h}} \left\{\lambda_q\left(\widehat{B}(h)\right) - \lambda_q\left(I\right)\right\}_{q=1,2},$$

where the first equality is a consequence of the assumptions on P (**n.b.**: $\widehat{B}(h), \mathbb{E}M_x(h) \in \mathbb{R}^{2\times 2}$). Using the

decomposition $\widehat{B}(h) = O\,\mathrm{diag}(\lambda_{B,1}, \lambda_{B,2})\,O^\top$, we have

$$\sqrt{\frac{n}{h}}\left(\lambda_q\left(\widehat{B}(h)\right) - 1\right) = \sqrt{\frac{n}{h}} \cdot \lambda_q\left(O\left(\begin{bmatrix}\lambda_{B,1} - 1 & \cdot \\ \cdot & \lambda_{B,2} - 1\end{bmatrix}\right) O^\top\right)$$

$$= \sqrt{\frac{n}{h}} \cdot \lambda_q\left(\begin{bmatrix}\lambda_{B,1} - 1 & \cdot \\ \cdot & \lambda_{B,2} - 1\end{bmatrix}\right). \tag{5.15}$$

However, we can show that

$$\sqrt{\frac{n}{h}} \cdot \mathrm{vec}_{\mathscr{S}}\left(\frac{\widehat{B}(h)}{h^{2H}} - I\right) \xrightarrow{d} \mathcal{N}(0, \Sigma) \tag{5.16}$$

where $\widehat{B}(h)_{11}, \widehat{B}(h)_{12}, \widehat{B}(h)_{22}$ are asymptotically independent (see Remark 5.2). Since the joint Gaussian limit is symmetric, the eigenvalues are distributed as those of a GOE matrix (up to scaling). Thus, by (5.15) and (5.16), we conclude that (5.14) holds.

We now return to the full expression (5.4), involving log-transformed eigenvalues. For $q \in \{p-1, p\}$, we begin with

$$\sqrt{\frac{n}{h}}\left(\log \lambda_q(M(h)) - \log \lambda_q(\mathbb{E}M(h))\right) = \sqrt{\frac{n}{h}}\left(\log \lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \log \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right).$$

Using a first-order Taylor expansion for the logarithm around $\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)$, we obtain

$$\sqrt{\frac{n}{h}} \cdot \left(\lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right) \cdot \frac{1}{\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)} + o\left(\lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right) \cdot \sqrt{\frac{n}{h}}, \tag{5.17}$$

where the term

$$o\left(\lambda_q\left(\frac{M(h)}{h^{2H}}\right) - \lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)\right) \cdot \sqrt{\frac{n}{h}} = o_{\mathbb{P}}(1).$$

Moreover, note that

$$\frac{1}{\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)} \to C > 0 \quad \text{as } n, h \to \infty.$$

Thus, (5.17) is asymptotically equal to the derivative of the log evaluated at $\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)$, scaled by the fluctuation of λ_q. However, recall from (5.8) that

$$\mathbb{E}\left(\frac{M(h)}{h^{2H}}\right) = P \cdot \frac{\mathbb{E}M_x(h)}{h^{2H}} \cdot P^\top + \frac{\mathbb{E}M_z(h)}{h^{2H}},$$

where

$$\frac{\mathbb{E}M_x(h)}{h^{2H}} = \mathbb{E}\widehat{B}(h) \sim P\begin{bmatrix}c_1 & 0 \\ 0 & c_2\end{bmatrix}P^\top, \quad \text{and} \quad \frac{\mathbb{E}M_z(h)}{h^{2H}} \to 0 \quad \text{as } n, h \to \infty.$$

So then

$$\mathbb{E}\left(\frac{1}{n-h}\sum_{j=1}^{n-h}\left(B_H(j+h) - B_H(j)\right)^2\right) = \mathbb{E}\left(\left(B_H(j+h) - B_H(j)\right)^2\right) = h^{2H},$$

which implies

$$\mathbb{E}\left(\frac{1}{h^{2H}} \cdot \frac{1}{n-h}\sum_{j=1}^{n-h}\left(B_H(j+h) - B_H(j)\right)^2\right) = 1.$$

This confirms that the normalization factor converges to 1 in expectation. Then, combining the Taylor

expansion in (5.17) with the fluctuation result in (5.16), we obtain

$$\sqrt{\frac{n}{h}}\left(\log\lambda_q(M(h)) - \log\lambda_q(\mathbb{E}M(h))\right) \xrightarrow{d} \frac{1}{\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right)} \cdot \lambda_q(\mathrm{GOE}_2(\eta)),$$

where $\lambda_q\left(\frac{\mathbb{E}M(h)}{h^{2H}}\right) \to C > 0$ as $n, h \to \infty$. Thus, by Slutsky's theorem as given in Theorem A.1, it follows that Theorem 5.2 holds. □

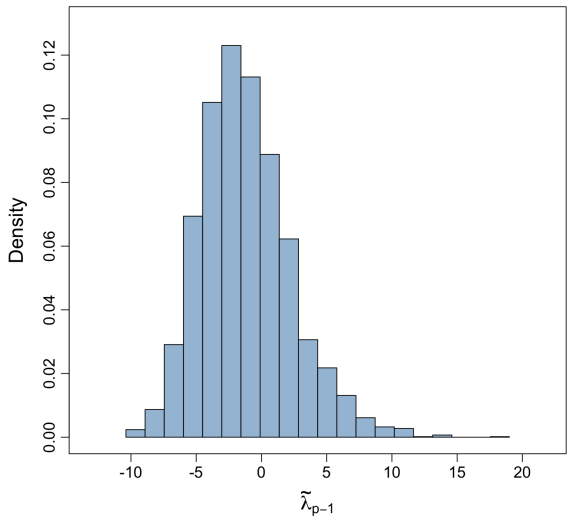
Remark 5.2. Proving (5.16) requires using the Isserlis theorem (e.g., Michalowicz et al. (2009)).

Remark 5.3. In both Theorems 5.1 and 5.2, for the sake of clarity we simply assumed that $ph/n = O(1)$. The precise rate of divergence among n, h and p needed for those results can be obtained as in Abry et al. (2024) (cf. Assumption $(A4)$ in that paper).

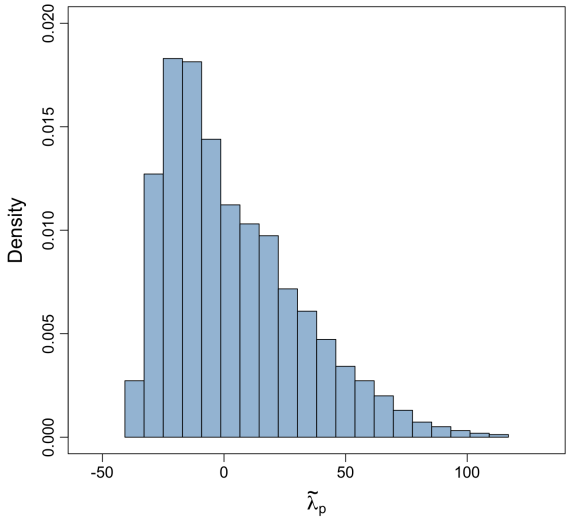
5.2.2 Post-Phase Transition

When $H_2 > 0.75$, the squared summability condition in (5.3) no longer holds. Consequently, the results of Theorems 5.1 and 5.2 do not apply. Nevertheless, the asymptotic fluctuations of the top eigenvalues of $M(h)$, as described in (5.4), remain well-structured, but no longer follow Gaussian or GOE-type limits. The following post-phase transition simulations are done with $p = 2$, but we are confident the results extend to high dimensions.

