

Statistics for Machine Learning

Moritz Wolter

September 13, 2023

High-Performance Computing and Analytics Lab, University of Bonn

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

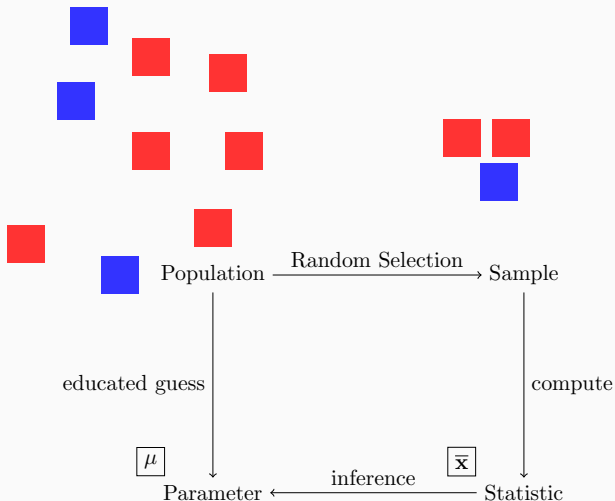


Figure 1: Statistical inference means inferring something about a population using information from samples [Has22].

Statistics for Machine Learning

Foundational Statistical Concepts

Foundational Statistical Concepts

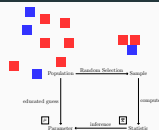


Figure 1: Statistical inference means inferring something about a population using information from samples [Hao22]

Population includes all of the elements from a set of data. Sample consists of one or more observations from the population.

Parameter Characteristic of a distribution describing a population, such as the mean or standard deviation of a normal distribution. Often notated using Greek letters.

Statistics: A numerical value that represents a property of a random sample. Examples of statistics are

- the mean value of the sample data.
- the range of the sample data.
- deviation of the data from the sample mean.

Machine learning and statistics are both concerned with the construction of models to explain data. In statistics data is observed and we try to figure out a process that explains the data. In machine learning we look for a best-fit [DFO20].

Sample space Ω

The sample 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 successive 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 $P(\mathcal{A})$. This number measures the probability that the event will occur.

Sample space Ω

The sample 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 successive 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 A we associate a number $P(A)$. This number measures the probability that the event will occur.

Lets consider a hypothetical bag. The bag contains dollar \$ and euro € coins. Of ten coins in the bag four are euro coins. If we draw twice the probability space will be the set $\{\text{€€}, \text{\$€}, \text{€\$}, \text{\$\$}\}$. We can now construct a discrete function for the random variable X . $X(\text{€}, \text{€}) = 2$, $X(\text{€}, \$) = 1$, $X(\$, \text{€}) = 1$, $X(\$, \$) = 0$

$$P(X = 2) = P((\text{€}, \text{€})) = P(\text{€}) \cdot P(\text{€}) = 0.4 \cdot 0.4 = 0.16 \quad (1)$$

$$\begin{aligned} P(X = 1) &= P((\text{€}, \$) \cup P((\$, \text{€})) = P((\text{€}, \$) + P((\$, \text{€})) \\ &= 0.4 \cdot (1 - 0.4) + (1 - 0.4) \cdot 0.4 = 0.48 \quad (2) \end{aligned}$$

$$P(X = 0) = P(\$) \cdot P(\$) = 0.6 \cdot 0.6 = 0.36 \quad (3)$$

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,

$$P(X = x_i, Y = y_i) = \frac{n_{ij}}{N} \quad (4)$$

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

		c_i						
		$\underbrace{\hspace{1.5cm}}$						
Y	y_1						$\} r_j$	
	y_2			n_{ij}				
	y_3							
		x_1	x_2	x_3	x_4	x_5		
		X						

Marginal and conditional probability

We can compute marginal probabilities by summing rows or columns.

