ULB ECOLE POLYTECHNIQUE DE BRUXELLES

Statistics for machine-learning: a brief intro

Data-driven engineering course 2025

Machine-learning is part of statistics

In machine-learning, the goal is to exploit data to:

- extract physical insights,
- model physical (or non-physical) processes,
- estimate input-output relationships.

In general, we assume that the **data** is generated by a **process** that is too **complex** to represent exactly.

We divide the problem in a **simple** part, which we can **represent**, and a **complex** part, which we sweep under the rug of the **uncertainty**.

This problem can be represented implicitly or explicitly.

Example: regression

In **regression**, we want to **estimate** the functional **relationship** between the **inputs** x and the **observations** y.

The simplest way to represent this is to write:

$$y = f(x)$$

We generally accept that we will not obtain a perfect predictions of the observations because:

- the observations contain **noise**,
- we do **not** have **enough information** on the system,
- we are choosing a **simple model**.

We can make explicit the **uncertainty** in the model by writing:

$$y = f(x) + \epsilon$$

Where ϵ is a sample from some probability distribution, which we want to model.

Fitting distributions

Usually we think of **machine-learning** (especially regression) as **fitting functions** to match the data that we have collected.

In the **statistical** representation, we want to **fit** probability **distributions** to our data.

This is beneficial because:

- it allows us to have a measure of the **confidence** in our prediction,
- it improves the **robustness** of the model,
- it **reduces** the number of training **samples**.

Glossary

- A probability distribution $P(X \in E)$ assigns a probability to outcomes.
- A random variable X is a variable that represents a random process.
- An event E is a subset of the sample space.
- The sample space Ω is the set of all possible outcomes.
- Independent and identically distributed (iid) refers to random variables that are independent between each others and are distributed in the same way.
- A discrete random variable is a variable X with a finite number of possible outcomes, which has an associated probability mass function $p_\theta(x_i)$ such that $\sum_{x \in \Omega} p_\theta(x_i) = 1$
- A continuous random variable is a variable X with an infinite number of possible outcomes, which has an associated probability density function $f_\theta(x)$ such that $\int_\Omega f_\theta(x)dx = 1$

Glossary

- The expected value of a probability distribution is average value weighted by the probability:

$$\mathbb{E}[p_\theta] = \sum_{x \in \Omega} x_i \; p_\theta(x_i)$$

$$\mathbb{E}[f_\theta] = \int_\Omega x \; f_\theta(x) \; dx$$

- The sample mean is the arithmetic average from the collected sample $X_1, \ldots, X_n$:

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i$$

- Law of large numbers states that the sample mean converges to the expected value when the number of samples tends to infinity:

$$\bar{X}_n \to \mathbb{E}[p_\theta] \text{ for } n \to \infty$$

Example

My random process is flipping a fair coin, meaning that I expect to see 50% of the times heads (1) and 50% times tails (0).

The sample space is $(0,1)$ and the mathematical distribution that represents this process is a Bernoulli distribution $p_p(X) = \text{Ber}_p(X) = p^X(1-p)^{1-X}$ where $p = 0.5$.

The expected value of the distribution is $\mathbb{E}[p_p(X)] = \sum_{\Omega}^{n} X_i p^{X_i}(1-p)^{1-X_i} = p = 0.5$

If we collect some random data $X_1, \ldots, X_n \sim \text{Ber}_p(X)$ for $n = 20$ we should see something like: $[0,1,0,1,1,1,0,1,0,1,1,1,0,0,0,1,0,1]$, with an associated sample mean $\bar{X}_n = 0.55$.