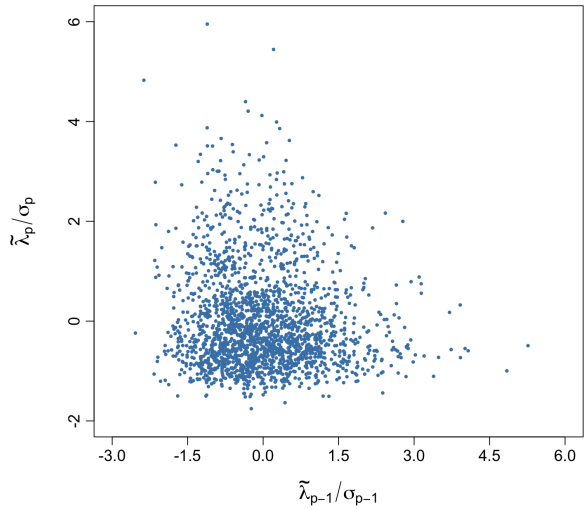
Conjecture 5.1. *Suppose* $0.75 < H_1 < H_2 < 1$. *In the high-dimensional, large-scale regime, the top two eigenvalues of* $M(h)$*, denoted* $\widetilde{\lambda}_{p-1}$ *and* $\widetilde{\lambda}_p$ *as in* (5.4)*, are asymptotically independent and non-Gaussian.*



(a)



(b)



(c)

Figure 12: **Eigenvalue Asymptotics for** $0.75 < H_1 < H_2 < 1$, generated using the code in listing 11, with $n = 2^{16}$, $h = 2^{12}$, $p = 2$, $H_1 = 0.8$, $H_2 = 0.9$, and $R = 2000$ iterations. The empirical distributions of $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$ demonstrate asymptotic non-Gaussian behavior, while the even, cloud-like formation with correlation $\rho = 0.06$ indicates asymptotic independence.

Conjecture 5.2. *Suppose* $0.75 < H_1 = H_2 < 1$. *In the high-dimensional, large-scale regime, the top two eigenvalues of* $M(h)$, *denoted* $\widetilde{\lambda}_{p-1}$ *and* $\widetilde{\lambda}_p$ *as in* (5.4), *are asymptotically non-Gaussian and exhibit non-GOE-like repulsion.*

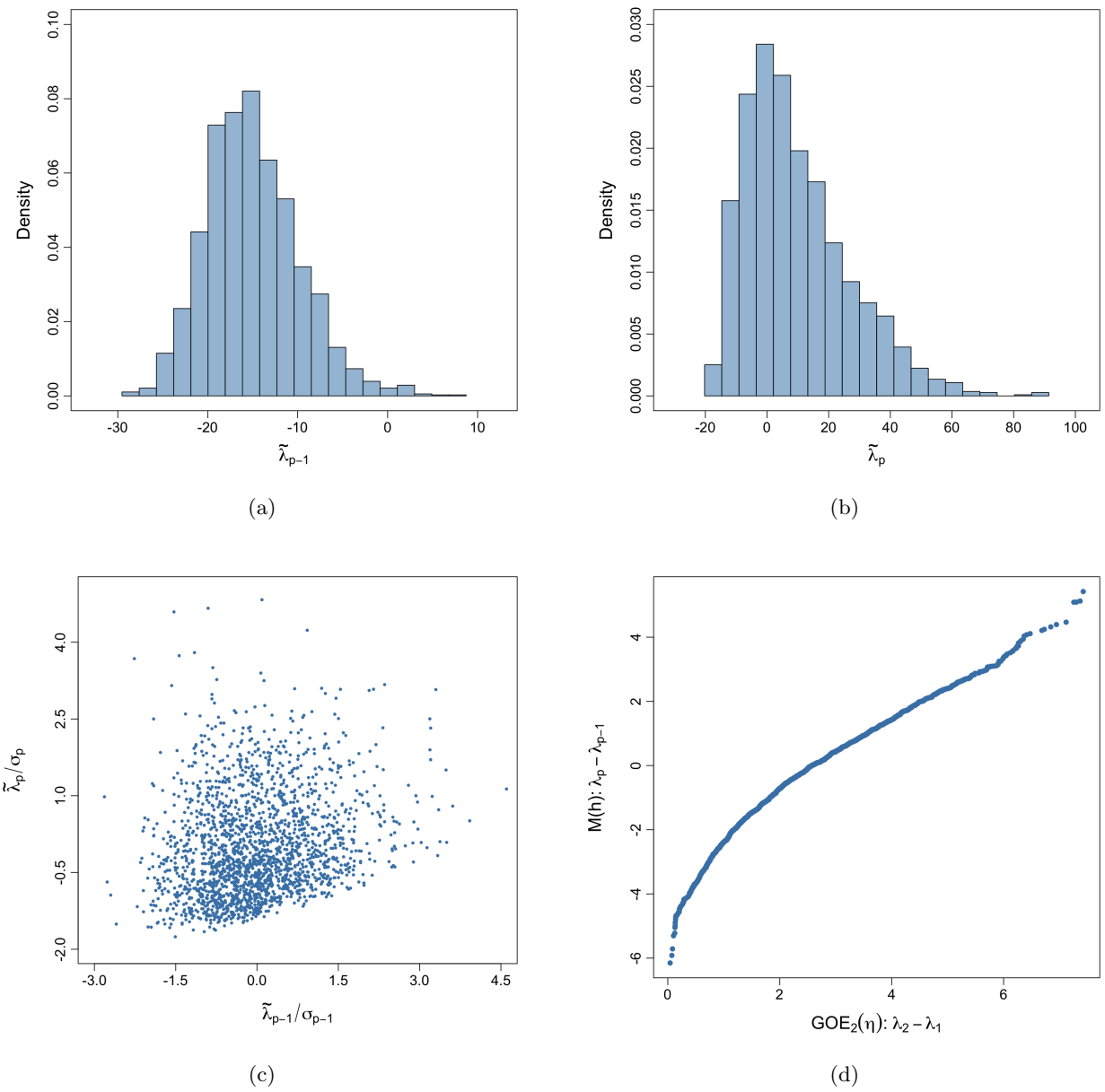
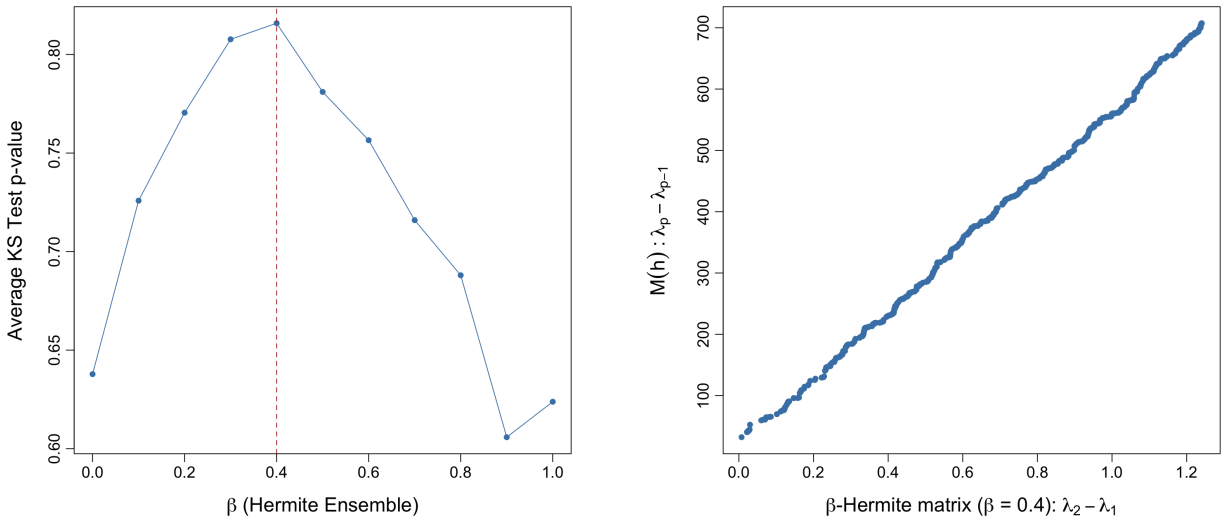


