The Covariance matrix

Sui Jiet Tay CS.JIET@GMAIL.COM

Keywords: sample mean; centering; shift invariance of sample variance; scaling behavior of sample variance; orthogonal projection; scalar projection / projection coefficient; projected (1D) variance; outer product; covariance matrix (sample); quadratic form; symmetry; positive semidefinite (PSD); Cauchy–Schwarz inequality; colinearity condition for equality in Cauchy–Schwarz; Lagrange multiplier / KKT conditions (equality-constrained); sensitivity to outliers; Asymptotic tight bound; Big-Theta notation; time complexity; space complexity; Gram matrix; row-major order; spatial locality; strided access; cache lines; cache misses; memory bandwidth

Contents

1. Motivation

This writing highlights several properties of the covariance matrix that help us understand its strengths and limitations in practical modeling.

We will see that this is a useful tool because it sits at the nexus of many foundational ideas in statistics and machine learning — the correlation matrix, PCA, the multivariate Gaussian, Gaussian mixture models, and Gaussian processes—each offering a richer modeling language in a different way, and all of which remain prevalent even in the era of deep learning. For example, 3D Gaussian Splatting makes explicit use of the same covariance-as-ellipse geometry (via symmetric positive semidefinite matrices and quadratic forms) that we develop here. Moreover, many of the arguments in this section serve as a "proof to myself" of the covariance matrix's properties, so that I can apply it in the correct setting and let it inform modeling decisions.

Here are some questions this framework partially informs: What does each entry of the covariance matrix actually capture? How can this guide simple model choices that fit within current computational resources while avoiding overfitting? Which features tend to be most informative under a second-moment summary, and is that consistent with what a deep neural network's input layer 'sees'? If not, why not? How does this perspective inform data preprocessing and representation? Finally, from an algorithmic standpoint, how expensive is it—in time and in memory—to compute and to use? When is covariance still a valid and useful part of the modern data scientist's toolkit, and when should we reach for something else?

2. Mean

(English):

(Statistics):

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\widetilde{x} : \Omega \to \mathbb{R}$ be a real-valued random variable.

We define the mean of $\widetilde{x}$ as

$$\mathbb{E}[\widetilde{x}]$$

In the empirical setup,

Let $D := (x_1, \ldots, x_n)$ be an n-tuple of real values — an observed (univariate) dataset. Let $m : \mathbb{R}^n \to \mathbb{R}$ be the empirical mean function.

$$m(D) = \frac{1}{n} \sum_{i=1}^{n} D[i] \tag{1}$$

The idea is that from sufficient finite data

$$m(D) \approx \mathbb{E}[\widetilde{x}]$$

3. Variance

(English): The fact that two or more things are different, or the amount or number by which they are different.

(Statistics): The expected value of the squared deviation from the mean.

let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\widetilde{x} : \Omega \to \mathbb{R}$ be a real-valued random variable.

We define the variance of $\widetilde{x}$ by

$$\mathrm{Var}[\widetilde{x}] = \mathbb{E}\big[\big(\widetilde{x} - \mathbb{E}[\widetilde{x}]\big)^2\big].$$

In the empirical setup,

Let D denote the observed dataset, let $v : \mathbb{R}^n \to \mathbb{R}$ be the empirical variance function, and $m : \mathbb{R}^n \to \mathbb{R}$ be the empirical mean function.

$$v(D) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - m(D))^2 \tag{2}$$

The idea is that from sufficient finite data

$$v(D) \approx \mathrm{Var}[\widetilde{x}]$$

Why squaring make sense? When we say that a quantity "varies," we can only mean that it varies *relative to* a chosen reference point. In the definition of variance, the reference point is the mean. That is, we measure how far each data point deviates from the mean.

Next, we do not want positive and negative deviations to cancel, since both represent variability. Thus we apply a *nonnegative transform* to each deviation. One option is the absolute value $|\cdot|$, but the conventional choice is squaring. Squaring has the additional effect of penalizing larger deviations more strongly: as the deviation grows in magnitude, its contribution grows superlinearly (quadratically), so points farther from the mean contribute disproportionately more to the overall variability.

To show this, fix a mean $\mu \in \mathbb{R}$ and a data value $x \in \mathbb{R}$, and define the deviation

$$a := x - \mu.$$

Now perturb the data value by an offset $b \in \mathbb{R}$, i.e., replace x by $x + b$. The new deviation is

$$(x + b) - \mu = a + b.$$

Hence,

$$(a + b) - a = b,$$

so studying $(a + b) - a$ is exactly studying how the *gap* (variable minus mean) changes when we perturb the variable by b.

With this notation fixed, we compare how the *absolute* deviation and the *squared* deviation change under the same perturbation:

$$|a + b| - |a| \qquad \text{versus} \qquad (a + b)^2 - a^2.$$

Squared deviation. A direct expansion gives

$$(a + b)^2 - a^2 = 2ab + b^2.$$

In particular, if $a \geq 0$ and $b \geq 0$, then

$$(a + b)^2 - a^2 = 2ab + b^2 \geq b^2,$$

and for any fixed $b > 0$ this increment grows linearly in a through the term $2ab$. Thus, as $|a|$ becomes large, the additional penalty induced by the same offset b becomes larger.

Absolute deviation. The absolute-value increment is only piecewise linear. By a case analysis,

$$|a+b| - |a| = \begin{cases} (a+b) - a = b, & a \geq 0,\ a+b \geq 0, \\ (a+b) - (-a) = 2a + b, & a < 0,\ a+b \geq 0, \\ -(a+b) - a = -2a - b, & a \geq 0,\ a+b < 0, \\ -(a+b) - (-a) = -b, & a < 0,\ a+b < 0. \end{cases}$$

Interpretation. The key point is that under squaring, the increment

$$(a+b)^2 - a^2 = 2ab + b^2$$

contains the interaction term $2ab$, which makes the increase depend on the current deviation level a. Thus, when a point is already far from the mean (large $|a|$), an additional offset produces a larger increase in the squared penalty than it would near the mean.

Finally, variance averages these squared deviations to obtain a single second-moment summary:

$$v(D) = \frac{1}{n-1} \sum_{i=1}^{n} \bigl(x^{(i)} - \mu\bigr)^2 \qquad \text{(with } \mu \text{ taken as the sample mean in practice).}$$

Moreoever, there are several other reasons why squaring is preferred, squaring is smooth and differentiable, which makes it convenient for analysis and optimization. Third, squared deviations align with the rich geometry of inner products and ℓ_2 norms: many powerful inequalities and algebraic identities (e.g., Cauchy–Schwarz, Pythagorean relations, and norm bounds) become available and often yield clean, closed-form manipulations. In short, squaring gives both a meaningful notion of spread and a particularly tractable mathematical structure.

Finally, the factor $\frac{1}{n-1}$ (rather than $\frac{1}{n}$) is used so that the sample variance is an *unbiased estimator* of the population variance under the usual i.i.d. sampling model.

Knowing the one-dimensional definitions of the mean and variance, the question now becomes: how do we generalize these constructions to multivariate data, where each observation has multiple features? This can be thought of as a random vector $\widetilde{\mathbf{x}} : \Omega \to \mathbb{R}^d$, where $\widetilde{\mathbf{x}} := (\widetilde{x_1}, \widetilde{x_2}, \cdots, \widetilde{x_n})$, where each realization is a vector of the features of a single datum.

4. Mean vector

There are several ways one might attempt to encode the "central tendency" of a dataset of vectors. A natural approach is to study each coordinate *with respect to a fixed basis* independently. Concretely, if we view each component/ entry of the vector as univariate data and average it component-wise, we obtain another vector called the *sample mean vector*. An intuitive way to represent a collection of multivariate datum is by stacking each vector row-wise to form a matrix.

Sampling model. Unless stated otherwise, we assume there exists an $\mathbb{R}^d$-valued random feature vector $\widetilde{\mathbf{x}} : \Omega \to \mathbb{R}^d$ such that the observed samples

$$\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)} \overset{\text{i.i.d.}}{\sim} \widetilde{\mathbf{x}}$$

are independent and identically distributed draws from the distribution (law) of $\widetilde{\mathbf{x}}$. Equivalently, for each i, the vector $\mathbf{x}^{(i)}$ is a realization of $\widetilde{\mathbf{x}}$.

Dataset as a matrix. Given an observed multivariate dataset

$$D := (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}), \qquad \mathbf{x}^{(i)} \in \mathbb{R}^d,$$

we may equivalently represent it by a matrix $X_{\text{row}} \in \mathbb{R}^{n \times d}$, where the subscript row indicates the *rows-as-samples* convention (each row is one datum, each column is one feature). Thus the i-th row

$$\mathbf{x}^{(i)} := X_{\text{row}}[i] \in \mathbb{R}^d \qquad (i \in \{1, \dots, n\})$$

is the realized feature vector of datum i.

Note. In linear algebra, it is common to view a matrix as a collection of its *column* vectors. Accordingly, one must be explicit about whether samples are stored as rows or as columns, since silently mixing these conventions can lead to incorrect calculations. In much of machine learning and data science, it is common to represent a dataset by a matrix whose *rows* are samples and whose *columns* are features, largely for notational convenience. To avoid ambiguity, we reserve X_{row} for the row-major (rows-as-samples) convention, and we use X for the column-major (columns-as-vectors) convention commonly used in linear algebra. When comparing sources, it is therefore prudent to check whether a given author uses X or $X^\top$ to represent the same dataset.

Row-wise stacking. With the rows-as-samples convention, the dataset matrix is

$$X_{\text{row}} = \begin{bmatrix} \mathbf{x}^{(1)\top} \\ \mathbf{x}^{(2)\top} \\ \vdots \\ \mathbf{x}^{(n)\top} \end{bmatrix} \in \mathbb{R}^{n \times d}.$$

So let $X_{\text{row}} \in \mathbb{R}^{n \times d}$ denote a data matrix whose i-th row $X_{\text{row}}[i] \in \mathbb{R}^d$ is the i-th sample, and let $X_{\text{row}}[i][j]$ denote its j-th coordinate. Then the sample mean vector is

$$\mathbf{m}(X_{\text{row}}) = \Big(\frac{1}{n}\sum_{i=1}^{n} X_{\text{row}}[i][1], \cdots, \frac{1}{n}\sum_{i=1}^{n} X_{\text{row}}[i][d]\Big) \in \mathbb{R}^d.$$

An unorthodox notion of a "mean" for vector-valued data is to aggregate all coordinates into a single real number. However, such a scalar aggregation generally discards the coordinate-wise structure of the data and collapses the problem back to a univariate summary. For this reason, and by standard convention, it is more natural to encode the mean of vector-valued data as a vector (the mean vector), thereby preserving a distinct summary for each coordinate.

An even cooler way to think and define the mean vector is through *orthogonal projections*. As we will soon see, this provides an intuitive and compact viewpoint in higher dimensions, since it does not require a separate definition of the mean in vector form. Instead, we can reuse the one-dimensional sample mean: by projecting onto a line, we reduce the problem to the familiar univariate setting, apply the usual sample mean, and then recover the multivariate statement from this single one-dimensional formula.