How do we compute the probability

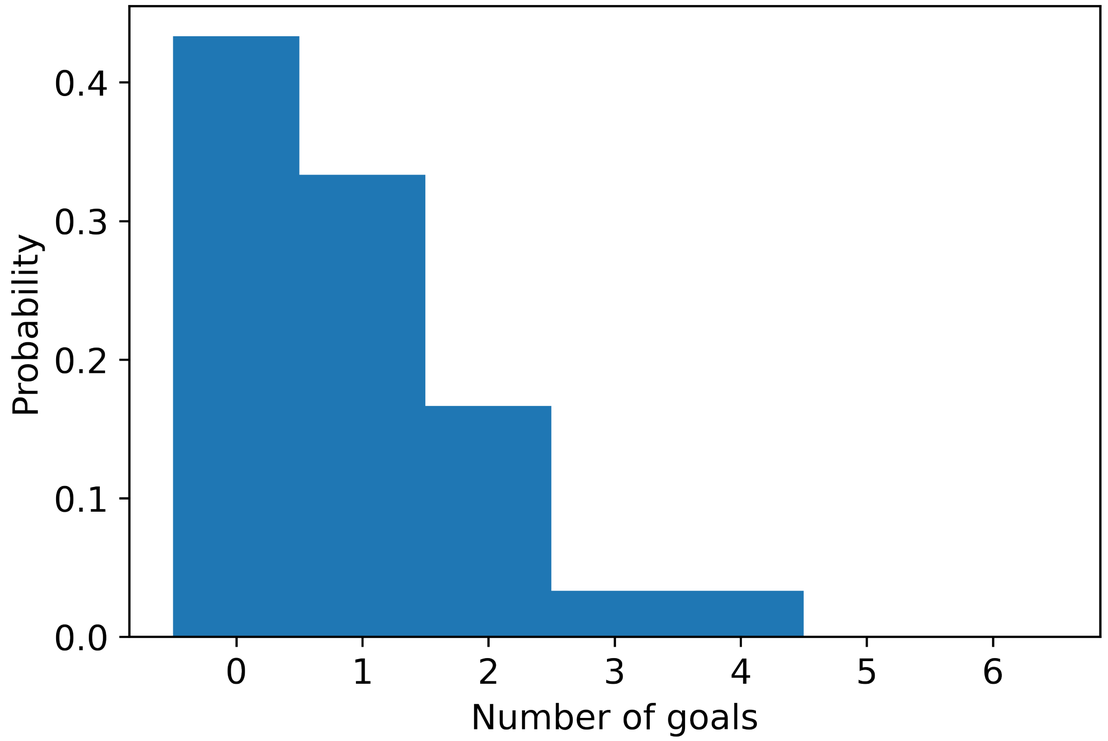
Using a frequentist approach, we can count the number of outcomes, divided by the total number of outcomes.

Example: we want to compute the probability of a player scoring in a football match.

Collect the data: [2,0,1,3,1,1,0,0,1,2,0,2,1,0,0,0,0,2,0,1,0,1,0,1,1,1,4,0,2,0]

Count the outcomes:

outcomes	0	1	2	3	4	5	6 or more
probability	0.43	0.33	0.17	0.03	0.03	0	0



Can we model it using a standard distribution

If possible, we want to model the random process using a canonical distribution.

This let us:

- reduce the number of parameters,
- reduce the data that we need to represent the distribution.

Can we model it using a standard distribution

If possible, we want to model the random process using a canonical distribution.

This let us:

- reduce the number of parameters,
- reduce the data that we need to represent the distribution.

There are different distributions that can be used to model physical systems depending on the context:

- Bernoulli: $p_p(x) = p^x(1-p)^{1-x} \quad p \in [0,1]$,
- Binomial: $p_p(x,n) = \frac{n!}{x!(n-x)!}p^x(1-p)^{n-x} \quad p \in [0,1]$,
- Poisson: $p_\lambda(x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad \lambda \in [0,\infty]$
- Gaussian: $f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{1}{2}(\frac{x-\mu}{\sigma})^2) \quad \mu \in \mathbb{R}, \sigma^2 \in [0,\infty]$