$$P(X = x_i) = \frac{c_i}{N} = \frac{\sum_{j=1}^3 n_{ij}}{N} \quad (5)$$

$$P(X = y_1) = \frac{r_j}{N} = \frac{\sum_{i=1}^3 n_{ij}}{N} \quad (6)$$

The marginal probabilities allow us to define conditional probability:

$$P(Y = y_i | X = x_i) = \frac{n_{ij}}{c_i} \quad (7)$$

$$P(X = x_i | Y = y_i) = \frac{n_{ij}}{r_j} \quad (8)$$

Statistics for Machine Learning

Foundational Statistical Concepts

Marginal and conditional probability

Marginal and conditional probability

We can compute marginal probabilities by summing rows or columns.

$$P(X = x_i) = \frac{c_i}{N} = \frac{\sum_{j=1}^J a_{ji}}{N} \quad (5)$$

$$P(Y = y_j) = \frac{r_j}{N} = \frac{\sum_{i=1}^I a_{ij}}{N} \quad (6)$$

The marginal probabilities allow us to define conditional probability:

$$P(Y = y_j | X = x_i) = \frac{a_{ji}}{c_i} \quad (7)$$

$$P(X = x_i | Y = y_j) = \frac{a_{ji}}{r_j} \quad (8)$$

Made up data:

		$\leq 1 \text{ h}$	$> 1 \text{ h}$
y_1	Bonn	0.24	0.08
y_2	Cologne	0.04	0.32
y_3	Siegburg	0.16	0.16
		x_1	x_2

The table entries contain normalized frequencies. Row and column frequency sums are the so-called marginal probabilities.

$$\Rightarrow P(X = x_1) = 0.44. \quad (9)$$

$$\Rightarrow P(Y = y_2) = 0.36. \quad (10)$$

Probability for more than one hour from cologne:

$$P(X = x_2 | Y = y_2) = 0.32 / (0.32 + 0.04) = 0.89 \quad (11)$$

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

$$p(x) \geq 0, \forall x \in \mathbb{R}, \quad (12)$$

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

$$p(a < x < b) = \int_a^b p(x) dx, \quad (13)$$

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

$$\int_{-\infty}^{\infty} p(x) dx = 1. \quad (14)$$

Empirical mean

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

$$\hat{\mu}_x = \frac{\sum_{i=1}^n x_i}{n}. \quad (15)$$

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

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

Empirical variance

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

$$\hat{\sigma}_x^2 = \frac{\sum_{i=1}^n (x_i - \hat{\mu}_x)^2}{n - 1}. \quad (16)$$

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],

$$\hat{\sigma}_x = \sqrt{\frac{\sum_{i=1}^n (x_i - \hat{\mu}_x)^2}{n - 1}}. \quad (17)$$