Figure 13: **Eigenvalue Asymptotics for** $0.75 < H_1 = H_2 < 1$, generated using the code in listing 11, with $n = 2^{16}$, $h = 2^{12}$, $p = 2$, $H_1 = H_2 = 0.9$, and $R = 2000$ iterations. The empirical distributions of $\widetilde{\lambda}_{p-1}$ and $\widetilde{\lambda}_p$ demonstrate asymptotic non-Gaussian behavior, while the tilted, uneven cloud formation with correlation $\rho = 0.32$ indicates asymptotic dependence. The curved QQ plot shows a poor fit to $\text{GOE}_2(\eta)$, indicating non-GOE-like repulsion.

We observe that in the high-dimensional, large-scale regime, when $0.75 < H_1 = H_2 < 1$, the top two eigenvalues of $M(h)$ exhibit dependence, but not GOE-like repulsion. Since the GOE corresponds to the β-Hermite ensemble with $\beta = 1$, it is natural to ask which value of β best approximates the eigenvalue spacing of $M(h)$ in this regime. To address this, we perform the Kolmogorov–Smirnov test (which measures the similarity between two empirical distributions) between the eigenvalue spacing of $M(h)$ and that of β-Hermite ensembles across a range of β values, selecting the β that maximizes the K-S test p-value. This optimal β best approximates the $M(h)$ eigenvalue repulsion in this regime. Because a single β-Hermite ensemble is unlikely to capture the full distribution of eigenvalue spacings in this regime, we focus on only

the bottom 30% of values.



(a) (b)

Figure 14: **Best-Fit β Estimation**, generated using the code in listings 12 and 13, with $n = 2^{16}$, $h = 2^{12}$, $p = 2$, $H_1 = H_2 = 0.9$, and $R = 2000$ iterations. The Kolmogorov–Smirnov p-values peak at $\beta \approx 0.4$, indicating the optimal fit. The QQ plot comparing the bottom 30% of $\lambda_p - \lambda_{p-1}$ of $M(h)$ to $\lambda_2 - \lambda_1$ of a 2×2 β-Hermite matrix with $\beta = 0.4$ has a straight-line fit, confirming that $\beta \approx 0.4$ best captures the eigenvalue repulsion of $M(h)$ when $0.75 < H_1 = H_2 < 1$.

A Auxiliary results

Lemma A.1 (Borel-Cantelli Lemma)**.** *Let $\{A_n\}_{n\in\mathbb{N}}$ be a sequence of events.*

1. *If $\sum_{n=1}^{\infty}\mathbb{P}(A_n) < \infty$, then $\mathbb{P}(A_n \text{ i.o.}) = 0$.*
2. *If the A_n's are independent and $\sum_{n=1}^{\infty}\mathbb{P}(A_n) = \infty$, then $\mathbb{P}(A_n \text{ i.o.}) = 1$.*

Proof. (Folland, 1999, p.321) □

Lemma A.2 (Orthogonal Mapping to e_1)**.** *For any vector $\mathbf{y} \in \mathbb{R}^n$, there exists an orthogonal matrix $O \in O(n)$ such that $O(\frac{\mathbf{y}}{||\mathbf{y}||}) = e_1 = (1, 0, \ldots, 0)^T$*

Proof. Let $\mathbf{w}$ be the unit vector $\frac{y}{||y||}$. Define the first row vector $\mathbf{o}_1$ to be $\mathbf{w}^T$. Now, let the row vectors $\mathbf{o}_2, \ldots, \mathbf{o}_p$ be an orthonormal basis of $\mathbf{w}^{\perp}$. Then, define O to be the matrix whose p rows are given by $\mathbf{o}_1, \ldots, \mathbf{o}_p$. Thus, the orthogonal matrix O exists such that $O\mathbf{w} = e_1$ □

Theorem A.1 (Slutsky's Theorem)**.** *Let $X_n \xrightarrow{d} X$ and $Y_n \xrightarrow{\mathbb{P}} c$ for some constant $c \in \mathbb{R}$. Then:*

1. *$X_n + Y_n \xrightarrow{d} X + c$*
2. *$X_n Y_n \xrightarrow{d} cX$*
3. *If $c \neq 0$ and $\mathbb{P}(Y_n \neq 0) = 1$, then $\frac{X_n}{Y_n} \xrightarrow{d} \frac{X}{c}$*

Proof. (van der Vaart, 2000, p.10) □

B Code

Listing 1: Chebyshev's Weak Law of Large Numbers

```
n <- 10000
mu <- 5
lambda <- 1 / mu
sigma <- 2

normal_samples <- rnorm(n, mean = mu, sd = sigma)
exp_samples <- rexp(n, rate = lambda)

cummean_normal <- cumsum(normal_samples) / seq_len(n)
cummean_exp <- cumsum(exp_samples) / seq_len(n)

plot(cummean_normal, type = "l", col = "#4682B4", lwd = 1.5,
     ylim = c(mu - 1, mu + 1),
     xlab = "Sample Size (n)",
     ylab = expression("Sample Mean (" * bar(X)[n] * ")"))
lines(cummean_exp, col = "#B22222", lwd = 1.5)
legend("topright",
       legend = c("Normal", "Exponential"),
       col = c("#4682B4", "#B22222"),
       lwd = 1.5,
       bg = "white",
       box.lwd = 1)
```

Listing 2: Convergence of $\hat{\Sigma}_n$

```
library(parallel)

n <- 5000
p <- 10
num_cores <- detectCores() - 1
RNGkind("L'Ecuyer-CMRG")

compute_spectral_radius <- function(mat) {
  max(abs(eigen(mat, only.values = TRUE)$values))
}

spectral_radii <- numeric(n)

spectral_radii[2:n] <- unlist(mclapply(2:n, function(n_val) {
  total <- 0
  for (i in seq_len(n_val)) {
    X <- matrix(rnorm(n_val * p), nrow = n_val)
    Sigma_hat <- cov(X)
    Delta_n <- Sigma_hat - diag(p)
    if (any(is.infinite(Delta_n)) || any(is.na(Delta_n))) next
    total <- total + compute_spectral_radius(Delta_n)
  }
  total / n_val
}, mc.cores = num_cores))
```

```
plot(2:n, spectral_radii[2:n],
     type = "l", col = "#4682B4", lwd = 1.5,
     xlab = "Sample Size (n)",
     ylab = expression("Spectral Radius of " * Delta[n]))
```

Listing 3: Wishart Eigenvalue Repulsion

```
library(MASS)

n <- 250
p <- 10
Sigma <- diag(p)
num_bins <- 25

simulate_wishart <- function(n, p) {
  X <- mvrnorm(n, mu = rep(0, p), Sigma = Sigma)
  S <- t(X) %*% X
  eigenvalues <- eigen(S, only.values = TRUE)$values
  return(eigenvalues)
}

eigenvalues <- as.vector(replicate(1000, simulate_wishart(n, p)))

range <- range(eigenvalues)
breaks_seq <- seq(range[1], range[2], length.out = num_bins + 1)

densities <- hist(
  eigenvalues,
  breaks = breaks_seq,
  plot = FALSE,
  probability = TRUE
)

hist(
  eigenvalues,
  breaks = breaks_seq,
  probability = TRUE,
  ylim = c(0, 1.1 * max(densities$density)),
  main = "",
  xlab = "Wishart Bulk Eigenvalues",
  ylab = "Density",
  col = "#A3C1DA",
  border = "black",
  axes = FALSE,
  cex.axis = 1.2,
  cex.lab = 1.3
)

axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")
```