Can we model it using a standard distribution

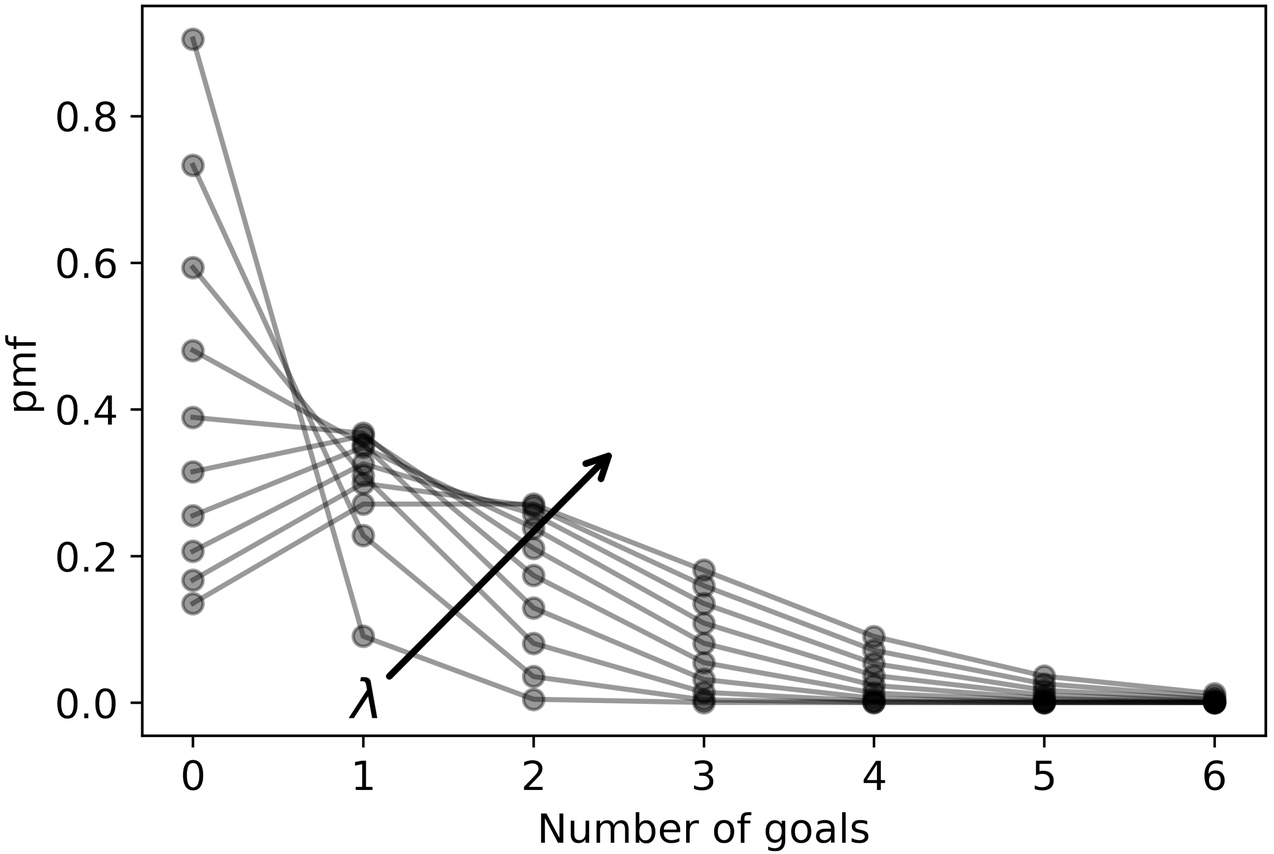
In our example, we can use the Poisson distribution:

$$p_\lambda(x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

which tells us the probability of observing x number of goals for a match.

The parameter λ controls the shape of the distribution.

The higher λ, the more goals I will expect on average.



Can we model it using a standard distribution

Now the problem becomes to find the correct λ that fits the data I have.