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

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

Mean and variance in Gaussian probability density

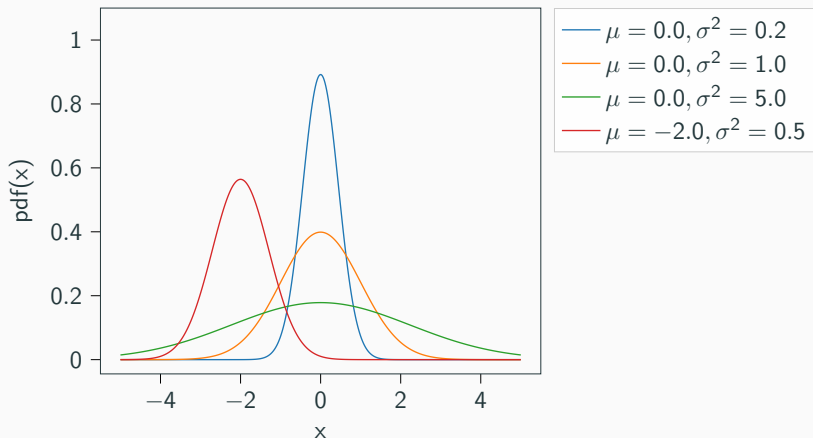
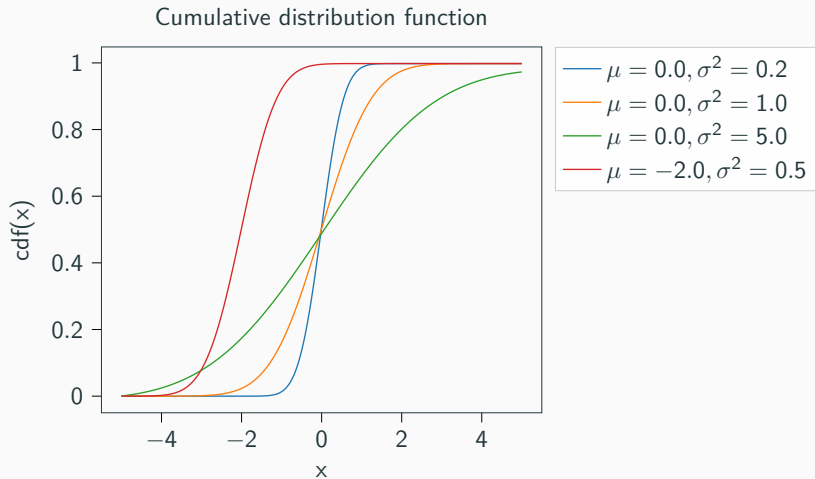


Figure 2: Normal distribution density functions for different values of μ and σ . Integrating between two points on x tells us how likely the random variable will end up between those two points.

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.

$$P[a < X < b] = \int_a^b p(x)dx = P(b) - P(a). \quad (18)$$

Gaussian Distribution

$$f(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (19)$$

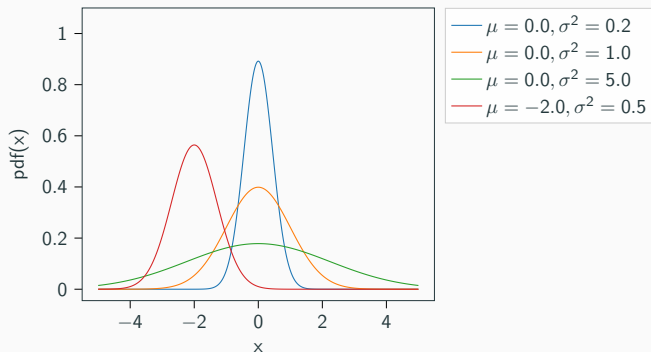


Figure 3: Plot of a Gaussian probability density function.

Uniform Distribution

$$f(x) = \begin{cases} 1/(b-a) & \text{for } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases} \quad (20)$$

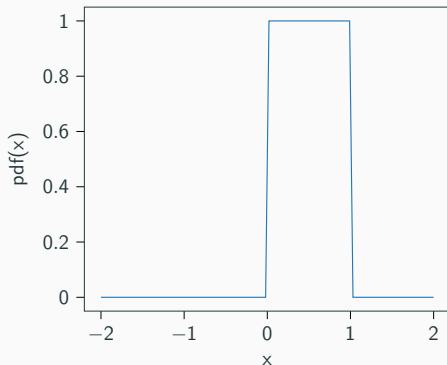


Figure 4: Plot of a uniform probability density function.

Multidimensional Probability distributions [DFO20]

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

$$\forall \mathbf{x} \in \mathbb{R}^D : f(\mathbf{x}) \geq 0. \quad (21)$$

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

$$\int_{\mathbb{R}^D} f(\mathbf{x}) d\mathbf{x} = 1. \quad (22)$$

Multivariate distributions and marginals

Continuous probability distributions can have multiple variables.

Consider for example $p(\mathbf{x}, \mathbf{y})$. In this case

$$p(\mathbf{x}) = \int_{-\infty}^{\infty} p(\mathbf{x}, \mathbf{y}) d\mathbf{y}, \quad (23)$$

$$p(\mathbf{y}) = \int_{-\infty}^{\infty} p(\mathbf{x}, \mathbf{y}) d\mathbf{x}. \quad (24)$$

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

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})}, \quad (25)$$

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

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

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \quad (26)$$

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 $p(\mathbf{y})$.