Listing 4: Asymptotic Gaussianity in Low Dimensions

```
n <- 5000
p <- 2
R <- 2000
num_bins <- 20

simulate_eigenvalues <- function(n, R) {
  lambda2 <- numeric(R)
  for (i in 1:R) {
    X <- matrix(rnorm(n * p, mean = 0, sd = sqrt(c(1, 2))), nrow = n, ncol
        = p, byrow = TRUE)
    Sigma_hat <- (t(X) %*% X) / n
    lambda2[i] <- max(eigen(Sigma_hat, symmetric = TRUE)$values)
  }
  sqrt(n) * (lambda2 - 2)
}

asymp_lambda2 <- simulate_eigenvalues(n, R)

range <- range(asymp_lambda2)
breaks_seq <- seq(range[1], range[2], length.out = num_bins + 1)
densities <- hist(asymp_lambda2, breaks = breaks_seq, plot = FALSE,
   probability = TRUE)

hist(
  asymp_lambda2,
  breaks = breaks_seq,
  probability = TRUE,
  ylim = c(0, 1.1 * max(densities$density)),
  xlab = expression(tilde(lambda)[2]),
  ylab = "Density",
  main = "",
  col = "#A3C1DA",
  border = "black",
  axes = FALSE,
  cex.axis = 1.2,
  cex.lab = 1.3
)

axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")
```

Listing 5: GOE-like Repulsion in Low Dimensions

```
library(MASS)
library(ggplot2)
library(parallel)

n <- 5000
p <- 2
R <- 2000
RNGkind("L'Ecuyer-CMRG")
```

```
lambda_diff_SCM <- function() {
  X <- matrix(rnorm(n * p), nrow = n)
  Sigma_hat <- (1 / n) * t(X) %*% X
  lambda <- sort(eigen(Sigma_hat, symmetric = TRUE)$values)
  sqrt(n) * (lambda[2] - lambda[1])
}

lambda_diff_GOE <- function() {
  M <- matrix(rnorm(p^2), nrow = p)
  GOE <- sqrt(p) * 0.5 * (M + t(M))
  lambda <- sort(eigen(GOE, symmetric = TRUE)$values)
  lambda[2] - lambda[1]
}

num_cores <- detectCores() - 1

scm <- unlist(mclapply(seq_len(R), function(i) lambda_diff_SCM(), mc.cores
    = num_cores))
goe <- unlist(mclapply(seq_len(R), function(i) lambda_diff_GOE(), mc.cores
    = num_cores))

qqplot(goe, scm,
       xlab = expression(GOE[2](eta) * ":" ~ lambda[2] - lambda[1]),
       ylab = expression(hat(Sigma)[n] * ":" ~ lambda[2] - lambda[1]),
       pch = 16,
       col = "#4682B4",
       cex.lab = 1.5,
       cex.axis = 1.2,
       axes = FALSE)

axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")
```

Listing 6: Marchenko-Pastur Distribution

```
library(MASS)
library(dplyr)
library(purrr)
library(ggplot2)
library(gridExtra)
library(grid)

n <- 1000
c_vals <- c(1/10, 1/2, 1, 2)
num_bins <- 20

simulate_wishart <- function(n, p) {
  Sigma <- diag(p)
  X <- mvrnorm(n, mu = rep(0, p), Sigma = Sigma)
  S <- t(X) %*% X
  eigen(S, only.values = TRUE)$values
}
```

```
results <- list()

for (c in c_vals) {
  p <- floor(c * n)
  eigenvalues <- replicate(1000, simulate_wishart(n, p))
  eigenvalues_df <- data.frame(
    value = as.vector(eigenvalues),
    c = as.factor(rep(c, length(eigenvalues)))
  )
  results[[as.character(c)]] <- eigenvalues_df
}

combined_df <- do.call(rbind, results)

bin_info <- combined_df %>%
  group_by(c) %>%
  summarize(
    min_value = min(value),
    max_value = max(value),
    range_value = max_value - min_value,
    .groups = "drop"
  ) %>%
  mutate(
    margin = 0.1 * range_value,
    xlim_low = min_value - margin,
    xlim_high = max_value + margin,
    breaks_list = map2(min_value, max_value, ~ seq(.x, .y, length.out =
       num_bins + 1))
  )

combined_split <- split(combined_df, combined_df$c)

get_max_hist_density <- function(df, breaks) {
  h <- hist(df$value, breaks = breaks, plot = FALSE)
  max(h$density)
}

density_max_df <- map_df(names(combined_split), function(c_val) {
  df_sub <- combined_split[[c_val]]
  breaks_sub <- bin_info %>% filter(c == c_val) %>% pull(breaks_list) %>%
     .[[1]]
  ymax <- 1.1 * get_max_hist_density(df_sub, breaks_sub)
  data.frame(c = c_val, ymax = ymax)
})

plots <- lapply(names(combined_split), function(c_val) {
  df_sub <- combined_split[[c_val]]
  breaks_sub <- bin_info %>% filter(c == c_val) %>% pull(breaks_list) %>%
     .[[1]]
  ymax_sub <- density_max_df %>% filter(c == c_val) %>% pull(ymax)

  ggplot(df_sub, aes(x = value)) +
    geom_histogram(
      aes(y = after_stat(density)),
```

```
      breaks = breaks_sub,
      color = "black",
      fill = "#4682B4",
      alpha = 0.5,
      position = "identity"
    ) +
    coord_cartesian(ylim = c(0, ymax_sub)) +
    labs(x = NULL, y = NULL) +
    ggtitle(paste("c =", c_val)) +
    theme(
      panel.background = element_blank(),
      panel.grid = element_blank(),
      panel.border = element_rect(color = "black", fill = NA, size = 0.5),
      axis.line = element_line(color = "black"),
      axis.text = element_text(size = 12),
      axis.title = element_text(size = 13),
      plot.title = element_text(size = 15, hjust = 0.5),
      plot.margin = margin(5.5, 5.5, 5.5, 5.5, unit = "pt")
    )
})

arranged_plots <- function(...) {
  plots <- list(...)
  grid.arrange(
    arrangeGrob(
      grobs = plots,
      ncol = 2,
      nrow = 2
    ),
    bottom = textGrob(
      expression("Eigenvalue (" * lambda * ")"),
      gp = gpar(fontsize = 1.5 * 11)
    ),
    left = textGrob(
      "Density", rot = 90,
      gp = gpar(fontsize = 1.5 * 11)
    )
  )
}

arranged_plots(plots[[1]], plots[[2]], plots[[3]], plots[[4]])
```