4.1. Orthogonal projection

Recall that the orthogonal projection of a vector onto a subspace can be expressed in coordinates; in the one-dimensional case, these coordinates coincide with the familiar scalar projection from elementary geometry. Let $U \subseteq \mathbb{R}^d$ be a k-dimensional subspace, and let $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ be a basis of U. Then for any $\mathbf{x} \in \mathbb{R}^d$, the orthogonal projection of $\mathbf{x}$ onto U is a vector in U and hence can be written as

$$P_U(\mathbf{x}) = \sum_{i=1}^{k} \alpha_i \, \mathbf{u}_i$$

for some coefficients $\alpha_1, \dots, \alpha_k \in \mathbb{R}$. This equation can be called the "*coordinate isomorphism map*". Equivalently, if we define the coordinate vector $\alpha := (\alpha_1, \dots, \alpha_k)^\top \in \mathbb{R}^k$, then α is the coordinate representation of $P_U(\mathbf{x})$ with respect to the basis $\{\mathbf{u}_i\}_{i=1}^k$. In general, we must solve a linear system to determine these coefficients, which can be pretty tedious by hand since it typically means running Gaussian elimination to extract them.

However, if $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ is an *orthonormal* basis of U, then the coefficients are obtained immediately without needing to solve a linear system:

$$P_U(\mathbf{x}) = \sum_{i=1}^{k} \underbrace{\langle \mathbf{x}, \mathbf{u}_i \rangle}_{\alpha_i} \, \mathbf{u}_i \tag{3}$$

This expression is read as: "the projection of $\mathbf{x}$ onto the subspace U, expressed in the orthonormal basis $\{\mathbf{u}_i\}_{i=1}^k$".

Recall that we can always obtain such a basis with ease via the Gram–Schmidt process algorithm: given any linearly independent set of vectors that spans the subspace U as input, Gram–Schmidt produces an orthonormal basis that spans the same subspace.

Hence, we are ready. Since the usual arithmetic mean is first introduced in the univariate setting, it acts on scalars. To connect this to vector-valued data, we select an arbitrary nonzero direction u and consider the one-dimensional subspace $U := \operatorname{span}\{\mathbf{u}\}$, which is geometrically a line. We then project each data vector $\mathbf{x}$ onto this line. When $\mathbf{u}$ is unit-length, the projection coefficient is the scalar

$$P_{\operatorname{span}\{\mathbf{u}\}}(\mathbf{x}) \;=\; \langle \mathbf{x}, \mathbf{u} \rangle \, \mathbf{u}, \qquad \text{and the corresponding scalar projection coefficient is} \qquad \langle \mathbf{x}, \mathbf{u} \rangle \in \mathbb{R},$$

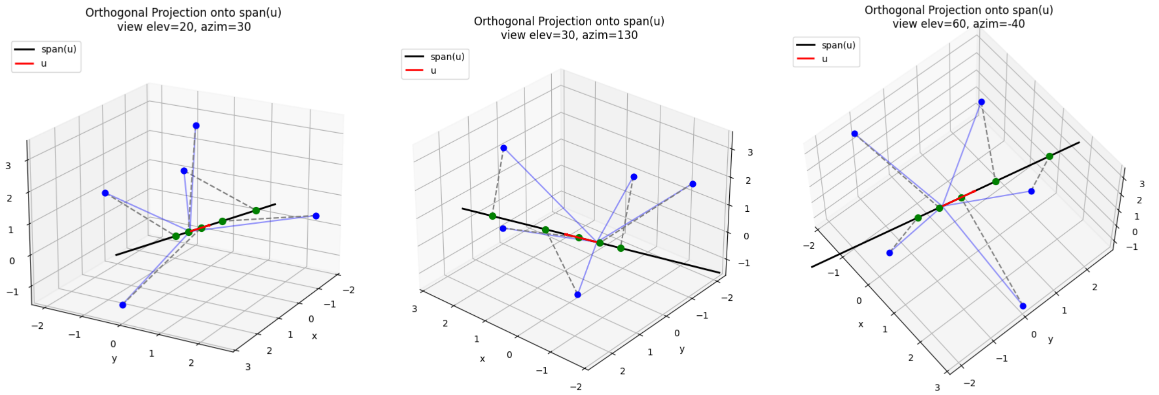
which is exactly the (one-dimensional) coordinate of $\mathbf{x}$ along the axis U. We may therefore take the usual one-dimensional sample mean of these projection coefficients.

This is a crucial difference from the earlier "unorthodox" scalar aggregation across coordinates: projection does not indiscriminately merge all coordinates into a single number. Instead, doing an orthogonal projection preserves meaningful directional information — different directions u induce different projections — so the summary still reflects how the data behave along that chosen axis. In this sense, projection provides a principled scalar viewpoint that remains faithful to the multivariate structure.

If we project all n data points, we get a set of 1D coordinates

$$D = \Big((\langle \mathbf{x}^{(1)}, \mathbf{u} \rangle), \cdots, (\langle \mathbf{x}^{(n)}, \mathbf{u} \rangle) \Big)$$

which we can feed as input into the empirical mean function defined in (1).



(a) Enter caption (b) Enter caption (c) Enter caption

Figure 1: Consider a 3D subspace projecting to a 1D subspace

By some algebra we see

$$\begin{aligned} m(D) &= \frac{1}{n}\sum_{i=1}^{n} D[i] \\ &= \frac{1}{n}\sum_{i=1}^{n} \langle \mathbf{x}^{(i)}, \mathbf{u} \rangle \\ &= \frac{1}{n}\left\langle \sum_{i=1}^{n} \mathbf{x}^{(i)}, \mathbf{u} \right\rangle && (\text{ bilinearity of inner product }) \\ &= \left\langle \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)}}_{\boldsymbol{\mu}}, \mathbf{u} \right\rangle && (\text{ bilinearity of inner product }) \\ &\text{Define the } \textit{sample mean vector} \text{ by } \boldsymbol{\mu} \\ &\text{Then we obtain} \\ &= \langle \boldsymbol{\mu}, \mathbf{u} \rangle \in \mathbb{R} \end{aligned}$$

We see that the mean of the projection is the projection of the mean.

$$\frac{1}{n}\sum_{i=1}^{n} \langle \mathbf{x}^{(i)}, \mathbf{u} \rangle \;=\; \left\langle \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)}}_{\boldsymbol{\mu}}, \mathbf{u} \right\rangle.$$

Thus, the familiar component-wise averaging in $\mathbb{R}^d$ emerges naturally, with $\boldsymbol{\mu}$ as the mean vector. In particular, the sample mean vector is

$$\boldsymbol{\mu} := \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)} \in \mathbb{R}^d.$$

This projection viewpoint — reducing vector-valued data to one-dimensional scalar coordinates along a chosen span — will later provide a principled route to defining and computing the covariance matrix.

5. Covariance matrix

(English) “co-” (prefix): together; jointly, “variance”: The fact that two or more things are different, or the amount or number by which they are different.

Hence combining them, it is how things are jointly different.

(Statistics)

We are interested in scalar quantities that capture how variables vary and co-vary. For a single variable, this is variance; for two variables, this is covariance, which measures how they move together.

Since there are many reasonable design choices for a measure of co-variation (each with its own strengths and assumptions), we begin with the simplest and most standard construction: the classical covariance and covariance matrix as introduced in elementary statistics. We will later make explicit the assumptions implicit in this definition and discuss the settings in which it becomes limited.

Here are some common simplifications implicit in the classical covariance which we will show below:

1. **Centered about the mean and scale dependence:** it measures co-variation around the mean specifically; this is appropriate for many purposes but is still a design choice (other centers are possible, e.g., medians in robust statistics). Covariance is not scale-invariant. Rescaling a feature (e.g., dollars vs. cents) rescales covariances; comparability across coordinates typically requires standardization (leading to correlation).

2. **Pairwise summary:** covariance measures relationships only in a pairwise manner (one coordinate pair at a time).

3. **Linear co-variation:** covariance primarily captures linear dependence; nonlinear relationships may not be reflected.

4. **Sensitivity to outliers:** because it is based on squared deviations, covariance can be dominated by extreme values; it is not a robust measure of dependence.

5. **Second-moment summary:** covariance depends only on second moments (means and products). Distinct distributions can share the same covariance while having very different higher-order structure (skewness, kurtosis).

The covariance is defined as

$$\mathrm{Cov}(\widetilde{x}, \widetilde{y}) := \mathbb{E}\Big[(\widetilde{x} - \mathbb{E}[\widetilde{x}])(\widetilde{y} - \mathbb{E}[\widetilde{y}])\Big].$$

In the empirical setup, we observe multivariate data. Let

$$D := \big(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\big), \qquad \mathbf{x}^{(i)} \in \mathbb{R}^d.$$

Write the j-th coordinate of $\mathbf{x}^{(i)}$ as $x_j^{(i)} \in \mathbb{R}$, and define the sample mean of coordinate j by

$$\boldsymbol{\mu}(j) := \frac{1}{n}\sum_{i=1}^{n} x_j^{(i)}$$

Then the empirical (sample) covariance between coordinates j and k is

$$\widehat{\mathrm{Cov}}(j,k) := \frac{1}{n-1}\sum_{i=1}^{n}\big(x_j^{(i)} - \boldsymbol{\mu}_j\big)\big(x_k^{(i)} - \boldsymbol{\mu}_k\big) \tag{4}$$

5.1. Data centering

The simplest form of centering is to translate the data by subtracting its mean. We first show how centering is computed for a univariate dataset, and then extend the same construction to the multivariate setting.

5.1.1. UNIVARIATE DATASET CENTERING

Before introducing the full covariance matrix, it is helpful to isolate the underlying invariances in the *scalar* setting.

In a **univariate** dataset, each sample contributes a single scalar $x^{(i)} \in \mathbb{R}$, so the relevant second-moment summary is the *sample variance*. By contrast, *covariance is inherently a pairwise notion*: it is defined between *two* scalar-valued features (or random variables). Thus, discussing covariance already presumes a multivariate dataset. Concretely, we work with data in which each sample carries at least two feature coordinates; we extract the paired scalars $(x^{(i)}, y^{(i)}) \in \mathbb{R}^2$ (two 1-dimensional feature coordinates from each sample) and study the *sample covariance* $\widehat{\mathrm{Cov}}(x,y)$.

Once the algebra is clear for *variance* (one feature), we then move to *covariance* (two features), where the covariance matrix for d features is obtained by assembling all pairwise covariances into a single $d \times d$ matrix.

We now prove that *centering* (subtracting the sample mean) does *not* change the sample variance.

