sample covariance matrix

Author and editors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland (ORCID 0000-0001-7538-0990)
Editors: Konstantina Olioumtsevits (ORCID 0000-0003-2697-8659), Ekkehard Schnoor (ORCID 0000-0001-8252-1499), Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Abstract

The sample covariance matrix of a dataset is the covariance matrix of the random vector obtained by selecting one of its data points uniformly at random. Its diagonal entries are the sample variances of the individual features, and its off-diagonal entries are the covariances between different features. Two divisor conventions are in use: dividing the sum of outer products by the number m of data points gives the covariance matrix under the empirical distribution, while dividing by $m-1$ gives an unbiased estimator of the covariance matrix of an underlying probability distribution. The two differ by the factor $m/(m-1)$ and therefore have the same eigenvectors, so a dimensionality reduction built on them does not depend on the choice.

Definition

Consider a dataset $\mathcal{D}$ of m data points, characterized by feature vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)} \in \mathbb{R}^d$. A running example is the dataset of the 1824 days of the years 2020 to 2024 at Krems an der Donau (Austria) with both readings on record, each day characterized by its air temperature at 06:00 and at 15:00 (UTC). The sample covariance matrix records how much each feature varies across the data points and how strongly two features vary together, without assuming any probabilistic model behind the measurements.

The sample covariance matrix of $\mathcal{D}$ is defined as the covariance matrix with respect to the empirical distribution $\mathbb{P}^{(\mathcal{D})}$ induced by $\mathcal{D}$. It is given explicitly by

$$\widehat{\mathbf{C}} = \frac{1}{m} \sum_{r=1}^{m} \big(\mathbf{x}^{(r)} - \widehat{\boldsymbol{\mu}}\big)\big(\mathbf{x}^{(r)} - \widehat{\boldsymbol{\mu}}\big)^{\top},$$

where $\widehat{\boldsymbol{\mu}}$ denotes the sample mean of $\mathcal{D}$. Equivalently, $\widehat{\mathbf{C}}$ is the covariance matrix of the random vector obtained by selecting a data point of $\mathcal{D}$ uniformly at random. For the weather measurements, the two diagonal entries of $\widehat{\mathbf{C}}$ are 63.01 and 85.72 squared degrees Celsius and both off-diagonal entries are 65.40 (`pythondemos/samplecovmtx.py`).

Two conventions for the divisor are in use. The definition above divides by m, which is what makes $\widehat{\mathbf{C}}$ the covariance matrix of a data point selected uniformly at random, and what the maximum likelihood method returns for

a Gaussian probabilistic model (Bishop, 2006, Sect. 2.3.4). Many texts and software libraries divide by $m-1$ instead. Under the independent and identically distributed assumption (i.i.d. assumption), with the data points modeled as realizations of independent and identically distributed (i.i.d.) random vectors, that second version is an unbiased estimator of the covariance matrix of the underlying probability distribution, while the version above is smaller on average by the factor $(m-1)/m$ (Bishop, 2006, Sect. 2.3.4). The reason is that $\widehat{\boldsymbol{\mu}}$ is computed from the same data points: the centered vectors sum to the zero vector, so only $m-1$ of them can be chosen freely.

The two versions differ by the factor $m/(m-1)$, so the choice matters only when the dataset is small. For the 1824 days it moves the first diagonal entry from 63.01 to 63.05 squared degrees Celsius. Fig. 1 shows what happens for few data points: averaged over many datasets drawn from the 1824 days, the divisor m gives half the variance of the temperature at 06:00 for two data points and is still two percent below it for one hundred, while the divisor $m-1$ sits on it throughout.

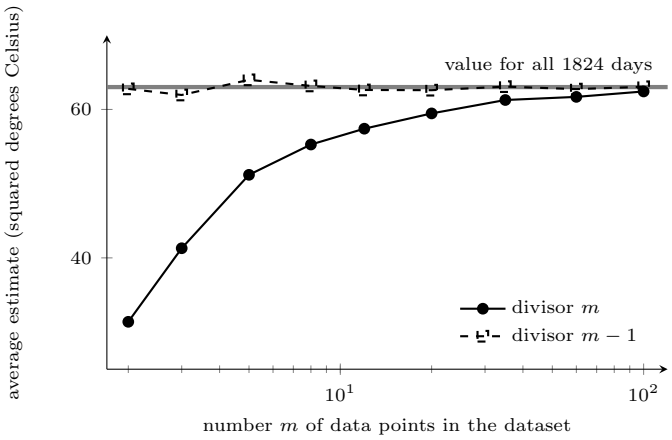

Figure 1: The two divisor conventions for the sample variance of the temperature at 06:00 at Krems an der Donau, averaged over 4000 datasets of each size drawn from the 1824 days of 2020 to 2024. The divisor m (solid curve, filled circles) approaches the value for all 1824 days (thick gray line) from below; the divisor $m-1$ (dashed curve, open squares) matches it at every size. Data generated by `pythondemos/samplecovmtx.py`

Which convention to use follows from what the matrix is for. As a summary of the dataset at hand, the divisor m is the one to pick: it makes $\widehat{\mathbf{C}}$ the covariance matrix of a data point selected uniformly at random, so every identity that holds for a covariance matrix holds for it. As an estimator of the covariance matrix of a probabilistic model behind the data points, the divisor $m-1$ removes the bias. Neither convention gives the smallest expected squared error: shrinking

the entries toward a multiple of the identity matrix trades a bias for a smaller spread and beats both when d is comparable to m (Chen et al., 2010).

In machine learning (ML), the sample covariance matrix of the feature vectors of a training set is what a method reads when it compares the features with each other rather than with the label. Principal component analysis (PCA) extracts the eigenvectors of $\widehat{\mathbf{C}}$ belonging to the largest eigenvalues and uses them as the new features (see dimensionality reduction). The choice of divisor does not move these directions: the two versions differ by a positive scalar factor, so they have the same eigenvectors, and their eigenvalues differ by that same factor (`pythondemos/samplecovmtx.py`). With centered features, $\widehat{\mathbf{C}}$ is also the matrix $(1/m)\mathbf{X}^\top\mathbf{X}$ built from the feature matrix $\mathbf{X}$, whose eigenvalues set the condition number that controls how many steps gradient descent (GD) needs for the empirical risk minimization (ERM) of linear regression (see condition number).

Cross References

covariance matrix, covariance, matrix, sample mean, sample variance, empirical distribution, variance, estimator, PCA

References

1. Bishop (2006). Pattern Recognition and Machine Learning. Springer Science+Business Media, New York, NY, USA.

2. Chen et al. (2010). Shrinkage Algorithms for MMSE Covariance Estimation. IEEE Trans. Signal Processing, 58(10), pp. 5016–5029.