Statistics for Machine Learning

Moritz Wolter

February 4, 2025

High-Performance Computing and Analytics Lab, University of Bonn

Overview

Foundational Statistical Concepts

Gaussian mixture models

Why statistics?

- Its useful and can help us make decisions when outcomes are uncertain.
- Like getting a vaccination.
- Statistics is also an integral part of machine learning. Without it, we won't understand many machine learning methods.
- Neural networks, for example, model class probabilities in the classification case.

Today's talk is mostly based on [Has22], [DFO20] and some [Unp22].

Foundational Statistical Concepts

Foundational Statistical Concepts

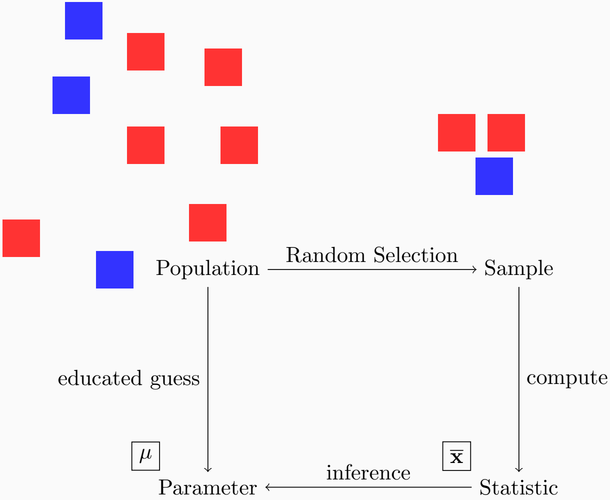


Figure 1: Statistical inference means inferring something about a population using information from samples [Has22].

Probability and random variables [DFO20]

Sample space Ω
The samples space contains all possible outcomes of an experiment. A coin toss, for example, can have two outcomes heads (h) or tails (t). Which leads to the set $\{h, t\}$. Two sucessive tosses generate the larger space $\{hh, tt, ht, th\}$.

Event space $\mathcal{A}$
A set of events, an event is a set of outcomes from the sample space.

Probability P
With each event $\mathcal{A}$ we associate a number $\mathrm{P}(\mathcal{A})$. This number measures the probability that the event will occur.

Towards distribution functions [Has22]

Random Variable

A random variable X is an uncertain quantity. Its value depends on random events. A good example is the result of a dice roll.

Probability Distribution

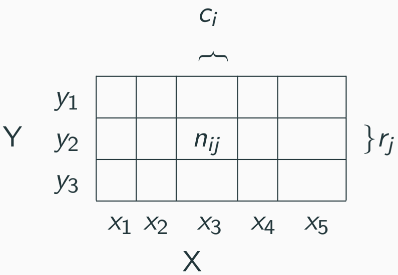
Probability density functions are a mathematical tool to describe the randomness of data in populations and samples.

Discrete probabilities [DFO20]

We can think about probabilities for multiple discrete random variables, by filling out multidimensional arrays or tables. Our arrays contain probability numbers. For two variables,

$$P(X = x_i, Y = y_i) = \frac{n_{ij}}{N} \tag{1}$$

above n_{ij} counts the events for each corresponding event x_i, y_i. And N measures all events in total.



Marginal and conditional probability

We can compute marginal probabilities by summing rows or columns.

$$P(X = x_i) = \frac{c_i}{N} = \frac{\sum_{j=1}^{3} n_{ij}}{N} \tag{2}$$

$$P(X = y_1) = \frac{r_j}{N} = \frac{\sum_{i=1}^{3} n_{ij}}{N} \tag{3}$$

The marginal probabilities allow us to define conditional probability:

$$P(Y = y_i | X = x_i) = \frac{n_{ij}}{c_i} \tag{4}$$

$$P(X = x_i | Y = y_i) = \frac{n_{ij}}{r_j} \tag{5}$$

Discrete versus continuous probability

Coin flips have discrete outcomes therefore we assign a probability to every possible event in a table.