Lemma 1 (Shift invariance and scaling behavior of univariate (1-dimensional) sample variance) *Let $D := (x^{(1)}, \dots, x^{(n)}) \in \mathbb{R}^n$ be a univariate dataset, where $x^{(i)} \in \mathbb{R}$. Define the sample mean*

$$\mu := \frac{1}{n}\sum_{i=1}^{n} x^{(i)},$$

and the sample variance

$$v(D) := \frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)} - \mu\big)^2.$$

Fix $\alpha, \beta \in \mathbb{R}$ and define the transformed data $x^{(i)\prime} := \alpha x^{(i)} + \beta$, with dataset $D' := (x^{(1)\prime}, \dots, x^{(n)\prime})$. Then

$$v(D') = \alpha^2\, v(D).$$

In particular, $v(D') = v(D)$ for any pure shift ($\alpha = 1$), whereas scaling by α multiplies the variance by α^2.

Proof First compute the transformed mean:

$$\begin{aligned}\mu' &:= \frac{1}{n}\sum_{i=1}^{n} x^{(i)\prime} \\ &= \frac{1}{n}\sum_{i=1}^{n}(\alpha x^{(i)} + \beta) \\ &= \alpha\left(\frac{1}{n}\sum_{i=1}^{n} x^{(i)}\right) + \beta \\ &= \alpha\,\mu + \beta.\end{aligned}$$

Then

$$\begin{aligned}v(D') &= \frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)\prime} - \mu'\big)^2 \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\big((\alpha x^{(i)} + \beta) - (\alpha\mu + \beta)\big)^2 \\ &= \frac{1}{n-1}\sum_{i=1}^{n}\big(\alpha(x^{(i)} - \mu)\big)^2 \\ &= \alpha^2\left(\frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)} - \mu\big)^2\right) \\ &= \alpha^2\, v(D),\end{aligned}$$

as claimed. ■

Since shifting does not affect the value of the variance, why do we do it? For several reasons: it improves numerical stability computationally, simplifies many algebraic expressions, and makes the interpretation “variation about the mean” explicit. For this reason, it is convenient to *center* the data, i.e., shift all data points so that the mean becomes zero.

This operation corresponds to subtracting the sample mean from every data vector:

$$\mathbf{x}^{(i)} \;\longmapsto\; \mathbf{x}^{(i)} - \boldsymbol{\mu}, \qquad i \in \{1, \dots, n\},$$

where $\boldsymbol{\mu} := \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}^{(i)}$ is the sample mean vector.

5.1.2. MULTIVARIATE DATASET CENTERING

Equivalently, in matrix form (with rows-as-samples), we define centering as

$$X_{\text{row}} \;\longmapsto\; X_{\text{row}} - \mathbf{1}\,\boldsymbol{\mu}^{\top},$$

where $\mathbf{1} \in \mathbb{R}^n$ denotes the all-ones column vector.

Whether we work in one dimension or component-wise in the multivariate setting, centering has a simple geometric meaning. On the real line, centering changes only absolute position: it translates every point by the same amount, while preserving all relative differences and the ordering of the points. This reflects a key property of variance: *variance is invariant under shifts* (translation-invariant).

Lemma 2 (Shift invariance and scaling behavior of sample covariance) *Let $D := \big\{ (\, x^{(i)}, y^{(i)} \,) \big\}_{i=1}^{n} \subset \mathbb{R}^2$ be a bivariate dataset, where $x^{(i)}, y^{(i)} \in \mathbb{R}$. Define the sample means*

$$\mu(x) := \frac{1}{n}\sum_{i=1}^{n} x^{(i)}, \qquad \mu(y) := \frac{1}{n}\sum_{i=1}^{n} y^{(i)},$$

and the sample covariance

$$\widehat{\mathrm{Cov}}_D(x, y) := \frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)} - \mu(x)\big)\big(y^{(i)} - \mu(y)\big).$$

Fix $\alpha, \beta, \gamma, \delta \in \mathbb{R}$ and define the transformed data

$$x^{(i)\prime} := \alpha\, x^{(i)} + \beta, \qquad y^{(i)\prime} := \gamma\, y^{(i)} + \delta,$$

with transformed dataset $D' := \big\{ (\, x^{(i)\prime}, y^{(i)\prime} \,) \big\}_{i=1}^{n}$. Then

$$\widehat{\mathrm{Cov}}_{D'}(x', y') = \alpha\gamma\, \widehat{\mathrm{Cov}}_D(x, y).$$

In particular, $\widehat{\mathrm{Cov}}$ is shift-invariant in each coordinate (adding β to x and/or δ to y does not change it), whereas scaling x by α and y by γ multiplies the covariance by $\alpha\gamma$.

Proof First compute the transformed means:

$$\begin{aligned}
\mu(x)' &:= \frac{1}{n}\sum_{i=1}^{n} x^{(i)\prime} \\
&= \frac{1}{n}\sum_{i=1}^{n}\big(\alpha x^{(i)} + \beta\big) \\
&= \alpha\left(\frac{1}{n}\sum_{i=1}^{n} x^{(i)}\right) + \beta \\
&= \alpha\, \mu(x) + \beta,
\end{aligned}$$

and similarly,

$$\begin{aligned}
\mu(y)' &:= \frac{1}{n}\sum_{i=1}^{n} y^{(i)\prime} \\
&= \frac{1}{n}\sum_{i=1}^{n}\big(\gamma y^{(i)} + \delta\big) \\
&= \gamma\left(\frac{1}{n}\sum_{i=1}^{n} y^{(i)}\right) + \delta \\
&= \gamma\, \mu(y) + \delta.
\end{aligned}$$

Now expand the sample covariance of the transformed dataset:

$$\begin{aligned}
\widehat{\mathrm{Cov}}_{D'}(x', y') &= \frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)\prime} - \mu(x)'\big)\big(y^{(i)\prime} - \mu(y)'\big) \\
&= \frac{1}{n-1}\sum_{i=1}^{n}\big((\alpha x^{(i)} + \beta) - (\alpha\mu(x) + \beta)\big)\big((\gamma y^{(i)} + \delta) - (\gamma\mu(y) + \delta)\big) \\
&= \frac{1}{n-1}\sum_{i=1}^{n}\big(\alpha(x^{(i)} - \mu(x))\big)\big(\gamma(y^{(i)} - \mu(y))\big) \\
&= \alpha\gamma\left(\frac{1}{n-1}\sum_{i=1}^{n}\big(x^{(i)} - \mu(x)\big)\big(y^{(i)} - \mu(y)\big)\right) \\
&= \alpha\gamma\,\widehat{\mathrm{Cov}}_D(x, y),
\end{aligned}$$

as claimed. ■

We typically refer to this as "zero-centering" to achieve a mean of zero. To see why this holds true consider the following lemma.

Lemma 3 *Let* $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)} \in \mathbb{R}^d$ *be an observed dataset, and define the sample mean vector by*

$$\boldsymbol{\mu}(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}^{(i)} \in \mathbb{R}^d,$$

and the centered observations by

$$\mathbf{c}^{(i)} := \mathbf{x}^{(i)} - \boldsymbol{\mu}(\mathbf{x}) \qquad \textit{for all } i \in \{1, \ldots, n\}.$$

Fix any unit vector $\mathbf{u} \in \mathbb{R}^d$ *and define the projected scalars*

$$\mathbf{s}^{(i)} := \langle \mathbf{c}^{(i)}, \mathbf{u} \rangle \in \mathbb{R}.$$

Then the sample mean of the centered projected scalars is zero:

$$\boldsymbol{\mu}(\mathbf{s}) := \frac{1}{n}\sum_{i=1}^{n} s^{(i)} = 0.$$

Proof First we center the data:

$$\begin{aligned}
\boldsymbol{\mu}(\mathbf{x}) &:= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}^{(i)}, \\
\mathbf{c}^{(i)} &:= \mathbf{x}^{(i)} - \boldsymbol{\mu}(\mathbf{x}), \qquad \forall i, \\
C &:= \left(\mathbf{c}^{(i)}\right)_{i=1}^{n}.
\end{aligned}$$

Again because we want to reuse the univariate formulas, we project onto the span of an arbitrary unit vector $\mathbf{u}$. Since $U := \mathrm{span}\{\mathbf{u}\}$ is one-dimensional, the orthogonal projection is

$$P_U(\mathbf{c}^{(i)}) = \langle \mathbf{c}^{(i)}, \mathbf{u} \rangle \mathbf{u}.$$

Extract the scalar coordinate

$$s^{(i)} := \langle \mathbf{c}^{(i)}, \mathbf{u} \rangle.$$

Then we take the mean of these centered coordinates:

$$\begin{aligned}
\boldsymbol{\mu}(\mathbf{s}) &:= \frac{1}{n}\sum_{i=1}^{n} s^{(i)} \\
&= \frac{1}{n}\sum_{i=1}^{n} \langle \mathbf{c}^{(i)}, \mathbf{u} \rangle \\
&= \left\langle \frac{1}{n}\sum_{i=1}^{n} \mathbf{c}^{(i)}, \mathbf{u} \right\rangle \qquad (\text{linearity in the first argument}) \\
&= \left\langle \frac{1}{n}\sum_{i=1}^{n} \big(\mathbf{x}^{(i)} - \boldsymbol{\mu}(\mathbf{x})\big), \mathbf{u} \right\rangle \\
&= \left\langle \left(\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)} - \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\mu}(\mathbf{x}) \right), \mathbf{u} \right\rangle \qquad (\text{linearity of summation}) \\
&= \left\langle \left(\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)} - \frac{1}{n} \cdot n \cdot \boldsymbol{\mu}(\mathbf{x}) \right), \mathbf{u} \right\rangle \\
&= \left\langle \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)}}_{\boldsymbol{\mu}(\mathbf{x})} - \boldsymbol{\mu}(\mathbf{x}), \mathbf{u} \right\rangle \\
&= \langle \boldsymbol{\mu}(\mathbf{x}) - \boldsymbol{\mu}(\mathbf{x}), \mathbf{u} \rangle \\
&= \langle \mathbf{0}_d, \mathbf{u} \rangle \\
\boldsymbol{\mu}(\mathbf{s}) &= 0
\end{aligned}$$

as claimed. ■

$$\boldsymbol{\mu}(\mathbf{s}) = 0 \tag{5}$$

The centering operation is already baked into the definition of variance and covariance, where each summand explicitly centers the feature variables by subtracting their (coordinate-wise) means. This both justifies the presence of centering in the formula and explains why centering is a natural first step in practice.

Key Idea

- Variance and covariance are shift invariant, but scale dependent. Both sample variance and sample covariance are *shift-invariant* (adding a constant offset does not change their values), but they are *affected by scaling* (multiplying the data by a factor rescales them).
- *Centering/ Zero centering.* Centering is already baked into the definitions of sample variance and sample covariance via the terms $\big(x^{(i)} - \mu(x)\big)$ and $\big(y^{(i)} - \mu(y)\big)$ Equivalently, by shift invariance, subtracting the sample mean — i.e., adding the same offset to every data point so that the sample mean becomes zero —*does not change* the sample variance or sample covariance. In practice, centering is still useful: it often improves numerical stability and makes the geometry and algebra cleaner. However, if the *intrinsic scale* of the features is large (or differs greatly across features), then variance/covariance magnitudes can be dominated by units. In that case, one typically resorts to *normalization/standardization* to remove scale effects.