Statistics for Machine Learning

└ Foundational Statistical Concepts

└ 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

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \quad (26)$$

The law is a consequence of our ability to factorize distributions as $p(x, y) = p(x|y)p(y)$. If we can't observe x directly, we may have expectations of its distribution $p(x)$, and the likelihood $p(y|x)$. Bayes allows us to find a posterior $p(x|y)$ given evidence $p(y)$.

Say 50 in 100k people of a population have a given illness.

We have $P(S) = 0.0005$, and $P(H) = 1 - (50/100000) = 0.9995$. We have a test that detects the disease with an accuracy of 98%. In other words, $P(T|S) = 0.98$. Unfortunately, it also yields a positive result for 1% of healthy people $P(T|H) = 0.01$. What happens if we use the test to look for the disease in the general population?

$$P(S|T) = \frac{P(T|S)P(S)}{P(T|S)P(S) + P(T|H)P(H)} \quad (27)$$

$$= \frac{0.98 \cdot 0.0005}{0.98 \cdot 0.0005 + 0.01 \cdot 0.9995} = 0.05 \quad (28)$$

In this case, testing is probability not a great idea. Note, total probability for exclusive events: $P(T) = P(T|S)P(S) + P(T|H)P(H)$ if H is not S .

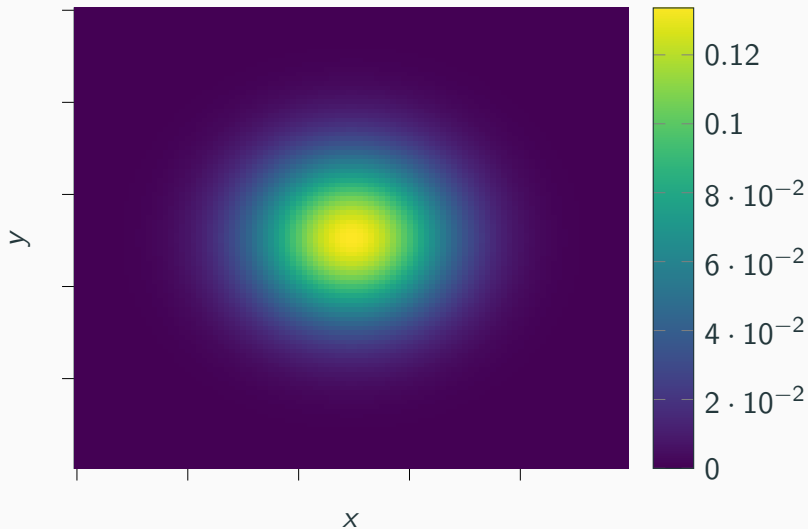
Multidimensional Gaussians

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

$$\phi_2(\mathbf{x}|\mu_g, \Sigma_g) = \frac{1}{\sqrt{(2\pi)^N \|\Sigma_g\|}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_g)^T \Sigma_g^{-1}(\mathbf{x} - \mu_g)\right). \quad (29)$$

$\mu_g \in \mathbb{R}^N$ 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 distribution, which will be important later.

The Bell curve in two dimensions



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

$$\hat{\sigma}_{xy} = \frac{1}{n-1} \sum_{i=1}^n (x_i - \hat{\mu}_x)(y_i - \hat{\mu}_y) \quad (30)$$

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

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

$$\Sigma = \begin{pmatrix} \hat{\sigma}_{xx} & \hat{\sigma}_{xy} \\ \hat{\sigma}_{yx} & \hat{\sigma}_{yy} \end{pmatrix} \quad (31)$$

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

$$r_{xy} = \frac{\hat{\sigma}_{xy}}{\hat{\sigma}_x \hat{\sigma}_y} \quad (32)$$

$$= \frac{1}{(n-1)\hat{\sigma}_x \hat{\sigma}_y} \sum_{i=1}^n (x_i - \hat{\mu}_x)(y_i - \hat{\mu}_y). \quad (33)$$