Additionally, we can consider continuous functions, where intermediate values are also defined. This is going to be important for the Gaussian distribution.

See [DFO20] for a more formal discussion of the differences.

The Probability Density Function

In the continuous world, pdfs $p(x)$ are always positive

$$p(x) \geq 0, \ \forall x \in \mathbb{R}, \tag{6}$$

The probability for a value to end up between a and b is

$$p(a < x < b) = \int_a^b p(x) dx, \tag{7}$$

and the area under its curve must sum up to one,

$$\int_{-\infty}^{\infty} p(x) dx = 1. \tag{8}$$

Empirical mean

Typically, people mean the arithmetic mean when speaking about the mean,

$$\hat{\mu}_x = \frac{\sum_{i=1}^{n} x_i}{n}. \tag{9}$$

For the sample size $n \in 0, 1, 2, 3, \ldots$ or $\mathbb{N}$.

`np.mean` allows you to compute the mean.

Emperical variance

Variance measures the spread in the measurements of a random variable. It is defined as:

$$\hat{\sigma}_x^2 = \frac{\sum_{i=1}^{n}(x_i - \hat{\mu}_x)^2}{n-1}. \tag{10}$$

Again $n \in \mathbb{N}$ denotes the sample size. `np.var` implements this. The standard deviation is defined as the square root of the variance. Its main advantage is that it has the same dimension as the original data [Has22],

$$\hat{\sigma}_x = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \hat{\mu}_x)^2}{n-1}}. \tag{11}$$

`np.std` implements the computation of the standard deviation.

[Has22] uses $\overline{x}$ for $\hat{\mu}_x$ and s for $\hat{\sigma}_x$. Our notation is consistent width [McN16].

Mean and variance in Gaussian probability density

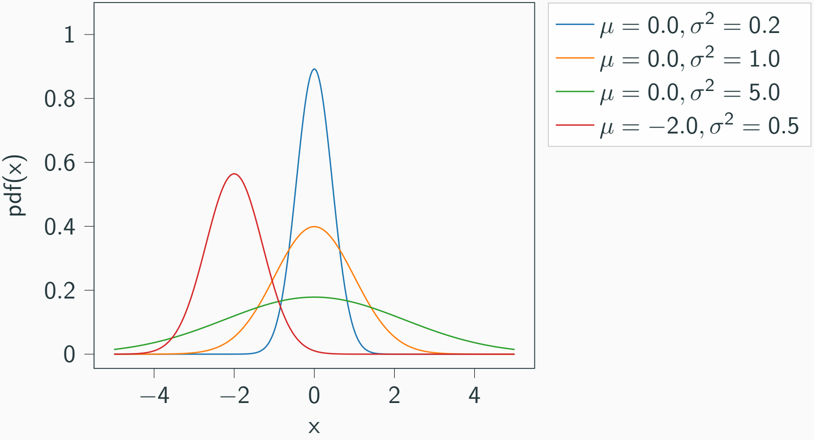
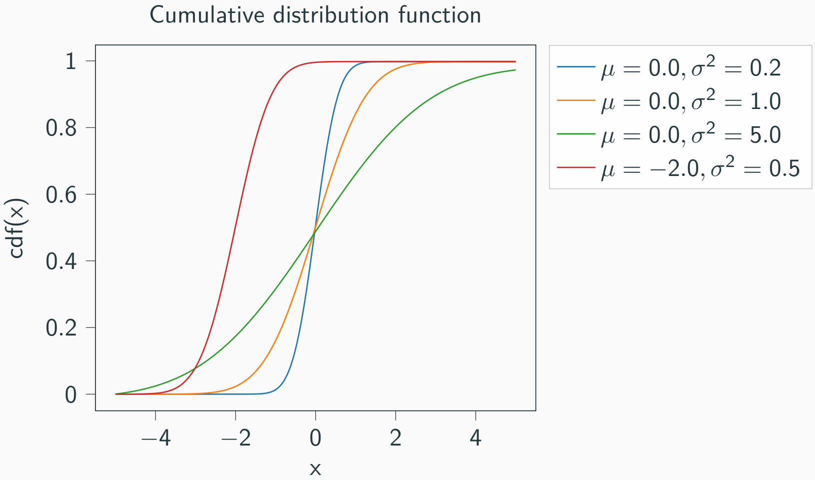