Listing 7: Tracy-Widom Distribution

```
library(MASS)
library(ggplot2)
library(parallel)

n <- 3000
p <- n / 2
c <- p / n
R <- 2000
num_bins <- 17
```

```
top_eigenvalue <- function(X) {
  Sigma_hat <- (1 / n) * t(X) %*% X
  max(abs(eigen(Sigma_hat, only.values = TRUE)$values))
}


values <- unlist(mclapply(seq_len(R), function(i) {
  X <- matrix(rnorm(n * p), nrow = n, ncol = p)
  top_eigenvalue(X)
}, mc.cores = num_cores))

scaled <- (values - mean(values)) / sd(values)

range <- range(scaled)
breaks_seq <- seq(range[1], range[2], length.out = num_bins + 1)
densities <- hist(scaled, breaks = breaks_seq, plot = FALSE, probability =
    TRUE)

par(mfrow = c(1, 1), mar = c(5, 5.5, 2, 1), mgp = c(3.1, 0.7, 0))
hist(
  scaled,
  breaks = breaks_seq,
  probability = TRUE,
  ylim = c(0, 1.1 * max(densities$density)),
  xlim = c(min(breaks_seq), max(breaks_seq)),
  xlab = expression("Normalized " * lambda[p]),
  ylab = "Density",
  main = "",
  col = "#A3C1DA",
  border = "black",
  axes = FALSE,
  cex.axis = 1.2,
  cex.lab = 1.5
)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")

par(mar = c(5, 5.5, 2, 1), mgp = c(3.1, 0.7, 0))
qqnorm(
  scaled,
  main = "",
  xlab = "N(0, 1) Sample",
  ylab = expression("Normalized" ~ lambda[p]),
  col = "#4682B4",
  pch = 16,
  axes = FALSE,
  cex.lab = 1.5,
  cex.axis = 1.2
)
qqline(scaled, col = "#4682B4", lwd = 2, lty = 2)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")
```

Listing 8: fBm Paths

```
library(fracdiff)
library(ggplot2)
library(dplyr)
library(grid)
library(longmemo)

n <- 2^12
hurst_values <- c(0.2, 0.5, 0.8)

simulate_fbm <- function(n, H) {
  fgn <- simFGN0(n, H)
  cumsum(fgn)
}

data <- data.frame(
  time = rep(seq_len(n), times = length(hurst_values)),
  value = unlist(lapply(hurst_values, function(H) simulate_fbm(n, H))),
  hurst = factor(rep(hurst_values, each = n), levels = c(0.8, 0.5, 0.2))
)

levels(data$hurst) <- c("H = 0.8", "H = 0.5", "H = 0.2")

ggplot(data, aes(x = time, y = value)) +
  geom_line(size = 0.4, color = "#4682B4") +
  facet_wrap(~ hurst, ncol = 1, scales = "free") +
  labs(
    x = expression("Time " * (t)),
    y = expression(B[H](t))
  ) +
  theme_minimal(base_size = 14) +
  theme(
    legend.position = "none",
    strip.text = element_text(size = 13, face = "plain", hjust = 0.5),
    strip.placement = "outside",
    panel.spacing = unit(1.5, "lines"),
    panel.border = element_rect(color = "black", fill = NA, linewidth = 1)
      ,
    plot.title = element_blank(),
    panel.grid = element_blank(),
    strip.background = element_blank()
  )
```

Listing 9: $M(h)$ Scaling Exponent Estimation

```
library(fracdiff)
library(ggplot2)
library(gridExtra)
library(grid)

n <- 2^16
h_values <- c(2^4, 2^6, 2^8)
H1 <- 0.25
```

```
H2 <- 0.75
theta <- pi / 3
P <- matrix(c(cos(theta), -sin(theta), sin(theta), cos(theta)), nrow = 2)

simulate_fbm <- function(n, H) {
  cumsum(fracdiff.sim(n, d = H - 0.5)$series)
}

simulate_Y <- function(n, H1, H2, P) {
  B1 <- simulate_fbm(n, H1)
  B2 <- simulate_fbm(n, H2)
  Y <- P %*% rbind(B1, B2)
  t(Y)
}

calculate_M <- function(Y, h) {
  n_h <- nrow(Y) - h
  diff <- Y[(h + 1):nrow(Y), ] - Y[seq_len(n_h), ]
  crossprod(diff) / n_h
}

estimate_alpha_diag <- function(H1, H2, h_values) {
  Y <- simulate_Y(n, H1, H2, P)
  log_h <- log(h_values)
  log_M11 <- numeric(length(h_values))
  log_M22 <- numeric(length(h_values))

  for (i in seq_along(h_values)) {
    M <- calculate_M(Y, h_values[i])
    log_M11[i] <- log(M[1, 1])
    log_M22[i] <- log(M[2, 2])
  }

  alpha_11 <- coef(lm(log_M11 ~ log_h))[2]
  alpha_22 <- coef(lm(log_M22 ~ log_h))[2]

  df <- data.frame(log_h, log_M11, log_M22)

  p1 <<- ggplot(df, aes(x = log_h, y = log_M11)) +
    geom_point(color = "black", size = 3) +
    geom_smooth(method = "lm", color = "#4682B4", se = FALSE, linewidth =
       1.5) +
    labs(
      x = expression(log(h)),
      y = expression(log(M(h)[list(1,1)])),
      title = bquote(M(h)[list(1,1)] ~ ": " ~ alpha == .(sprintf("%.3f",
         alpha_11)))
    ) +
    theme_minimal(base_size = 14) +
    theme(
      panel.border = element_rect(color = "black", fill = NA, linewidth =
         1),
      panel.grid = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 16)
```

```
    ) +
    ylim(1, 9)

  p2 <<- ggplot(df, aes(x = log_h, y = log_M22)) +
    geom_point(color = "black", size = 3) +
    geom_smooth(method = "lm", color = "#4682B4", se = FALSE, linewidth =
       1.5) +
    labs(
      x = expression(log(h)),
      y = expression(log(M(h)[list(2,2)])),
      title = bquote(M(h)[list(2,2)] ~ ": " ~ alpha == .(sprintf("%.3f",
         alpha_22)))
    ) +
    theme_minimal(base_size = 14) +
    theme(
      panel.border = element_rect(color = "black", fill = NA, linewidth =
         1),
      panel.grid = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 16)
    ) +
    ylim(1, 9)

  c(alpha_11 = alpha_11, alpha_22 = alpha_22)
}

estimate_alpha_eigen <- function(H1, H2, h_values) {
  Y <- simulate_Y(n, H1, H2, P)
  log_h <- log(h_values)
  log_lambda1 <- numeric(length(h_values))
  log_lambda2 <- numeric(length(h_values))

  for (i in seq_along(h_values)) {
    M <- calculate_M(Y, h_values[i])
    lambda <- sort(eigen(M)$values, decreasing = TRUE)
    log_lambda1[i] <- log(lambda[1])
    log_lambda2[i] <- log(lambda[2])
  }

  alpha_1 <- coef(lm(log_lambda1 ~ log_h))[2]
  alpha_2 <- coef(lm(log_lambda2 ~ log_h))[2]

  df <- data.frame(log_h, log_lambda1, log_lambda2)

  p3 <<- ggplot(df, aes(x = log_h, y = log_lambda1)) +
    geom_point(color = "black", size = 3) +
    geom_smooth(method = "lm", color = "#4682B4", se = FALSE, linewidth =
       1.5) +
    labs(
      x = expression(log(h)),
      y = expression(log(lambda[2](M(h)))),
      title = bquote(lambda[2](M(h)) ~ ": " ~ alpha == .(sprintf("%.3f",
         alpha_1)))
    ) +
    theme_minimal(base_size = 14) +
```

```
    theme(
      panel.border = element_rect(color = "black", fill = NA, linewidth =
         1),
      panel.grid = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 16)
    ) +
    ylim(1, 9)

  p4 <<- ggplot(df, aes(x = log_h, y = log_lambda2)) +
    geom_point(color = "black", size = 3) +
    geom_smooth(method = "lm", color = "#4682B4", se = FALSE, linewidth =
       1.5) +
    labs(
      x = expression(log(h)),
      y = expression(log(lambda[1](M(h)))),
      title = bquote(lambda[1](M(h)) ~ ": " ~ alpha == .(sprintf("%.3f",
         alpha_2)))
    ) +
    theme_minimal(base_size = 14) +
    theme(
      panel.border = element_rect(color = "black", fill = NA, linewidth =
         1),
      panel.grid = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 16)
    ) +
    ylim(1, 9)

  c(alpha_lambda1 = alpha_1, alpha_lambda2 = alpha_2)
}

alpha_diag <- estimate_alpha_diag(H1, H2, h_values)
alpha_eigen <- estimate_alpha_eigen(H1, H2, h_values)

subtitle_diag <- textGrob("Diagonal Elements", x = 0.418, hjust = 0, gp =
   gpar(fontsize = 16))
subtitle_eigen <- textGrob("Eigenvalues", x = 0.455, hjust = 0, gp = gpar(
   fontsize = 16))

grid.arrange(
  subtitle_diag,
  arrangeGrob(p1, p2, ncol = 2),
  textGrob(""),
  subtitle_eigen,
  arrangeGrob(p4, p3, ncol = 2),
  ncol = 1,
  heights = c(0.1, 1, 0.03, 0.1, 1)
)
```