First, we need a measure of similarity between distributions and we use the Kullback-Leibler (KL) divergence:

$$\mathrm{KL}(P_\theta, P_{\theta'}) = \sum_{x_i \in \Omega} p_\theta(x_i) \log\left(\frac{p_\theta(x_i)}{p_{\theta'}(x_i)}\right) \quad \text{for discrete probabilities}$$

$$\mathrm{KL}(P_\theta, P_{\theta'}) = \int_\Omega f_\theta(x) \log\left(\frac{f_\theta(x)}{f_{\theta'}(x)}\right) dx \text{ for continuous probabilities}$$

Some properties of the KL divergence:

- $\mathrm{KL}(P_\theta, P_{\theta'}) \neq \mathrm{KL}(P_{\theta'}, P_\theta)$
- $\mathrm{KL}(P_\theta, P_{\theta'}) \geq 0$
- if $\mathrm{KL}(P_\theta, P_{\theta'}) = 0$ then $P_\theta = P_{\theta'}$

Maximum likelihood estimation

We can write the KL divergence in terms of the expectation:

$$\mathrm{KL}(P_{\theta*}, P_\theta) = \mathbb{E}_{\theta*}[\log(\frac{p_{\theta*}(X)}{p_\theta(X)})] = \mathbb{E}_{\theta*}[\log(p_{\theta*}(X))] - \mathbb{E}_{\theta*}[\log(p_\theta(X))]$$

where $\theta*$ is our target (fixed) and θ is our variable.

This means that $\mathrm{KL}(P_{\theta*}, P_\theta) = C - \mathbb{E}_{\theta*}[\log(p_\theta(X))]$, and also we can estimate the expected value:

$$\hat{\mathrm{KL}}(P_{\theta*}, P_\theta) = C - \frac{1}{n}\sum_{i=1}^{n} \log(p_\theta(X_i))$$

To get the two probabilities close, we want to minimize the divergence:

$$\hat{\theta} = \arg\min_\theta \hat{\mathrm{KL}}(P_{\theta*}, P_\theta) = \arg\max_\theta \prod_{i=1}^{n} p_\theta(X_i) = \arg\max_\theta L(X_1, \ldots, X_n \,|\, \theta)$$

Example on the Poisson distribution

The probability of Poisson is $p_\lambda(X_i) = \dfrac{\lambda_i^X e^{-\lambda}}{X_i!}$

The likelihood is then $L(X_1, \ldots, X_n \,|\, \lambda) = \prod_{i=1}^{n} \dfrac{\lambda^{X_i} e^{-\lambda}}{X_i!} = \dfrac{\lambda^{\sum_{i=1}^n X_i} e^{-n\lambda}}{X_1! \cdots X_n!}$ and

$$\log L = \prod_{i=1}^{n} \frac{\log(\lambda_i^X e^{-\lambda})}{\log(X_i!)} = -\log(X_1! \cdots X_n!) + \log(\lambda) \sum_{i=1}^{n} X_i - n\lambda$$

To find the maximum likelihood estimator $\hat{\lambda}$ we put the derivative of the log likelihood at zero:

$$\frac{\partial \log L}{\partial \lambda} = \frac{\partial(-log(X_1! \cdots X_n!) + \log \hat{\lambda} \sum_{i=1}^n X_i - n\hat{\lambda})}{\partial \lambda} = 0$$

$$\hat{\lambda} = \frac{1}{n} \sum_{i=1}^{n} X_i = \bar{X}_n$$

The sample average is the maximum likelihood estimator for the Poisson distribution.