Figure 2: Normal distribution densitiy functons for different values of μ and σ. Integrating between two points on x tells us how likely the random variable will end up between those two points.

From Probability Density to Probability

Let p(x) be the Probability Density Function (PDF) of a random variable X. The integral over p(x) between a and b represents the probability of finding the value of X in that range [Has22].

The Cumulative distribution function



The Cumulative distribution function

The cumulative distribution function $P(x)$ allows us to compute the probability for a random variable X to be in a certain range.

$$P[a < X < b] = \int_a^b p(x)dx = P(b) - P(a). \tag{12}$$

Gaussian Distribution

$$f(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{13}$$

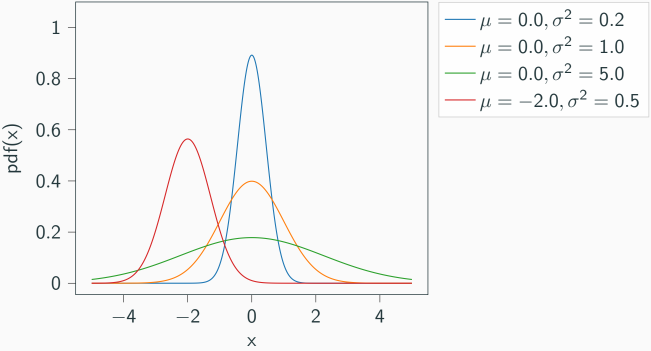


Figure 3: Plot of a Gaussian probability density function.

Uniform Distribution

$$f(x) = \begin{cases} 1/(b-a) & \text{for } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

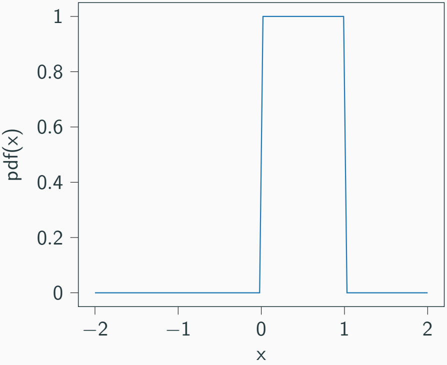


Figure 4: Plot of a uniform probability density function.

Multidimensional Probability distributions [DFO20]

The patterns we observed earlier generalize to many dimensions. The multi-dimensional view leads to functions $f : \mathbb{R}^D \to \mathbb{R}$. We expect

$$\forall \mathbf{x} \in \mathbb{R}^D : f(\mathbf{x}) > 0. \tag{15}$$

Similarly, the total area covered by the function should equal one,

$$\int_{\mathbb{R}^D} f(\mathbf{x}) d\mathbf{x} = 1. \tag{16}$$

Multivariate distributions and maginals

Continuous probability distributions can have multiple variables. Consider for example p($\mathbf{x}$, $\mathbf{y}$). In this case

$$p(\mathbf{x}) = \int_{-\infty}^{\infty} p(\mathbf{x}, \mathbf{y}) d\mathbf{y}, \tag{17}$$

$$p(\mathbf{y}) = \int_{-\infty}^{\infty} p(\mathbf{x}, \mathbf{y}) d\mathbf{x}. \tag{18}$$

In the discrete case, the integrals turn into sums [DFO20]. Let's now revisit continuous conditional probability,

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})}, \tag{19}$$

with $p(\mathbf{y}|\mathbf{x})$ instead of $p(\mathbf{y}|X = \mathbf{x})$.