Listing 10: $M(h)$ Eigenvalue Asymptotics before Phase Transition

```
library(longmemo)
library(ggplot2)
library(gridExtra)
library(grid)
```

```
library(parallel)

n <- 2^16
p <- 50
h <- 2^12
H1 <- 0.6
H2 <- 0.6
centering1 <- 2 * H1 * log(h)
centering2 <- 2 * H2 * log(h)
P <- matrix(0, nrow = 2, ncol = p)
P[1, 1] <- 1
P[2, 2] <- 1
R <- 2000
num_bins <- 20
num_cores <- detectCores() - 1
RNGkind("L'Ecuyer-CMRG")

simulate_fBm <- function(n, H) {
  fGn <- simFGN0(n, H)
  fBm <- cumsum(fGn)
  return(fBm)
}

simulate_Y <- function(n, H1, H2, P, p) {
  X1 <- simulate_fBm(n, H1)
  X2 <- simulate_fBm(n, H2)
  X <- cbind(X1, X2)
  Z <- matrix(rnorm(n * p), nrow = n, ncol = p)
  Y <- X %*% P + Z
  return(Y)
}

calculate_M <- function(Y, h) {
  n_h <- nrow(Y) - h
  differences <- Y[(h + 1):nrow(Y), ] - Y[seq_len(n_h), ]
  M <- crossprod(differences) / n_h
  return(M)
}

simulate_eigenvalues <- function(r) {
  Y <- simulate_Y(n, H1, H2, P, p)
  M_h <- calculate_M(Y, h)
  lambda <- sort(eigen(M_h, symmetric = TRUE)$values, decreasing = TRUE)
  transformed <- sqrt(n / h) * c(log(lambda[1]) - centering2,
                                 log(lambda[2]) - centering1)
  return(transformed)
}

eigen_matrix <- matrix(unlist(
  mclapply(seq_len(R), simulate_eigenvalues, mc.cores = num_cores, mc.
     preschedule = TRUE)
), ncol = 2, byrow = TRUE)

eigen_df <- data.frame(
```

```
  lambda_p_minus1 = eigen_matrix[, 1],
  lambda_p = eigen_matrix[, 2]
)

mu1 <- mean(eigen_df$lambda_p_minus1)
sigma1 <- sd(eigen_df$lambda_p_minus1)
mu2 <- mean(eigen_df$lambda_p)
sigma2 <- sd(eigen_df$lambda_p)

ks.test(eigen_df$lambda_p_minus1, "pnorm", mean = mu1, sd = sigma1)
ks.test(eigen_df$lambda_p, "pnorm", mean = mu2, sd = sigma2)

breaks1 <- seq(min(eigen_df$lambda_p_minus1), max(eigen_df$lambda_p_minus1
   ), length.out = num_bins + 1)
breaks2 <- seq(min(eigen_df$lambda_p), max(eigen_df$lambda_p), length.out
   = num_bins + 1)

dens1 <- hist(eigen_df$lambda_p_minus1, breaks = breaks1, plot = FALSE,
   probability = TRUE)
dens2 <- hist(eigen_df$lambda_p, breaks = breaks2, plot = FALSE,
   probability = TRUE)

par(mfrow = c(1, 1), mar = c(5, 5.5, 2, 1), mgp = c(3.8, 0.7, 0))
hist(eigen_df$lambda_p_minus1,
     breaks = breaks1,
     probability = TRUE,
     main = "",
     xlab = expression(tilde(lambda)[p-1]),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens1$density)),
     axes = FALSE,
     cex.axis = 1.2,
     cex.lab = 1.3)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

hist(eigen_df$lambda_p,
     breaks = breaks2,
     probability = TRUE,
     main = "",
     xlab = expression(tilde(lambda)[p]),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens2$density)),
     axes = FALSE,
     cex.axis = 1.2,
     cex.lab = 1.3)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
```

```
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

eigen_df$norm_lambda_p_minus1 <- as.numeric(scale(eigen_df$lambda_p_minus1
   ))
eigen_df$norm_lambda_p <- as.numeric(scale(eigen_df$lambda_p))

par(mar = c(6, 8, 2, 1), mgp = c(4, 0.7, 0))
plot(eigen_df$norm_lambda_p_minus1, eigen_df$norm_lambda_p,
     main = "",
     xlab = expression(tilde(lambda)[p-1] / sigma[p-1]),
     ylab = "",
     col = "#4682B4",
     pch = 16,
     cex = 0.6,
     xlim = c(-4, 4),
     ylim = c(-4, 4),
     axes = FALSE)
axis(1, at = seq(-4, 4, by = 2), cex.axis = 1)
axis(2, at = seq(-4, 4, by = 2), cex.axis = 1)
mtext(expression(tilde(lambda)[p] / sigma[p]), side = 2, line = 2, cex =
   1.3)
box(bty = "o")

eigen_df$sum_squared <- eigen_df$norm_lambda_p^2 + eigen_df$norm_lambda_p_
   minus1^2
dens_sum <- hist(eigen_df$sum_squared, breaks = 30, plot = FALSE,
   probability = TRUE)

par(mar = c(6.5, 4, 2, 2), mgp = c(4.5, 0.7, 0))
hist(eigen_df$sum_squared,
     breaks = 30,
     probability = TRUE,
     main = "",
     xlab = expression((tilde(lambda)[p-1] / sigma[p-1])^2 + (tilde(lambda
        )[p] / sigma[p])^2),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens_sum$density)),
     cex.axis = 1.2,
     cex.lab = 1.0)
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

ks.test(eigen_df$sum_squared, "pchisq", df = 2)

simulate_GOE_diff <- function(i) {
  M <- matrix(rnorm(4), nrow = 2)
  GOE <- (M + t(M)) / sqrt(2)
  eigs <- sort(eigen(GOE, symmetric = TRUE)$values, decreasing = TRUE)
  return(eigs[1] - eigs[2])
}
```

```
GOE_diffs <- unlist(mclapply(seq_len(R), simulate_GOE_diff, mc.cores = num
   _cores, mc.preschedule = TRUE))
Mh_diffs <- eigen_df$lambda_p - eigen_df$lambda_p_minus1

par(mgp = c(2.2, 0.7, 0))
qqplot(GOE_diffs, Mh_diffs,
       xlab = expression("2x2 GOE:" ~ lambda[2] - lambda[1]),
       ylab = expression("M(h):" ~ lambda[p] - lambda[p-1]),
       pch = 16,
       col = "#4682B4",
       cex.lab = 1.3,
       cex.axis = 1.2,
       axes = FALSE)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")

ks.test(scale(Mh_diffs), scale(GOE_diffs))
```