5.2. Practical setup: How to compute the Covariance Matrix?

Key Idea

To compute the entire covariance matrix, all we need to do is start with the centered data.

1. Center the data matrix.
$$C_{\text{row}} \;:=\; X_{\text{row}} - \mathbf{1}_n\,\boldsymbol{\mu}^{\top},$$
where $\mathbf{1}_n \in \mathbb{R}^n$ is the all-ones column vector and $\boldsymbol{\mu} \in \mathbb{R}^d$ is the sample mean vector.
2. Compute the scaled feature–feature Gram matrix.
$$\Sigma \;:=\; \frac{1}{n-1} \quad \underbrace{C_{\text{row}}^{\top} C_{\text{row}}}_{\text{feature–feature Gram matrix}} \quad \in\; \mathbb{R}^{d\times d}$$

That's it! — you have the sample covariance matrix. It really is that simple.

I put this up front so you can apply it immediately. If you're curious about the strengths and limitations of this construction, read on.

5.3. How to do it in Python?

Key Idea

We can compute this directly in NumPy. Let $X \in \mathbb{R}^{n \times d}$ be the data matrix, where each row is a sample and each column is a feature. We first center the columns, then form the scaled feature–feature Gram matrix.

Listing 1: Sample covariance matrix from row-sample data

```python
import numpy as np

def covariance_matrix(X: np.ndarray) -> np.ndarray:
    X = np.asarray(X, dtype=float)
    Xc = X - X.mean(axis=0, keepdims=True) # center each column (
   feature)
    return (Xc.T @ Xc) / (Xc.shape[0] - 1) # sample covariance
```

Notice that what is returned is precisely the feature–feature Gram matrix of the centered data, `Xc.T @ Xc`, where `@` is Python's binary operator for matrix multiplication. In PEP 465, `@` is introduced with the mnemonic "`@` is $\star$ for m**AT**rices." Finally, the result is scaled by the usual $\frac{1}{n-1}$ factor via `(Xc.shape[0] - 1)`.

5.4. Pairwise summary

To simplify, we focus on pairwise relationships (between two variables at a time) because they are the simplest and most tractable building blocks: stacking all pairwise covariances gives the covariance matrix, which already encodes rich linear structure. This is a modeling choice for simplicity and usefulness, not because higher-order interactions do not exist.

A key design choice in defining covariance is to quantify whether two coordinates (features) exhibit a systematic relationship. In other words, even though they are distinct dimensions, they may be statistically dependent.

Consider the centered deviations of two features (coordinates) j and k in sample i, namely

$$\big(x_j^{(i)} - \boldsymbol{\mu}_j\big) \qquad \text{and} \qquad \big(x_k^{(i)} - \boldsymbol{\mu}_k\big).$$

Here $x_j^{(i)} \in \mathbb{R}$ denotes the j-th coordinate (feature) of the i-th data vector $\mathbf{x}^{(i)} \in \mathbb{R}^d$, and $\boldsymbol{\mu}_j \in \mathbb{R}$ denotes the j-th coordinate of the sample mean vector $\boldsymbol{\mu} := \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}^{(i)}$ (similarly for k).

The question is then: what patterns in these deviations would indicate that the two quantities co-vary?

5.4.1. DERIVING THE COVARIANCE MATRIX FROM SCRATCH

.

To define covariance in a way that reuses our familiar sample-variance notation, we proceed similarly and project the centered data onto a one-dimensional subspace.

We first, fix a unit vector $\mathbf{u}$ and let $U := \operatorname{span}\{\mathbf{u}\}$. For each centered vector $\mathbf{c}_i \in \mathbb{R}^d$, the orthogonal projection onto U is

$$P_U(\mathbf{c}_i) \;=\; \langle \mathbf{c}_i, \mathbf{u} \rangle \, \mathbf{u},$$

and the associated scalar coordinate (projection coefficient) is $\langle \mathbf{c}_i, \mathbf{u} \rangle \in \mathbb{R}$.

Define the projected scalars and their dataset by

$$\begin{aligned} s_i &:= \langle \mathbf{c}_i, \mathbf{u} \rangle \in \mathbb{R}, \\ S &:= (s_1, \ldots, s_n) \in \mathbb{R}^n, \\ \boldsymbol{\mu}(\mathbf{s}) &:= \frac{1}{n} \sum_{i=1}^{n} \mathbf{s}_i. \end{aligned}$$

Plugging these scalars into the one-dimensional sample variance formula yields

$$\begin{aligned} v(S) &= \frac{1}{n-1} \sum_{i=1}^{n} (s_i - m(S))^2 \\ &= \frac{1}{n-1} \sum_{i=1}^{n} (s_i - m(S))(s_i - m(S)) \\ &= \frac{1}{n-1} \sum_{i=1}^{n} \big(\langle \mathbf{c}_i, \mathbf{u} \rangle - m(S)\big)\big(\langle \mathbf{c}_i, \mathbf{u} \rangle - m(S)\big) \quad (\text{substituting } s_i = \langle \mathbf{c}_i, \mathbf{u} \rangle) \\ &= \frac{1}{n-1} \sum_{i=1}^{n} \big(\langle \mathbf{c}_i, \mathbf{u} \rangle\big)\big(\langle \mathbf{c}_i, \mathbf{u} \rangle\big) \quad (\text{since } m(S) = 0 \text{ for centered data by lemma 2}) \\ &= \frac{1}{n-1} \sum_{i=1}^{n} \big(\mathbf{c}_i^\top \mathbf{u}\big)\big(\mathbf{c}_i^\top \mathbf{u}\big) \quad (\text{inner product as a dot product}) \\ &= \frac{1}{n-1} \sum_{i=1}^{n} \big(\mathbf{u}^\top \mathbf{c}_i\big)\big(\mathbf{c}_i^\top \mathbf{u}\big) \quad (\text{symmetry of the dot product}) \\ &= \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{u}^\top \big(\mathbf{c}_i \mathbf{c}_i^\top\big) \mathbf{u} \quad (\text{associativity of matrix multiplication}) \\ &= \mathbf{u}^\top \left(\frac{1}{n-1} \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^\top \right) \mathbf{u} \quad (\text{linearity of summation}) \\ &= \mathbf{u}^\top \underbrace{\left(\frac{1}{n-1} \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^\top \right)}_{\Sigma \in \mathbb{R}^{d \times d}} \mathbf{u}. \end{aligned}$$

We see that we arrived at the quadratic form $\mathbf{u}^\top \Sigma \mathbf{u}$, and that the covariance matrix Σ arises naturally from this algebraic manipulation.

Recall that a quadratic form is geometrically useful: it can be viewed as the inner product of a vector with its transformed version. Indeed, for any matrix A and vector $\mathbf{x}$ for which the product is defined,

$$\mathbf{x}^\top A \mathbf{x} \;=\; \langle \mathbf{x},\, A\mathbf{x} \rangle.$$

If $A = I$, then

$$\mathbf{x}^\top I\mathbf{x} = \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|_2^2,$$

which is just the squared length of $\mathbf{x}$ ("no deformation").

More generally:

- If $A\mathbf{x}$ points mostly in the same direction as $\mathbf{x}$, then $\langle \mathbf{x}, A\mathbf{x} \rangle$ is large and positive.
- If $A\mathbf{x}$ is orthogonal to $\mathbf{x}$, then $\langle \mathbf{x}, A\mathbf{x} \rangle = 0$.
- If $A\mathbf{x}$ points mostly opposite to $\mathbf{x}$, then $\langle \mathbf{x}, A\mathbf{x} \rangle$ is negative.

Intuitively, $\mathbf{x}^\top A\mathbf{x}$ measures how much the transformation A aligns with $\mathbf{x}$, weighted by the length of $A\mathbf{x}$.

Consequence. In particular, this representation implies:

- Σ must be square of size $d \times d$.
- Σ is symmetric.
- Σ is positive semidefinite.

The quadratic form immediately forces Σ to be a square $d \times d$ matrix (so that the product is well-defined), and in fact in our construction

$$\Sigma := \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^\top$$

is automatically *symmetric* and *positive semidefinite*.

Remark (scaled vs. unscaled). If we instead define the unscaled "scatter matrix" (unscaled and unnormalized covariance)

$$S \; := \; \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^\top,$$

then S is also symmetric and positive semidefinite, and $\Sigma = \frac{1}{n-1} S$ inherits the same properties since $\frac{1}{n-1} > 0$.

Proposition 4 *Let*

$$\Sigma := \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^\top \in \mathbb{R}^{d \times d}.$$

Then Σ is symmetric and positive semidefinite.

Proof (Symmetry). Notice it is symmetric because each term $\mathbf{c}_i \mathbf{c}_i^\top$ is an outer product of a vector with itself, hence

$$(\mathbf{c}_i \mathbf{c}_i^\top)^\top = \mathbf{c}_i \mathbf{c}_i^\top.$$

Therefore, Σ is a sum of symmetric matrices and is itself symmetric.

(Positive semidefinite). Moreover, it is positive semidefinite by the definiteness test: it suffices to show that $\mathbf{u}^\top \Sigma \mathbf{u} \geq 0$ for all $\mathbf{u} \in \mathbb{R}^d$. Indeed, if we manipulate symbolically, since it has to hold for all $\mathbf{u}$,

$$\mathbf{u}^\top \Sigma \mathbf{u} = \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{u}^\top (\mathbf{c}_i \mathbf{c}_i^\top) \mathbf{u} = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{u}^\top \mathbf{c}_i)^2 \geq 0,$$

and the sum of squares is always nonnegative. ■

The fully expanded form of the sample covariance matrix is

$$\Sigma := \frac{1}{n-1} \sum_{i=1}^{n} \big(\mathbf{x}^{(i)} - \boldsymbol{\mu}(\mathbf{x})\big)\big(\mathbf{x}^{(i)} - \boldsymbol{\mu}(\mathbf{x})\big)^\top \in \mathbb{R}^{d \times d}.$$

and entry-wise using indexing notation

$$\Sigma_{jk} := \frac{1}{n-1} \sum_{i=1}^{n} \big(x_j^{(i)} - \boldsymbol{\mu}(\mathbf{x})_j\big)\big(x_k^{(i)} - \boldsymbol{\mu}(\mathbf{x})_k\big).$$

Since $x_j^{(i)}$ and $\boldsymbol{\mu}(\mathbf{x})_j$ are scalars, no transpose is needed in this entry-wise expression.

This exactly matches the fully expanded form of the sample covariance $\widehat{\mathrm{Cov}}(j, k)$ in (4), where each entry measures the covariance between a pair of feature variables. Moreover, the entry-wise formula is symmetric in the indices (i.e., $\Sigma_{jk} = \Sigma_{kj}$), so it is clear why the resulting matrix is symmetric.

Difference compared to variance. Notice that we are still comparing deviations within the same datum, but now across *distinct* coordinates. Unlike variance, the sign now matters: if the two centered coordinates have opposite signs, the product in the summand is negative; if they have the same sign, the product is positive. Geometrically, since we are working with two variables, we can visualize each centered pair in the 2-dimensional subspace spanned by the corresponding coordinate axes (for simplicity, think of the standard basis axes). Points in the first and third quadrants yield a positive product, whereas points in the second and fourth quadrants yield a negative product.