Bayes Law [DFO20]

Sometimes, we have no direct way of observing a property. We are forced to infer knowledge indirectly. In such cases, Bayes law helps. Bayes states

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \tag{20}$$

The law is a consequence of our ability to factorize distributions as $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y})p(\mathbf{y})$. If we cant observe $\mathbf{x}$ directly, we may have expectations of its distribution $p(\mathbf{x})$, and the likelihood $p(\mathbf{y}|\mathbf{x})$. Bayes allows us to find a posterior $p(\mathbf{x}|\mathbf{y})$ given evidence $p(\mathbf{y})$.

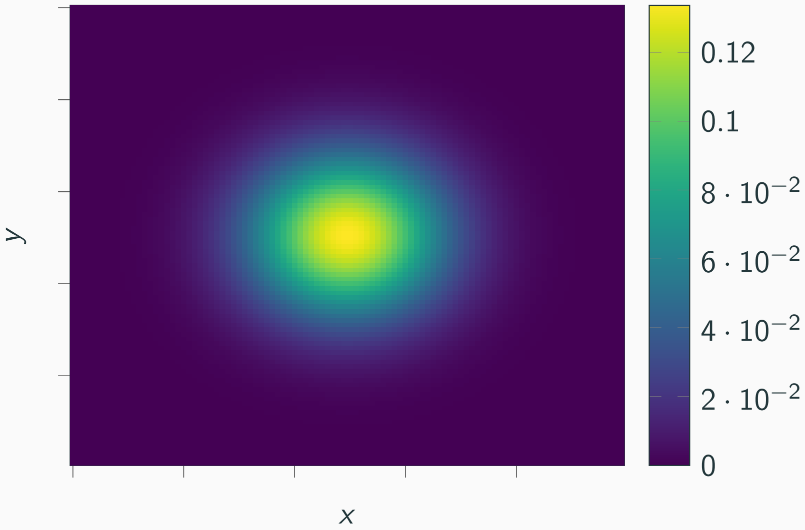
Multidimensional Gaussians

N-dimensional Gaussian pdfs are defined as [McN16],

$$\phi_2(\mathbf{x}|\mu_g, \Sigma_g) = \frac{1}{\sqrt{(2\pi)^N \|\Sigma_g\|}} \exp(-\frac{1}{2}(\mathbf{x} - \mu_g)^T \Sigma_g^{-1}(\mathbf{x} - \mu_g)). \tag{21}$$

$\mu_g \in \mathbb{R}^N$ denotes the mean vector, $\Sigma_g \in \mathbb{R}^{N \times N}$ the covariance matrix, $^{-1}$ the matrix inverse, T the transpose and $g \in \mathbb{N}$ the number of the distrubtion, which will be important later.

The Bell curve in two dimensions



Covariance

Covariance describes how two random variables "vary together"[Has22]. More formally,

$$\hat{\sigma}_{xy} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{\mu}_x)(y_i - \hat{\mu}_y) \tag{22}$$

For two n sized samples x and y and real numbers x, y and μ.

Covariance Matrix

The covariance matrix of multidimensional variables is filled with individual variables. Consider the two-dimensional case:

$$\Sigma = \begin{pmatrix} \hat{\sigma}_{xx} & \hat{\sigma}_{xy} \\ \hat{\sigma}_{yx} & \hat{\sigma}_{yy} \end{pmatrix} \tag{23}$$

Correlation

Correlation tells us how much the relationship between two random variables is linearly connected [Has22]

$$r_{xy} = \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x \hat{\sigma}_y} \tag{24}$$

$$= \frac{1}{(n-1)\hat{\sigma}_x \hat{\sigma}_y} \sum_{i=1}^{n} (x_i - \hat{\mu}_x)(y_i - \hat{\mu}_y). \tag{25}$$

Auto-Correlation

Auto-correlation [Has22] is correlation of a time delayed signal with itself. The operation is typically written as a function of the delay.