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

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (x_t - \hat{\mu}_x)(x_{t+k} - \hat{\mu}_x) \quad (34)$$

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].

autocorrelation

Gaussian mixture models

Gaussian mixture models

A Gaussian mixture model has the density [McN16]

$$f(\mathbf{x}|\theta) = \sum_{g=1}^G \rho_g \phi(\mathbf{x}|\mu_g, \Sigma_g). \quad (35)$$

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 g s 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.

Statistics for Machine Learning

└ Gaussian mixture models

└ Gaussian mixture models

A Gaussian mixture model has the density [McN16]

$$f(\mathbf{x}|\theta) = \sum_{g=1}^G \rho_g \phi(\mathbf{x}|\mu_g, \Sigma_g). \quad (35)$$

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 g s 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.

Typically we want as many g as we have classes in the data. I.e. one for healthy and one for diabetic. The data vectors are p dimensional $\mathbf{x} \in \mathbb{R}^p$. Sampling $\phi(\mathbf{x})$ tells us how likely it was to see the point we have. Big values mean it was likely small mean it was not.

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

$$\mathcal{L}_c(\theta) = \prod_{i=1}^n \prod_{g=1}^G [\rho_g \phi(\mathbf{x}_i | \mu_g, \Sigma_g)]^{z_{ig}}. \quad (36)$$

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.

Statistics for Machine Learning

└ Gaussian mixture models

└ Likelihood

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

$$\mathcal{L}_n(\theta) = \prod_{i=1}^n \prod_{g=1}^G [\rho_g \phi(\mathbf{x}_i | \mu_g, \Sigma_g)]^{z_{ig}} \quad (36)$$

We want to maximize the likelihood.

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

To maximize ϕ it needs to sit on top of the points it labels. When a gaussian sits on top of many points it's ρ_g should be large. Finally, when this works well we want a big weight from z_{ig} .

The log-likelihood is easier to work with consider,

$$l_c(\theta) = \sum_{i=1}^n \sum_{g=1}^G z_{ig} [\log \rho_g + \log \phi(\mathbf{x}_i | \mu_g, \Sigma_g)]. \quad (37)$$

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],

$$\hat{z}_{ig} = \frac{\rho_g \phi(\mathbf{x}_i | \hat{\mu}_g, \hat{\Sigma}_g)}{\sum_{h=1}^G \rho_h \phi(\mathbf{x}_i | \hat{\mu}_h, \hat{\Sigma}_h)} \quad (38)$$

tells us the probability with which point \mathbf{x}_i came from gaussian g .

It creates an association between the data points and the Gaussians. Numerically evaluation results in a matrix $\mathbf{Z} \in \mathbb{R}^{G \times n}$.

Use the maxima in it's output to select the points which belong to each class.

Statistics for Machine Learning

└ Gaussian mixture models

└ Clustering using a GMM

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

$$z_{ig} = \frac{\phi_g(x_i | \hat{\mu}_g, \hat{\Sigma}_g)}{\sum_{k=1}^K \phi_k(x_i | \hat{\mu}_k, \hat{\Sigma}_k)} \quad (38)$$

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 $\mathbf{Z} \in \mathbb{R}^{G \times n}$. Use the maxima in it's output to select the points which belong to each class.

The z_{ig} are the true labels, \hat{z}_{ig} is our estimation. The \hat{z}_{ig} are the expected value of the complete data log-likelihood. Why? ϕ is a pdf. A pdf can be interpreted as providing a relative likelihood that the value of the random variable would be equal to that sample ¹. We ask for all gaussians and every point and normalize.

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 $\hat{\Sigma}_g = \frac{1}{n_g} \sum_{i=1}^n (\mathbf{x}_i - \hat{\mu}_g)(\mathbf{x}_i - \hat{\mu}_g)^T$.

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

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.