The magnitudes of the deviations also matter — large deviations in either coordinate will scale the product, so extreme values can have a disproportionate effect on the covariance. We will revisit this formally later in the section on *sensitivity to outliers* 5.6, which is an intrinsic limitation of covariance-based summaries. Finally, the total contribution is averaged across the dataset; when defining the usual unbiased estimator, we use the factor $\frac{1}{n-1}$.

Key Idea

- The covariance between two features (i.e., a single off-diagonal entry of the covariance matrix) is **a purely *linear directional* summary**: it is ***not* a claim about the exact shape** of the joint distribution. It only indicates whether the *centered* pairs tend to place more mass along one diagonal direction versus the other.
- The **covariance matrix** can be written as a scaled *feature–feature Gram matrix*. Hence it is **square, symmetric, and positive semidefinite**, which opens up a plethora of spectral tools (e.g., eigenvalue–eigenvector analysis) for understanding directions of variation.

5.5. Covariance as *linear* co-variation (The linear directional argument)

This is a good time to segue into geometry in a 2-dimensional coordinate system and what covariance captures geometrically. After centering, each sample contributes the product of its two coordinate deviations, so the sign and magnitude of this contribution depend on which quadrant the point lies in and how far it is from the origin. Equivalently, covariance reflects how strongly the centered cloud hugs a diagonal direction, and how far it extends along that diagonal.

If many points with large Euclidean norm lie predominantly in the diagonally opposite quadrants $(+,+)$ and $(-,-)$, then most products are positive and the covariance is positive (often large). If, instead, most of the mass lies in $(+,-)$ and $(-,+)$, then most products are negative and the covariance is negative (often large in magnitude). By contrast, if the cloud is roughly balanced across these diagonal quadrants, then positive and negative contributions tend to cancel and the covariance is close to zero.

A useful geometric picture is a scatter plot of the (possibly centered) pairs in the plane. For intuition, you may temporarily ignore centering and simply imagine plotting the raw pairs (a, b). The qualitative picture is similar, although centering makes the interpretation cleaner by shifting the natural reference point to the origin.

In general, we can distinguish three qualitative cases:

- **No clear linear relationship.** The point cloud has no visibly preferred direction (it looks roughly isotropic).

- **Positive co-variation.** Larger values of a tend to occur with larger values of b (and smaller with smaller), so the cloud shows a clear tendency to align with the $45°$ diagonal; after centering, many points fall in the $(+,+)$ and $(-,-)$ quadrants.

- **Negative co-variation.** Larger values of a tend to occur with smaller values of b (and vice versa), so the cloud shows a clear tendency to align with the $-45°$ diagonal; after centering, many points fall in the $(+,-)$ and $(-,+)$ quadrants.

Importantly, this is **a *linear directional* statement, not a claim about the exact shape of the joint distribution**. Covariance only can indicate whether the centered pairs tend to place more mass along one diagonal direction versus the other: positive co-variation corresponds to a stronger tendency toward the $45°$ diagonal, whereas negative co-variation corresponds to a stronger tendency toward the $-45°$ diagonal. If there is no preferred diagonal tendency, the positive and negative products tend to balance, and the covariance is weak (near zero). (A proof of this is provided in 6)

Thus, the diagonal directions explain why covariance primarily reflects *linear* co-movement: it measures whether the centered cloud is elongated along the $45°$ diagonal (positive), along the $-45°$ diagonal (negative), or has no preferred diagonal direction (covariance near zero). Here "linear" refers to the best linear trend suggested by the scatter plot, not to the full underlying relationship, which may still contain nonlinear structure; covariance is insensitive to purely nonlinear dependence.

A ***common misconception is to conflate the covariance matrix with the Gaussian distribution***, likely because the two appear together so often. They are not the same object! The covariance matrix is a "second-moment" summary of data, whereas the Gaussian distribution is a parametric model that *uses* a covariance matrix as one of its parameters. A Gaussian distribution is an additional modeling step, where we fit a *parametric* distribution for sampling or statistical inference, whose

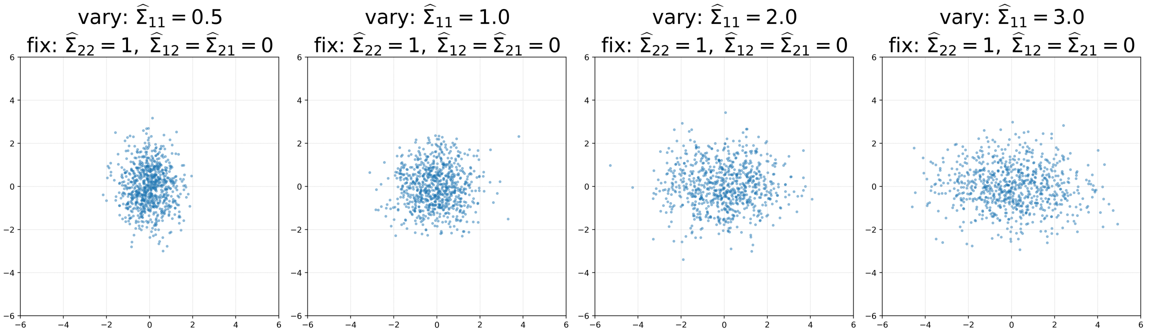


Figure 2: Vary: Σ_{11}

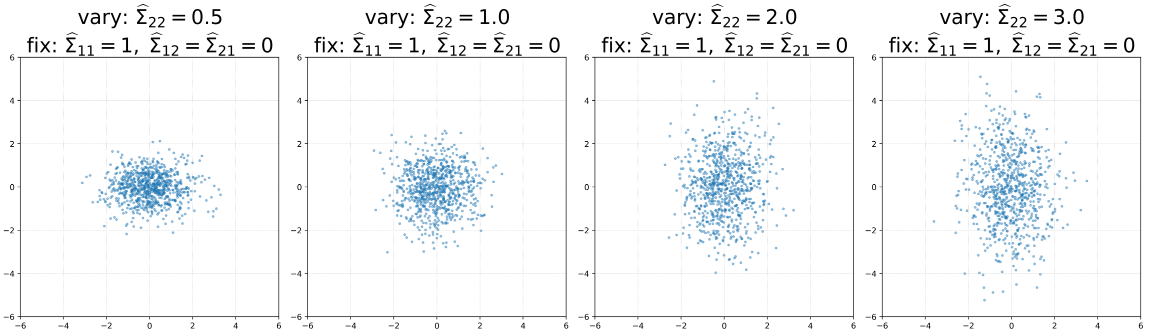


Figure 3: Vary: Σ_{22}

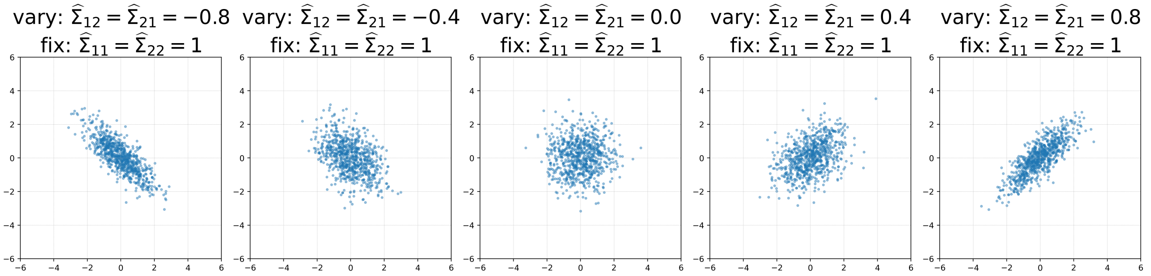
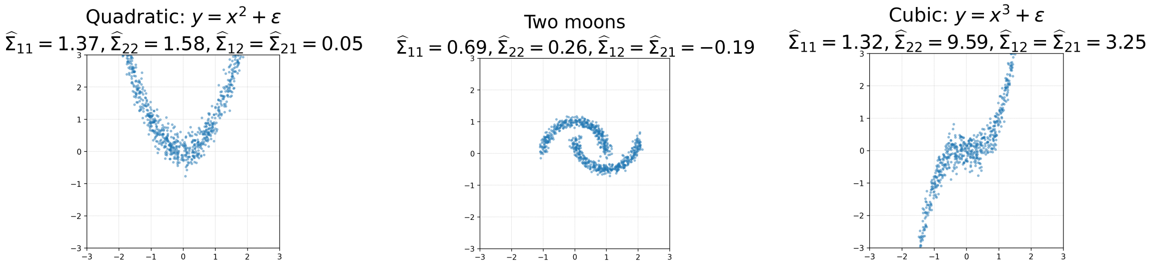


Figure 4: Vary: $\Sigma_{12} = \Sigma_{21}$

parameters include a mean vector and a covariance matrix. By contrast, the covariance matrix alone is already a powerful tool for feature screening and preprocessing. It provides descriptive statistics about the dataset—most notably, how features co-vary in a *linear* (second-moment) sense.

Importantly, **covariance does *not* require the data to be Gaussian**: it is a *model-agnostic* second-moment summary, and it does not assume any particular parametric family. For a finite dataset with $n \geq 2$, the sample covariance can always be computed. This means, the covariance matrix, Σ by itself may *not* reveal the complete shape of the joint distribution (e.g., multimodality or curved/ nonlinear structure), and an entry Σ_{jk} may be small or even zero despite a strong *nonlinear* relationship between features j and k. For this reason, it is good practice to complement covariance with pairwise scatter plots (or other dependence diagnostics) when assessing relationships between features.



(a) Samples concentrate around a quadratic parabola

(b) Samples that resemble two-moons

(c) samples concentrate around a cubic curve

Figure 5: The covariance matrix can still be computed with no Gaussian assumption required

This is where things get interesting. To understand the joint behavior between two features, we can take another view and encode each feature's real-valued deviations across the dataset as a vector. That is, we fix one *feature coordinate across the dataset*. For example, we can fix the length-n real-valued sequence

$$\mathbf{a} = (a^{(1)}, \ldots, a^{(n)}) \in \mathbb{R}^n,$$

where $a^{(i)} := \mathbf{x}_j^{(i)}$ is the j-th coordinate (feature) extracted from the i-th sample. Similarly, for another feature we define

$$\mathbf{b} = (b^{(1)}, \ldots, b^{(n)}) \in \mathbb{R}^n,$$

where $b^{(i)} := \mathbf{x}_k^{(i)}$ is the k-th coordinate extracted from the i-th sample.

Recall, we can utilize the dot product as an *alignment* measure (a term that is now widely used in machine learning as an interpretation of "similarity", often called the "*cosine similarity*"). Because geometrically, under the Euclidean inner product, we see

$$\mathbf{a}^\top \mathbf{b} = \langle \mathbf{a}, \mathbf{b} \rangle = \|\mathbf{a}\|_2 \, \|\mathbf{b}\|_2 \cos\theta,$$

where the dot product decomposes into a *magnitude term* $\|\mathbf{a}\|_2 \|\mathbf{b}\|_2$ and an *alignment term* $\cos\theta$, which encodes the angular difference between the two vectors. Recall the general results.