Listing 11: $M(h)$ Eigenvalue Asymptotics after Phase Transition

```
library(longmemo)
library(parallel)

n <- 2^16
h <- 2^12
H1 <- 0.9
H2 <- 0.9
centering1 <- 2 * H1 * log(h)
centering2 <- 2 * H2 * log(h)
theta <- pi / 3
P <- matrix(c(cos(theta), -sin(theta), sin(theta), cos(theta)), nrow = 2)
R <- 2000
num_bins <- 20
num_cores <- detectCores() - 1
RNGkind("L'Ecuyer-CMRG")

simulate_fBm <- function(n, H) {
  fGn <- simFGN0(n, H)
  fBm <- cumsum(fGn)
  return(fBm)
}

simulate_Y <- function(n, H1, H2, P) {
  X1 <- simulate_fBm(n, H1)
  X2 <- simulate_fBm(n, H2)
  X <- rbind(X1, X2)
  Y <- t(P %*% X)
  return(Y)
}

calculate_M <- function(Y, h) {
  n_h <- nrow(Y) - h
  differences <- Y[(h + 1):nrow(Y), ] - Y[seq_len(n_h), ]
```

```
  M <- crossprod(differences) / n_h
  return(M)
}

simulate_eigenvalues <- function(r) {
  Y <- simulate_Y(n, H1, H2, P)
  M_h <- calculate_M(Y, h)
  lambda <- sort(eigen(M_h, symmetric = TRUE)$values, decreasing = TRUE)
  transformed <- sqrt(n / h) * c(log(lambda[2]) - centering1,
                                 log(lambda[1]) - centering2)
  return(transformed)
}

eigen_matrix <- matrix(unlist(
  mclapply(seq_len(R), simulate_eigenvalues, mc.cores = num_cores, mc.
     preschedule = TRUE)
), ncol = 2, byrow = TRUE)

eigen_df <- data.frame(
  lambda_p_minus1 = eigen_matrix[, 1],
  lambda_p = eigen_matrix[, 2]
)

mu1 <- mean(eigen_df$lambda_p_minus1)
sigma1 <- sd(eigen_df$lambda_p_minus1)
mu2 <- mean(eigen_df$lambda_p)
sigma2 <- sd(eigen_df$lambda_p)

ks.test(eigen_df$lambda_p_minus1, "pnorm", mean = mu1, sd = sigma1)
ks.test(eigen_df$lambda_p, "pnorm", mean = mu2, sd = sigma2)

breaks1 <- seq(min(eigen_df$lambda_p_minus1), max(eigen_df$lambda_p_minus1
   ), length.out = num_bins + 1)
breaks2 <- seq(min(eigen_df$lambda_p), max(eigen_df$lambda_p), length.out
   = num_bins + 1)

dens1 <- hist(eigen_df$lambda_p_minus1, breaks = breaks1, plot = FALSE,
   probability = TRUE)
dens2 <- hist(eigen_df$lambda_p, breaks = breaks2, plot = FALSE,
   probability = TRUE)

par(mfrow = c(1, 1), mar = c(5, 5.5, 2, 1), mgp = c(3.8, 0.7, 0))
hist(eigen_df$lambda_p_minus1,
     breaks = breaks1,
     probability = TRUE,
     main = "",
     xlab = expression(tilde(lambda)[p-1]),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens1$density)),
     axes = FALSE,
     cex.axis = 1.2,
     cex.lab = 1.3)
```

```
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

hist(eigen_df$lambda_p,
     breaks = breaks2,
     probability = TRUE,
     main = "",
     xlab = expression(tilde(lambda)[p]),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens2$density)),
     axes = FALSE,
     cex.axis = 1.2,
     cex.lab = 1.3)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

eigen_df$norm_lambda_p_minus1 <- as.vector(scale(eigen_df$lambda_p_minus1)
   )
eigen_df$norm_lambda_p <- as.vector(scale(eigen_df$lambda_p))

par(mar = c(6, 8, 2, 1), mgp = c(4, 0.7, 0))
plot(eigen_df$norm_lambda_p_minus1, eigen_df$norm_lambda_p,
     main = "",
     xlab = expression(tilde(lambda)[p-1] / sigma[p-1]),
     ylab = "",
     col = "#4682B4",
     pch = 16,
     cex = 0.6,
     xlim = c(-4, 4),
     ylim = c(-4, 4),
     axes = FALSE)
axis(1, at = seq(-4, 4, by = 2), cex.axis = 1)
axis(2, at = seq(-4, 4, by = 2), cex.axis = 1)
mtext(expression(tilde(lambda)[p] / sigma[p]), side = 2, line = 2, cex =
   1.3)
box(bty = "o")

eigen_df$sum_squared <- eigen_df$norm_lambda_p^2 + eigen_df$norm_lambda_p_
   minus1^2
dens_sum <- hist(eigen_df$sum_squared, breaks = 30, plot = FALSE,
   probability = TRUE)

par(mar = c(6.5, 4, 2, 2), mgp = c(4.5, 0.7, 0))
hist(eigen_df$sum_squared,
     breaks = 30,
     probability = TRUE,
     main = "",
     xlab = expression((tilde(lambda)[p-1] / sigma[p-1])^2 + (tilde(lambda
```

```
        )[p] / sigma[p])^2),
     ylab = "",
     col = "#A3C1DA",
     border = "black",
     ylim = c(0, 1.1 * max(dens_sum$density)),
     cex.axis = 1.2,
     cex.lab = 1.0)
mtext("Density", side = 2, line = 2.2, cex = 1.3)
box(bty = "o")

ks.test(eigen_df$sum_squared, "pchisq", df = 2)

simulate_GOE_diff <- function(i) {
  M <- matrix(rnorm(4), nrow = 2)
  GOE <- (M + t(M)) / sqrt(2)
  eigs <- sort(eigen(GOE, symmetric = TRUE)$values, decreasing = TRUE)
  return(eigs[1] - eigs[2])
}

simulate_M_diff <- function(i) {
  Y <- simulate_Y(n, H1, H2, P)
  M_h <- calculate_M(Y, h)
  eigs <- sort(eigen(M_h, symmetric = TRUE)$values, decreasing = TRUE)
  diff <- eigs[1] - eigs[2]
  return(diff)
}

GOE_diffs <- unlist(mclapply(seq_len(R), simulate_GOE_diff, mc.cores = num
   _cores, mc.preschedule = TRUE))
Mh_diffs <- unlist(mclapply(seq_len(R), simulate_M_diff, mc.cores = num_
   cores, mc.preschedule = TRUE))

par(mgp = c(2.2, 0.7, 0))
qqplot(GOE_diffs, Mh_diffs,
       xlab = expression("2x2 GOE:" ~ lambda[2] - lambda[1]),
       ylab = expression("M(h):" ~ lambda[p] - lambda[p-1]),
       pch = 16,
       col = "#4682B4",
       cex.lab = 1.3,
       cex.axis = 1.2,
       axes = FALSE)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")

ks.test(scale(Mh_diffs), scale(GOE_diffs))
```