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (x_t - \hat{\mu}_x)(x_{t+k} - \hat{\mu}_x) \tag{26}$$

For a signal of length N. To allow k to move to all possible positions zeros are typically added on both sides. In the engineering literature, the normalization is typically dropped [Has22].

Auto-Correlation

autocorrelation

Gaussian mixture models

Gaussian mixture models

A Gaussian mixture model has the density [McN16]

$$f(\mathbf{x}|\theta) = \sum_{g=1}^{G} \rho_g \phi(\mathbf{x}|\mu_g, \Sigma_g). \tag{27}$$

With the normal distribution ϕ defined as before. ρ_g denotes the global probability with which a data value could originate from gaussian g. The gs number the gaussians, and G is the total number of Gaussians in the mix. We will use two. ϕ denotes the Gaussian function. Parameters μ_g and Σ_g are mean vector and covariance matrix.

Likelihood

Likelihood models the probability of data originating from a distribution as a function of the parameters. The gaussian case is modelled by [McN16]

$$\mathcal{L}_c(\theta) = \prod_{i=1}^{n} \prod_{g=1}^{G} [\rho_g \phi(\mathbf{x}_i | \mu_g, \Sigma_g)]^{z_{ig}}. \tag{28}$$

We want to maximize the likelihood.
In other words, we want to transform the bells in such a way, that they explain the points as plausible as possible.

Log-Likelihood

The log-likelihood is easier to work with consider,

$$l_c(\theta) = \sum_{i=1}^{n} \sum_{g=1}^{G} z_{ig}[\log \rho_g + \log \phi(\mathbf{x}_i|\mu_g, \Sigma_g)]. \tag{29}$$

Now the exponent is gone, and the products turned into sums. The logs rescale the bells but do not change their maxima.

Clustering using a GMM

After guessing an initial choice for all $\hat{\mu}_g$ and $\hat{\Sigma}_g$ [McN16],

$$\hat{z}_{ig} = \frac{\rho_g \phi(\mathbf{x_i} | \hat{\mu}_g, \hat{\Sigma}_g)}{\sum_{h=1}^{G} \rho_h \phi(\mathbf{x_i} | \hat{\mu}_h, \hat{\Sigma}_h)} \tag{30}$$

tells us the probability with which point x_i came from gaussian g. It creates an association between the data points and the Gaussians. Numerically evaluation results in a matrix $\mathbf{Z} \in \mathbb{R}^{\mathsf{G} \times \mathsf{n}}$. Use the maxima in it's output to select the points which belong to each class.

Fitting a GMM

Optimizing the gaussian parameters θ, requires four steps per gaussian and iteration,

1. update $\hat{z}_{ig}$.
2. update $\hat{\rho}_g = n_g/n$.
3. update $\hat{\mu}_g = \frac{1}{n_g}\sum_{i=1}^{n}\hat{z}_{ig}\mathbf{x}_i$.
4. update $\hat{\Sigma}_g = \frac{1}{n_g}\sum_{i=1}^{n}\hat{z}_{ig}(\mathbf{x}_i - \hat{\mu}_g)(\mathbf{x}_i - \hat{\mu}_g)^T$.

Above n_g denotes the number of points in class g. These four steps must be repeated until the solution is good enough.

Fitting a GMM

Gauss optimization

References

[DFO20] Marc Peter Deisenroth, A Aldo Faisal, and Cheng Soon Ong. ***Mathematics for machine learning.*** Cambridge University Press, 2020.

[Has22] Thomas Haslwanter. ***An Introduction to Statistics with Python With Applications in the Life Sciences.*** 2nd ed. Springer, 2022.

[McN16] Paul D McNicholas. ***Mixture model-based classification.*** Chapman and Hall/CRC, 2016.

[Unp22] José Unpingco. ***Python for probability, statistics, and machine learning.*** 3rd ed. Springer, 2022.