- $\theta = 0$: same direction (max positive alignment)
- $\theta = \pi/2$: orthogonal (zero dot product)

- $\theta = \pi$: opposite direction (max negative alignment)

When we encode each feature coordinate across the dataset as a vector, we can interpret the dot product as measuring the *alignment* of these two feature-vectors — that is, how similarly the two features co-vary across the dataset (same sign and large magnitudes contribute positively, whereas opposite signs contribute negatively).

The first interesting result is to apply Cauchy–Schwarz to yield an upper bound on the magnitude of this alignment value.

Lemma 5 (Cauchy–Schwarz bound for sample covariance) *Let* $(a^{(i)}, b^{(i)}) \in \mathbb{R}^2$ *be the centered pairs*

$$a^{(i)} := x_j^{(i)} - \boldsymbol{\mu}_j, \qquad b^{(i)} := x_k^{(i)} - \boldsymbol{\mu}_k, \qquad i \in \{1, \dots, n\}.$$

Define the sample covariance

$$\widehat{\mathrm{Cov}}(j,k) := \frac{1}{n-1} \sum_{i=1}^{n} a^{(i)} b^{(i)}.$$

Then

$$\big|\widehat{\mathrm{Cov}}(j,k)\big| \le \sqrt{v(j)}\,\sqrt{v(k)},$$

where

$$v(j) := \frac{1}{n-1} \sum_{i=1}^{n} \big(a^{(i)}\big)^2, \qquad v(k) := \frac{1}{n-1} \sum_{i=1}^{n} \big(b^{(i)}\big)^2.$$

Proof By definition,

$$\widehat{\mathrm{Cov}}(j,k) = \frac{1}{n-1} \sum_{i=1}^{n} a^{(i)} b^{(i)}$$

Now collect these centered coordinates into vectors

$$\mathbf{a} := \big(a^{(1)}, \dots, a^{(n)}\big) \in \mathbb{R}^n, \qquad \mathbf{b} := \big(b^{(1)}, \dots, b^{(n)}\big) \in \mathbb{R}^n$$

Then the covariance is a scaled dot product,

$$\widehat{\mathrm{Cov}}(j,k) = \frac{1}{n-1}\, \mathbf{a}^\top \mathbf{b}$$

Applying Cauchy-Schwarz gives the clean upper bound on the expression

$$\big|\mathbf{a}^\top \mathbf{b}\big| \le \|\mathbf{a}\|_2 \, \|\mathbf{b}\|_2.$$

Divide by $(n-1)$ and rewrite each norm:

$$\begin{aligned}
\big|\widehat{\mathrm{Cov}}(j,k)\big| &= \left|\frac{1}{n-1}\, \mathbf{a}^\top \mathbf{b}\right| \\
&\le \frac{1}{n-1}\, \|\mathbf{a}\|_2 \, \|\mathbf{b}\|_2 \qquad \text{(Cauchy–Schwarz)} \\
&= \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \big(a^{(i)}\big)^2}\; \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \big(b^{(i)}\big)^2} \\
&= \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \big(a^{(i)} - \underbrace{0}_{\mu(a)}\big)^2}\; \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \big(b^{(i)} - \underbrace{0}_{\mu(b)}\big)^2} \qquad \text{(More explicitly when data is centered)}
\end{aligned}$$

Define the (empirical) variances of the centered sequences

$$v(j) := \frac{1}{n-1}\sum_{i=1}^{n}\big(a^{(i)}\big)^2, \qquad v(k) := \frac{1}{n-1}\sum_{i=1}^{n}\big(b^{(i)}\big)^2.$$

Then the bound becomes

$$\big|\widehat{\mathrm{Cov}}(j,k)\big| \le \sqrt{v(j)}\,\sqrt{v(k)},$$

or

$$\big|\widehat{\mathrm{Cov}}(j,k)\big| \le \sqrt{v(j)\,v(k)},$$

This upper bound holds *entry-wise* for the covariance matrix: for every pair of features (j,k), the corresponding covariance entry satisfies

$$\big|\Sigma_{jk}\big| \;\le\; \sqrt{\Sigma_{jj}\,\Sigma_{kk}}.$$

which applies to *each* pairwise feature combination.

■

We see that Cauchy–Schwarz gives us an upper bound whose scale is characterized by the variance of each centered feature sequence.

A natural extension to the above analysis is to ask: **what is the behavior of the data when we attain the *maximum possible magnitude* of the sample covariance for a fixed dataset?** In that case, we must achieve equality in the Cauchy–Schwarz inequality.

$$\big|\mathbf{a}^\top\mathbf{b}\big| = \|\mathbf{a}\|_2\,\|\mathbf{b}\|_2$$

There are several ways to see what the solution must look like — how one feature vector must behave when the other is fixed. We will present two: an intuitive geometric argument, and a more principled derivation via constrained optimization.

Technically, in an empirical dataset, the vectors $\mathbf{a}$ and $\mathbf{b}$ are fully dictated by the samples and are therefore fixed. Hence, this treatment is primarily for pedagogical purposes. However, to understand the *extremal behavior* of a dot product (and hence of covariance-like quantities), it is instructive to fix $\mathbf{a}$ and ask:

> Among all vectors $\mathbf{b}$ with a fixed Euclidean length, which one maximizes $\mathbf{a}^\top\mathbf{b}$?

Theorem 6 (Maximizing a dot product under a norm constraint) *Fix $\mathbf{a} \in \mathbb{R}^d$ and fix a radius $r > 0$. Consider the optimization problem*

$$\max_{\mathbf{b}\in\mathbb{R}^d} \mathbf{a}^\top\mathbf{b} \qquad \textit{subject to} \qquad \|\mathbf{b}\|_2 = r.$$

Then the maximum value is $\|\mathbf{a}\|_2\, r$, and an optimizer is

$$\mathbf{b}^\star = \frac{r}{\|\mathbf{a}\|_2}\,\mathbf{a}.$$

Note. *The norm constraint is necessary because, without it, the objective can be made arbitrarily large by scaling $\mathbf{b}$. Also note that **this formulation is different from the PCA optimization***

$\max_{\|\mathbf{u}\|_2=1}\ \mathbf{u}^\top \Sigma\, \mathbf{u}$. *The problem above is a "purely geometric" statement and should not be confused with the "direction of maximal variance" problem in PCA. The optimization formulation we consider does* not *optimize over the dataset itself. Instead, we fix one* feature coordinate across the dataset—*that is, the length-n real-valued sequence $\mathbf{a} = (a^{(1)}, \dots, a^{(n)}) \in \mathbb{R}^n$ where $a^{(i)}$ is the j-th coordinate (feature) extracted from the i-th sample (typically after centering). We then allow the second feature coordinate sequence $\mathbf{b} = (b^{(1)}, \dots, b^{(n)}) \in \mathbb{R}^n$ to vary subject to a norm constraint $\|\mathbf{b}\|_2 = r$. In PCA, the data are fixed, and we seek a direction $\mathbf{u}$ that maximizes the* sample variance of the projected data*, where Σ is the fixed sample covariance matrix computed from the dataset.*

Proof We give two proofs: one via Cauchy–Schwarz, and a second via constrained optimization. In the second approach, the objective ($f(\mathbf{b}) = \mathbf{a}^\top \mathbf{b}$) is affine, and the equality constraint $\|\mathbf{b}\|_2 = r$ defines the Euclidean sphere (a smooth constraint set). Since both the objective and the constraint function are smooth, we can solve the problem using the Lagrange-multiplier (first-order KKT) conditions for smooth equality-constrained optimization.

(1) Intuitive way (Cauchy–Schwarz). By Cauchy–Schwarz,

$$\mathbf{a}^\top \mathbf{b} \le \|\mathbf{a}\|_2\, \|\mathbf{b}\|_2 = \|\mathbf{a}\|_2\, r,$$

so the maximum possible value is at most $\|\mathbf{a}\|_2 r$.

Moreover, equality holds if and only if $\mathbf{b}$ is colinear with $\mathbf{a}$, i.e.,

$$\mathbf{b} = c\,\mathbf{a} \quad \text{for some } c \in \mathbb{R}.$$

Imposing the constraint determines c:

$$\begin{aligned} \|\mathbf{b}\|_2 &= r \\ \|c\,\mathbf{a}\|_2 &= r \\ |c|\, \|\mathbf{a}\|_2 &= r. \end{aligned}$$

Thus

$$|c| = \frac{r}{\|\mathbf{a}\|_2} \qquad \Longrightarrow \qquad c = \pm \frac{r}{\|\mathbf{a}\|_2},$$

and therefore the only feasible colinear candidates are

$$\mathbf{b} = \pm \frac{r}{\|\mathbf{a}\|_2}\, \mathbf{a}.$$

Substituting back yields the two extremal values of the objective.

Since we are *maximizing* $\mathbf{a}^\top \mathbf{b}$, we select the positive sign (the same-direction case), because it attains the larger value:

$$\begin{aligned} \mathbf{b} &:= +\frac{r}{\|\mathbf{a}\|_2} \mathbf{a} \\ \mathbf{a}^\top \mathbf{b} &= \mathbf{a}^\top \left(+\frac{r}{\|\mathbf{a}\|_2} \mathbf{a} \right) \\ &= \frac{r}{\|\mathbf{a}\|_2}\, \mathbf{a}^\top \mathbf{a} \\ &= \frac{r}{\|\mathbf{a}\|_2}\, \|\mathbf{a}\|_2^2 \\ &= \|\mathbf{a}\|_2\, r. \end{aligned}$$

Therefore an optimizer is

$$\mathbf{b}^\star = \frac{r}{\|\mathbf{a}\|_2}\,\mathbf{a},$$

and the maximum value is

$$\max_{\|\mathbf{b}\|_2=r} \mathbf{a}^\top \mathbf{b} = \|\mathbf{a}\|_2 r.$$

(2) Karush–Kuhn–Tucker conditions (Lagrange multiplier method). We write the problem as

$$\max_{\mathbf{b}\in\mathbb{R}^d} \mathbf{a}^\top\mathbf{b} \qquad \text{subject to} \qquad \|\mathbf{b}\|_2 - r = 0.$$

Equivalently, we can rewrite the maximization problem as a minimization problem by negating the objective:

$$\min_{\mathbf{b}\in\mathbb{R}^d} -\mathbf{a}^\top\mathbf{b} \qquad \text{subject to} \qquad \|\mathbf{b}\|_2 - r = 0.$$

This is the canonical equality-constrained form used to apply the Lagrangian (a special case of the KKT conditions).