Listing 12: Best-Fit β Estimation

```
library(MASS)
library(ggplot2)
library(longmemo)
library(parallel)
```

```
n <- 2^16
h <- 2^12
H1 <- 0.9
H2 <- 0.9
theta <- pi / 3
P <- matrix(c(cos(theta), -sin(theta), sin(theta), cos(theta)), nrow = 2)
cutoff <- 0.3
R <- 2000
num_repeats <- 10
num_cores <- detectCores() - 1
beta_values <- seq(0, 1, by = 0.1)
RNGkind("L'Ecuyer-CMRG")

simulate_fBm <- function(n, H) {
  fGn <- simFGN0(n, H)
  fBm <- cumsum(fGn)
  return(fBm)
}

simulate_Y <- function(n, H1, H2, P) {
  X1 <- simulate_fBm(n, H1)
  X2 <- simulate_fBm(n, H2)
  X <- rbind(X1, X2)
  Y <- t(P %*% X)
  return(Y)
}

calculate_M <- function(Y, h) {
  n_h <- nrow(Y) - h
  differences <- Y[(h + 1):nrow(Y), ] - Y[seq_len(n_h), ]
  M <- crossprod(differences) / n_h
  return(M)
}

simulate_M_diff <- function(i) {
  Y <- simulate_Y(n, H1, H2, P)
  M_h <- calculate_M(Y, h)
  eigs <- sort(eigen(M_h, symmetric = TRUE)$values, decreasing = TRUE)
  diff <- eigs[1] - eigs[2]
  return(diff)
}

compute_beta_diff <- function(beta) {
  d <- rnorm(2, mean = 0, sd = sqrt(2))
  sub_diag <- sqrt(rchisq(1, df = beta))
  T_beta <- matrix(c(d[1], sub_diag, sub_diag, d[2]), nrow = 2)
  eigs <- sort(eigen(T_beta, symmetric = TRUE)$values, decreasing = TRUE)
  diff <- eigs[1] - eigs[2]
  return(diff)
}

p_values_avg <- numeric(length(beta_values))

for (i in seq_along(beta_values)) {
```

```
  beta <- beta_values[i]
  p_values_repeats <- replicate(num_repeats, {
    eigdiff_M <- unlist(mclapply(seq_len(R), simulate_M_diff, mc.cores =
       num_cores))
    eigdiff_beta <- unlist(mclapply(seq_len(R), function(r) compute_beta_
       diff(beta), mc.cores = num_cores))

    subset <- floor(cutoff * R)
    eigdiff_M_subset <- sort(eigdiff_M)[seq_len(subset)]
    eigdiff_beta_subset <- sort(eigdiff_beta)[seq_len(subset)]

    eigdiff_M_norm <- scale(eigdiff_M_subset)
    eigdiff_beta_norm <- scale(eigdiff_beta_subset)

    KS_result <- ks.test(eigdiff_M_norm[, 1], eigdiff_beta_norm[, 1])
    return(KS_result$p.value)
  })

  p_values_avg[i] <- mean(p_values_repeats)
}

par(mar = c(5, 5.5, 2, 1), mgp = c(3.1, 0.7, 0))
plot(beta_values, p_values_avg,
     type = "o", pch = 16, col = "#4682B4",
     xlab = expression(beta * " (Hermite Ensemble)"),
     ylab = "Average KS Test p-value",
     cex.lab = 1.5, cex.axis = 1.2,
     axes = FALSE)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")

optimal_beta <- beta_values[which.max(p_values_avg)]
abline(v = optimal_beta, col = "#B22222", lty = 2)
```

Listing 13: QQ Plot for $\beta = 0.4$

```
library(MASS)
library(ggplot2)
library(longmemo)

n <- 2^16
h <- 2^12
H1 <- 0.9
H2 <- 0.9
theta <- pi / 3
P <- matrix(c(cos(theta), -sin(theta), sin(theta), cos(theta)), nrow = 2)
cutoff <- 0.3
R <- 2000
beta <- 0.4

simulate_fBm <- function(n, H) {
  fGn <- simFGN0(n, H)
  fBm <- cumsum(fGn)
```

```
  return(fBm)
}

simulate_Y <- function(n, H1, H2, P) {
  X1 <- simulate_fBm(n, H1)
  X2 <- simulate_fBm(n, H2)
  X <- rbind(X1, X2)
  Y <- t(P %*% X)
  return(Y)
}

calculate_M <- function(Y, h) {
  n_h <- nrow(Y) - h
  differences <- Y[(h + 1):nrow(Y), ] - Y[seq_len(n_h), ]
  M <- crossprod(differences) / n_h
  return(M)
}

simulate_M_diff <- function(i) {
  Y <- simulate_Y(n, H1, H2, P)
  M_h <- calculate_M(Y, h)
  eigs <- sort(eigen(M_h, symmetric = TRUE)$values, decreasing = TRUE)
  diff <- eigs[1] - eigs[2]
  return(diff)
}

compute_beta_diff <- function(beta) {
  d <- rnorm(2, mean = 0, sd = sqrt(2))
  sub_diag <- sqrt(rchisq(1, df = beta))
  T_beta <- matrix(c(d[1], sub_diag, sub_diag, d[2]), nrow = 2)
  eigs <- sort(eigen(T_beta, symmetric = TRUE)$values, decreasing = TRUE)
  diff <- eigs[1] - eigs[2]
  return(diff)
}

eigdiff_M <- numeric(R)
eigdiff_beta <- numeric(R)

for (r in seq_len(R)) {
  eigdiff_M[r] <- simulate_M_diff(r)
  eigdiff_beta[r] <- compute_beta_diff(beta)
}

subset <- floor(cutoff * R)
eigdiff_M_subset <- sort(eigdiff_M)[seq_len(subset)]
eigdiff_beta_subset <- sort(eigdiff_beta)[seq_len(subset)]

eigdiff_M_norm <- scale(eigdiff_M_subset)
eigdiff_beta_norm <- scale(eigdiff_beta_subset)

par(mar = c(5, 5.5, 2, 1), mgp = c(3.1, 0.7, 0))
qqplot(
  eigdiff_beta_subset, eigdiff_M_subset,
  ylab = expression(M(h) ~ ":" ~ lambda[p] - lambda[p-1]),
```

```
  xlab = expression(paste(beta, "-Hermite matrix (", beta, " = 0.4): ",
     lambda[2] - lambda[1])),
  pch = 16,
  col = "#4682B4",
  cex.lab = 1.5,
  cex.axis = 1.2,
  axes = FALSE
)
axis(1, cex.axis = 1.2)
axis(2, cex.axis = 1.2)
box(bty = "o")

KS_result <- ks.test(eigdiff_M_norm[, 1], eigdiff_beta_norm[, 1])
```

References

Abry, P., Boniece, B. C., Didier, G., and Wendt, H. (2024). On high-dimensional wavelet eigenanalysis. *Annals of Applied Probability*, 34(2):5287–5350.

Abry, P., Didier, G., Orejola, O., and Wendt, H. (2025). On the empirical spectral distribution of large wavelet random matrices based on mixed-Gaussian fractional measurements in moderately high dimensions. *Electronic Journal of Probability*, 30:1–48.

Bai, Z. and Silverstein, J. W. (2010). *Spectral Analysis of Large Dimensional Random Matrices*. Springer Series in Statistics. Springer, 2nd edition.

Casella, G. and Berger, R. L. (2002). *Statistical Inference*. Duxbury, 2nd edition.

Folland, G. B. (1999). *Real Analysis: Modern Techniques and Their Applications*. John Wiley & Sons, 2nd edition.

Michalowicz, J., Nichols, J., Bucholtz, F., and Olson, C. (2009). An Isserlis' theorem for mixed Gaussian variables: Application to the auto-bispectral density. *Journal of Statistical Physics*, 136:89–102.

van der Vaart, A. W. (2000). *Asymptotic Statistics*. Cambridge University Press, 1st edition.

Zhang, K., Crizer, K. P. R., Schoenfisch, M. H., Hill, D. B., and Didier, G. (2018). [fluid heterogeneity detection based on the asymptotic distribution of the time-averaged mean squared displacement in single particle tracking experiments]. *Journal of Physics A: Mathematical and Theoretical*, 51.