Define

$$f(\mathbf{b}) := -\mathbf{a}^\top\mathbf{b}, \qquad h(\mathbf{b}) := \|\mathbf{b}\|_2 - r,$$

and the Lagrangian

$$\mathcal{L}(\mathbf{b},\nu) := f(\mathbf{b}) + \nu\, h(\mathbf{b}) = -\mathbf{a}^\top\mathbf{b} + \nu\big(\|\mathbf{b}\|_2 - r\big).$$

KKT conditions (equality-constrained case). Assume $\mathbf{b}^\star$ is a local optimum and $\mathbf{b}^\star \neq \mathbf{0}$ (since $r > 0$, $\|\mathbf{b}^\star\|_2 = r$ forces $\mathbf{b}^\star \neq \mathbf{0}$). Then there exists $\nu^\star \in \mathbb{R}$ such that:

- **Stationarity:**
$$\nabla_{\mathbf{b}}\mathcal{L}(\mathbf{b}^\star,\nu^\star) = \mathbf{0}.$$
- **Primal feasibility:**
$$h(\mathbf{b}^\star) = 0.$$

(There is no **dual feasibility** or **complementary slackness** here because there are no inequality constraints.)

Compute the gradients:

$$\nabla_{\mathbf{b}} f(\mathbf{b}) = \nabla_{\mathbf{b}}\big(-\mathbf{a}^\top\mathbf{b}\big) = -\mathbf{a}, \qquad \nabla_{\mathbf{b}} h(\mathbf{b}) = \nabla_{\mathbf{b}}\|\mathbf{b}\|_2 = \frac{\mathbf{b}}{\|\mathbf{b}\|_2}$$

Hence stationarity becomes

$$\begin{aligned} \nabla_{\mathbf{b}} f(\mathbf{b}^\star) + \nu^\star \nabla_{\mathbf{b}} h(\mathbf{b}^\star) &= \mathbf{0} \\ -\mathbf{a} + \nu^\star \frac{\mathbf{b}^\star}{\|\mathbf{b}^\star\|_2} &= \mathbf{0} \\ \mathbf{a} &= \nu^\star \frac{\mathbf{b}^\star}{\|\mathbf{b}^\star\|_2} \end{aligned}$$

Rearranging gives

$$\mathbf{b}^\star = \frac{\|\mathbf{b}^\star\|_2}{\nu^\star}\,\mathbf{a} \tag{6}$$

Now impose primal feasibility:

$$\|\mathbf{b}^\star\|_2 = r.$$

Taking norms on both sides of (6) yields

$$\begin{aligned}\|\mathbf{b}^\star\|_2 &= \left\|\frac{\|\mathbf{b}^\star\|_2}{\nu^\star}\,\mathbf{a}\right\|_2\\ r &= \frac{\|\mathbf{b}^\star\|_2}{|\nu^\star|}\,\|\mathbf{a}\|_2\\ r &= \frac{r}{|\nu^\star|}\,\|\mathbf{a}\|_2.\end{aligned}$$

Since $r > 0$, cancel r to obtain

$$|\nu^\star| = \|\mathbf{a}\|_2.$$

At this point, (6) together with the constraint $\|\mathbf{b}^\star\|_2 = r$ imply that $\mathbf{b}^\star$ must be colinear with $\mathbf{a}$ and have length r. To *maximize* $\mathbf{a}^\top\mathbf{b}$, we choose the same-direction solution:

$$\mathbf{b}^\star = \frac{r}{\|\mathbf{a}\|_2}\,\mathbf{a}$$

The maximum value is therefore

$$\max_{\|\mathbf{b}\|_2=r}\ \mathbf{a}^\top\mathbf{b} = \|\mathbf{a}\|_2\, r.$$

■

The two proofs show that, for fixed $\mathbf{a} \in \mathbb{R}^d$ and fixed radius $r > 0$, an optimizer $\mathbf{b}^\star$ of

$$\max_{\|\mathbf{b}\|_2=r}\ \mathbf{a}^\top\mathbf{b}$$

is

$$\mathbf{b}^\star = r\,\frac{\mathbf{a}}{\|\mathbf{a}\|_2} \qquad \text{(where } r = \|\mathbf{b}\|_2 \text{ is the prescribed norm constraint).}$$

Interpreting this: $\mathbf{b}^\star$ is obtained by taking the *unit vector* in the direction of $\mathbf{a}$ and scaling it to have length r. In particular, $\mathbf{b}^\star$ is a *positive* scalar multiple of $\mathbf{a}$, so the two vectors point in the same direction.

Entrywise, this reads

$$\mathbf{b}_i^\star = r\left(\frac{\mathbf{a}}{\|\mathbf{a}\|_2}\right)_i \qquad \text{for all } i \in \{1,\dots,d\}.$$

Thus, each coordinate of $\mathbf{b}^\star$ has the same sign pattern as the corresponding coordinate of $\mathbf{a}$ (whenever that coordinate is nonzero), because the multiplier $r/\|\mathbf{a}\|_2$ is positive.

To connect back to the (a, b) scatter-plot picture in $\mathbb{R}^2$: in two dimensions, "$\mathbf{b}$ is a positive multiple of $\mathbf{a}$" means the pairs $(a^{(i)}, b^{(i)})$ line up along the line $b = ca$ with $c > 0$, i.e., the $45°$-type diagonal direction. Equivalently, the points concentrate in the $(+,+)$ and $(-,-)$ quadrants, which is exactly the sign pattern that produces predominantly positive products $a^{(i)}b^{(i)}$ and hence a large positive covariance.

Key Idea

- We can understand the cosine-based angular "alignment" of two features across a dataset via the Euclidean inner product/ dot product, as a notion of "feature similarity".
- The (sample) covariance between two features is upper bounded in magnitude by the geometric mean of their (sample) variances:

$$\left|\widehat{\mathrm{Cov}}(j,k)\right| \leq \sqrt{\widehat{\mathrm{Var}}(j)}\,\sqrt{\widehat{\mathrm{Var}}(k)}.$$

- The maximal *magnitude* of linear co-variation is attained when the centered feature vectors are colinear (Cauchy–Schwarz equality). Geometrically, this corresponds to the centered pairs concentrating along one of the two diagonal directions: the 45° diagonal yields positive covariance, while the -45° diagonal yields negative covariance. Larger magnitude corresponds to stronger concentration and/or larger deviations along the corresponding diagonal direction.

5.6. Sensitivity to outliers

To proof sensitivity to outliers if suffices to observe an introduction of a new datapoint and how it values affect the overall value.

Theorem 7 *(One extreme point can make the sample covariance grow unbounded and arbitrarily large in magnitude.)*

Proof Recall the centered (empirical) covariance form in (5.5). In this section we work in the *already-centered* setup: the centered pairs

$$a^{(i)} := x_j^{(i)} - \boldsymbol{\mu}_j, \qquad b^{(i)} := x_k^{(i)} - \boldsymbol{\mu}_k,$$

are treated as fixed numbers (equivalently, we center with respect to a fixed reference mean, so we do not re-center after adding a new point).

Define the baseline covariance (a constant determined by the given data) by

$$C_0 := \frac{1}{n-1}\sum_{i=1}^{n} a^{(i)}b^{(i)}.$$

For compactness, we denote

$$S := \sum_{i=1}^{n} a^{(i)}b^{(i)}, \qquad \text{so that} \qquad C_0 = \frac{1}{n-1}S.$$

Now, by synthetic construction, add one extreme centered pair $(c,d) \in \mathbb{R}^2$, where $|c|$ and $|d|$ can be arbitrarily large. The new dataset has size $n+1$, hence the (centered) sample covariance becomes

$$\begin{aligned} C_1 &:= \frac{1}{(n+1)-1}\,(S+cd) \\ &= \frac{1}{n}\,(S+cd)\,. \end{aligned}$$

Now we can compute the change after adding the new extreme centered pair:

$$\begin{aligned} C_1 - C_0 &= \frac{1}{n}(S+cd) - \frac{1}{n-1}S \\ &= \left(\frac{1}{n} - \frac{1}{n-1}\right)S + \frac{1}{n}cd \\ &= \left(\frac{(n-1)-n}{n(n-1)}\right)S + \frac{1}{n}cd \quad \text{(rewriting fractions with a common denominator)} \\ &= -\frac{1}{n(n-1)}S + \frac{1}{n}cd \\ &= -\frac{1}{n}\left(\frac{1}{n-1}S\right) + \frac{1}{n}cd \\ &= \frac{1}{n}\big(cd - C_0\big) \quad \text{(substituting } C_0 := \frac{1}{n-1}S\text{)} \end{aligned}$$

Since C_0 is fixed by the original data, by taking $|cd| \to \infty$ we obtain

$$|C_1 - C_0| \to \infty.$$

Thus, even a single extreme point can arbitrarily change the covariance, which shows the covariance is sensitive to outliers. ■

Key Idea

- If features vary in scale (e.g., dollars vs. rupees), then the covariance can be dominated by the magnitude of the units, rather than by the true “tightness” of the centered mass around a diagonal direction that reflects linear co-movement. Hence, the *magnitude* of an entry of the covariance matrix (or covariance alone) is ambiguous: it can be large either because the variables truly co-vary strongly, *or* simply because one (or both) variables have large scale / variance in their units.

  To isolate a scale-free notion of linear co-variation, we typically perform a *normalization* step — commonly called *standardization*, i.e., dividing by the marginal standard deviations — so that units (and overall scale) do not dominate. This effectively ablates the effect of differing magnitudes and removes the ambiguity in covariance scale, leaving a more faithful (dimensionless) measure of *linear association* (direction and strength). This normalization leads to the *Pearson correlation*, i.e., the normalized covariance:

$$\rho_{jk} := \frac{\widehat{\mathrm{Cov}}(j,k)}{\sqrt{\widehat{\mathrm{Var}}(j)}\,\sqrt{\widehat{\mathrm{Var}}(k)}}.$$

5.7. Second-moment summary

The covariance are what's called the *second-moment* summary: it depends only on products of centered deviations. In the population setting,

$$\mathrm{Cov}(\widetilde{x}, \widetilde{y}) = \mathbb{E}\big[(\widetilde{x} - \mathbb{E}[\widetilde{x}])(\widetilde{y} - \mathbb{E}[\widetilde{y}])\big],$$

and in the empirical setting each summand is exactly a product of two centered coordinates,

$$\widehat{\mathrm{Cov}}(j, k) = \frac{1}{n-1} \sum_{i=1}^{n} \big(x_j^{(i)} - \boldsymbol{\mu}_j\big)\big(x_k^{(i)} - \boldsymbol{\mu}_k\big).$$

Thus covariance summarizes co-variation through *pairwise products*, which explains why it is sensitive to large-magnitude points: squaring/multiplying amplifies outliers.

In general, one can consider *higher-order moments* of a distribution. Roughly speaking, the k-th moment (or k-th *central* moment) captures different qualitative aspects of the distribution: second-order moments relate to spread (variance), third-order moments relate to asymmetry (skewness), fourth-order moments relate to tail-heaviness (kurtosis), and so on.

Covariance is fundamentally a *second-moment* object because it is a second-order *cross* moment:

$$\mathrm{Cov}(\widetilde{x}, \widetilde{y}) = \mathbb{E}\big[(\widetilde{x} - \mathbb{E}[\widetilde{x}])(\widetilde{y} - \mathbb{E}[\widetilde{y}])\big].$$

If we were to write an analogous *third-order* centered cross-moment (sometimes called a third-order cumulant / third central cross-moment in multivariate settings), it would look like

$$\mathbb{E}\big[(\widetilde{x} - \mathbb{E}[\widetilde{x}])(\widetilde{y} - \mathbb{E}[\widetilde{y}])(\widetilde{z} - \mathbb{E}[\widetilde{z}])\big].$$

We do not usually treat this as a basic "covariance" object because: (i) it is harder to interpret geometrically than the second-order case, (ii) it is more expensive to estimate reliably (it has higher variance as an estimator and typically needs much more data), and (iii) the second-order summary already gives a clean, linear-algebraic structure (quadratic forms, PSD matrices, eigenvectors).

As an analogy, moving to higher-order moments is like moving to higher-order local information: as the order increases, the summary can capture more subtle structure, but the cost in estimation and interpretation increases rapidly.

Summary Interpretation of Covariance Matrix Entries

1. We can delineate the sign and magnitude of each entry of the covariance matrix using the following cheat sheet.

   The *sign* (signum) of Σ_{jk} indicates the **orientation of linear co-variation**:

   - $\Sigma_{jk} > 0$: the centered cloud tends to align with the 45° diagonal (many points in $(+,+)$ and $(-,-)$).
   - $\Sigma_{jk} < 0$: the centered cloud tends to align with the -45° diagonal (many points in $(+,-)$ and $(-,+)$).
   - $\Sigma_{jk} \approx 0$: positive and negative contributions tend to cancel, indicating weak (or no) *linear* co-variation.

   The *magnitude* $|\Sigma_{jk}|$ is contributed by two effects:

   (a) **Scale (marginal spread).** If feature j or k is measured on a larger scale (larger units / wider numerical range), then the centered deviations tend to be larger in absolute value. This can inflate $|\Sigma_{jk}|$ even if the underlying linear relationship is not "tighter." In this sense, "more spread" may simply reflect the *unit of measure* rather than stronger coupling.

   (b) **Strength of linear alignment.** If the centered deviations tend to move together with a consistent sign and roughly proportional magnitude, then the products

   $$(x_j^{(i)} - \boldsymbol{\mu}_j)(x_k^{(i)} - \boldsymbol{\mu}_k)$$

   reinforce rather than cancel, increasing $|\Sigma_{jk}|$.

   Because covariance is scale-dependent, people often apply an additional operation on top of centering, namely *normalization* (standardization). This leads to the correlation coefficient

   $$\rho_{jk} := \frac{\Sigma_{jk}}{\sqrt{\Sigma_{jj}\Sigma_{kk}}},$$

   which keeps the *orientation* in the sign and produces a *scale-free* magnitude. In particular, a "tighter hug" to a diagonal line corresponds to $|\rho_{jk}|$ closer to 1, whereas $|\rho_{jk}|$ near 0 indicates weak linear co-variation.

6. Asymptotic Analysis

Runtime complexity.

Recall the scaled Gram matrix in (2) from the practical setup:

$$\Sigma \;=\; \frac{1}{n-1}\, C_{\text{row}}^{\top} C_{\text{row}}.$$

This is a single matrix–matrix product, one of the most fundamental operations in numerical linear algebra. Its time complexity is well studied, and **there are many optimized implementations that accelerate it substantially in practice**.

To build intuition, consider the naive implementation if we were to code it up from scratch. Since $C_{\text{row}}^{\top} C_{\text{row}} \in \mathbb{R}^{d \times d}$, the standard index expansion of the matrix–matrix product is

$$\left(C_{\text{row}}^{\top} C_{\text{row}}\right)_{jk} = \sum_{i=1}^{n} \left(C_{\text{row}}\right)_{ij} \left(C_{\text{row}}\right)_{ik}.$$

Thus, computing a single entry (j, k) requires a sum over $i \in \{1, \ldots, n\}$, which can be implemented as one loop. Since we must compute all entries of the $d \times d$ output matrix, we additionally need two loops over (j, k), yielding a *triple-loop structure*.

```
Algorithm 1: Naive sample covariance from centered data: $\Sigma \leftarrow \frac{1}{n-1} C^{\top} C$
Input: $C \in \mathbb{R}^{n \times d}$ (centered data)
Output: $\Sigma \in \mathbb{R}^{d \times d}$
Initialize $\Sigma \leftarrow 0_{d \times d}$
for $j \leftarrow 1$ to $d$ do
    for $k \leftarrow 1$ to $d$ do
        for $i \leftarrow 1$ to $n$ do
            $\Sigma[j, k] \leftarrow \Sigma[j, k] + C[i, j] \cdot C[i, k]$
        end
        $\Sigma[j, k] \leftarrow \frac{1}{n-1}\, \Sigma[j, k]$
    end
end
return $\Sigma$
```

In terms of asymptotic runtime, the naive computation of $C_{\text{row}}^{\top} C_{\text{row}}$ performs $\Theta(nd^2)$ multiply and adds, so the tight bound is

$$\Theta(nd^2).$$

(Using the same notation, n is the number of samples and d is the number of features.)

Memory access pattern and cache locality.

A further practical concern is memory access: since C_{row} is stored in row-major order, scanning down a column (varying i while holding j fixed) has poor spatial locality. Most CPUs fetch memory in cache lines (contiguous blocks of memory), so this strided column access wastes bandwidth and can lead to more cache misses and evictions compared to contiguous row-wise access.

Space complexity.

In terms of storage, the centered data matrix $C_{\text{row}} \in \mathbb{R}^{n \times d}$ contains nd real-valued entries. Thus, storing C_{row} requires

$$\Theta(nd)$$

memory.

The covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$ requires

$$\Theta(d^2)$$

memory to store. Thus, the total space usage if both are stored is

$$\Theta(nd + d^2).$$

Let's consider the regime where one summand dominates. This is common in practice: we may have many samples (large n) or relatively few samples in a high-dimensional setting (large d). In such cases, one of the two storage terms, $\Theta(nd)$ or $\Theta(d^2)$, dominates the sum, so the overall asymptotic tight bound is determined by the larger term.

$$\Theta(nd + d^2).$$

Formally, let

$$f(n, d) := nd + d^2 \qquad \text{and} \qquad g(n, d) := \max\{nd, d^2\}.$$

Since $nd \geq 0$ and $d^2 \geq 0$, we have the two-sided bounds

$$\mathbf{g}(\mathbf{n}, \mathbf{d}) = \max\{nd, d^2\} \ \leq \ \mathbf{f}(\mathbf{n}, \mathbf{d}) = nd + d^2 \ \leq \ \max\{nd, d^2\} + \max\{nd, d^2\} = 2\max\{nd, d^2\} = \mathbf{2g}(\mathbf{n}, \mathbf{d})$$

Therefore, for all $n, d \geq 1$,

$$1 \cdot g(n, d) \ \leq \ f(n, d) \ \leq \ 2 \cdot g(n, d),$$

so by the definition of the asymptotic tight bound, $\Theta(\cdot)$ (i.e., $c_1 g \leq f \leq c_2 g$ for constants $c_1, c_2 > 0$),

$$nd + d^2 = \Theta\big(\max\{nd, d^2\}\big).$$

In particular, this yields to types of regimes:

- **Data-rich setting (**$n \gg d$**).** Then $nd \gg d^2$, so

$$\max\{nd, d^2\} = nd \quad \Longrightarrow \quad nd + d^2 = \Theta(nd).$$

- **High-dimensional setting (**$d \gg n$**).** Then $d^2 \gg nd$, so

$$\max\{nd, d^2\} = d^2 \quad \Longrightarrow \quad nd + d^2 = \Theta(d^2).$$

Key Idea

1. In the naive setting for computing the covariance matrix (i.e., without modern optimization routines), the computation uses three nested loops. Concretely, $\Sigma \in \mathbb{R}^{d \times d}$ has d^2 entries, and each entry is a dot product over the n samples:

$$\Sigma_{jk} \ = \ \frac{1}{n-1} \sum_{i=1}^{n} C_{\text{row}_{ij}} \, C_{\text{row}_{ik}}.$$

Thus we have two loops over $(j, k) \in \{1, \ldots, d\}^2$, and an inner loop over $i \in \{1, \ldots, n\}$ to accumulate the dot product. Therefore the naive runtime performs $\Theta(nd^2)$ multiply and add operations, i.e., it is quadratic in d and linear in n (a tight asymptotic bound).

7. One step beyond: The correlation matrix

Because we now have all the tools needed to define the correlation matrix, we introduce it immediately rather than treating it as a disjoint concept. We previously noted a limitation of the covariance matrix: the magnitude of an entry Σ_{jk} conflates *two* effects—the *scale* of each feature (their standard deviations) and the *strength of linear co-variation* between them. Motivated by this, we proved an absolute upper bound on centered data covariance (via Cauchy–Schwarz, as seen in 5), namely

$$\left|\Sigma_{jk}\right| \;\le\; \sqrt{\Sigma_{jj}\,\Sigma_{kk}}.$$

This bound suggests a natural normalization. We can divide the covariance by the largest magnitude it could possibly attain given the marginal scales. The resulting quantity is the *correlation*:

$$\rho_{jk} \;:=\; \frac{\Sigma_{jk}}{\sqrt{\Sigma_{jj}\,\Sigma_{kk}}},$$

whenever $\Sigma_{jj} > 0$ and $\Sigma_{kk} > 0$. We emphasize this condition because if a feature has zero variance, then the corresponding diagonal entry satisfies $\Sigma_{tt} = 0, \forall t$. In that case, the normalization factor $\sqrt{\Sigma_{jj}\Sigma_{kk}}$ equals 0, so the correlation is undefined (division by zero).

Note 1 (Zero-variance features). In practice, numerical computing libraries return `NaN` (often with a runtime warning such as "divide by zero" or "invalid value encountered") for correlations involving a zero-variance feature. A standard preprocessing step is therefore to remove constant (zero-variance) features before forming a correlation matrix.

Note 2 (Why this normalization is natural). This idea of normalization is not foreign. In probability, we normalize so that total mass equals 1 (a PMF sums to 1; a PDF integrates to 1), turning raw nonnegative weights into comparable quantities. Here, we normalize to remove scale: after dividing by $\sqrt{\Sigma_{jj}\Sigma_{kk}}$, what remains is a pure, unitless measure of linear association between features j and k.

By construction, this forces the correlation to exist in the range $\rho_{jk} \in [-1, 1]$: where the value measures "pure" *direction and linear alignment* (sign and strength) after removing the effect of scaling.

8. How is covariance matrix still useful in the era of deep learning?

While writing this section, I was somewhat ambivalent. Many modern machine learning and deep learning tools emphasize *automatic feature selection* —the idea that an algorithm can dampening irrelevant information while amplifying what is more useful. However, I was unsure how this perspective aligns with the covariance/correlation-based notions of dependence discussed above. Motivated by this, I decided to run a small experiment.