Let's see if it makes sense

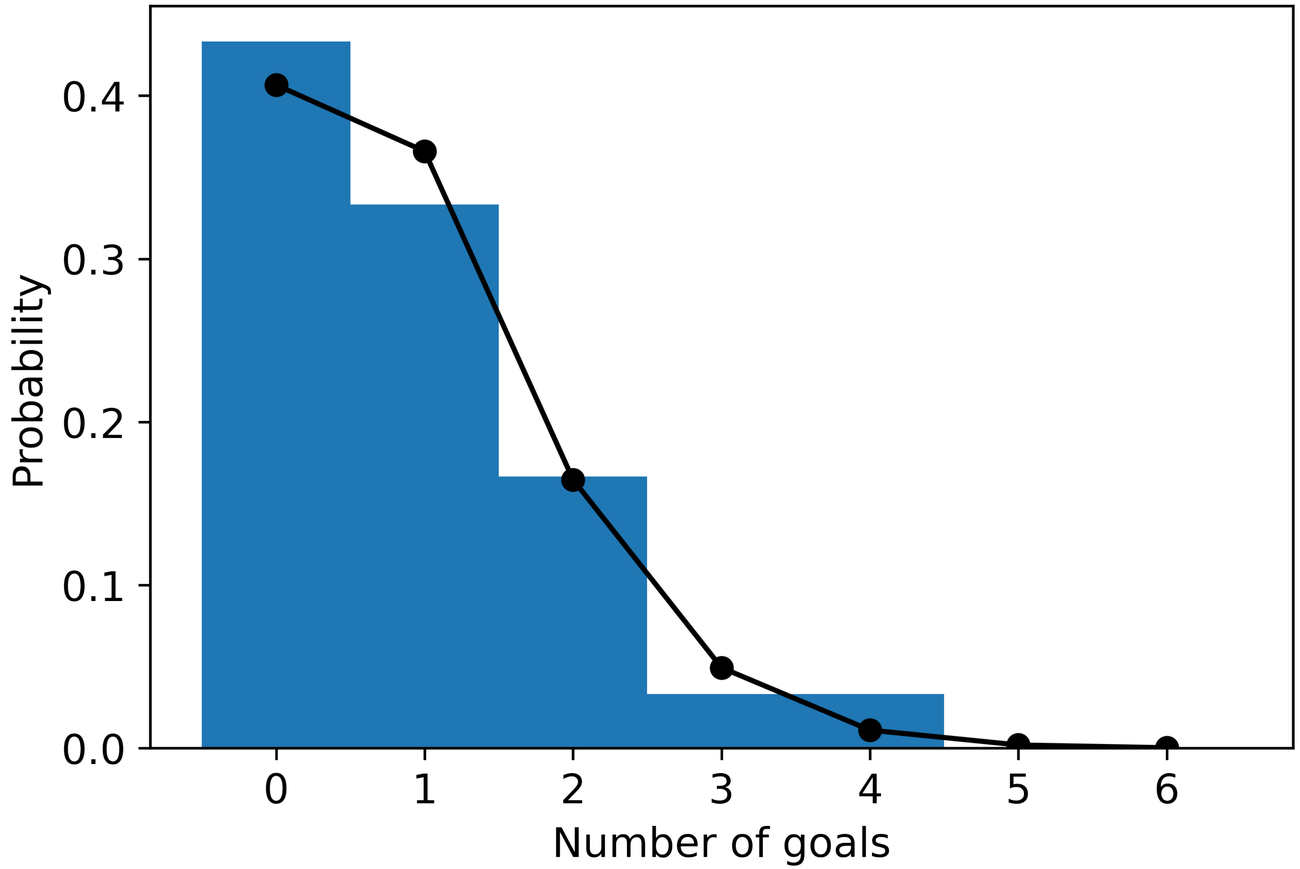
Our dataset was:

[2,0,1,3,1,1,0,0,1,2,0,2,1,0,0,0,0,2,0,1,0,1,0,1,1,1,4,0,2,0]

The sample average is equal to:

$$\frac{1}{n}\sum_{i=1}^{n} X_i = \frac{1}{30}(2+0+1+\cdots+0) = 0.9 = \hat{\lambda}$$

If we compare the histogram of the data with the Poisson distribution with $\lambda = 0.9$ we obtain a pretty good match.



Another one

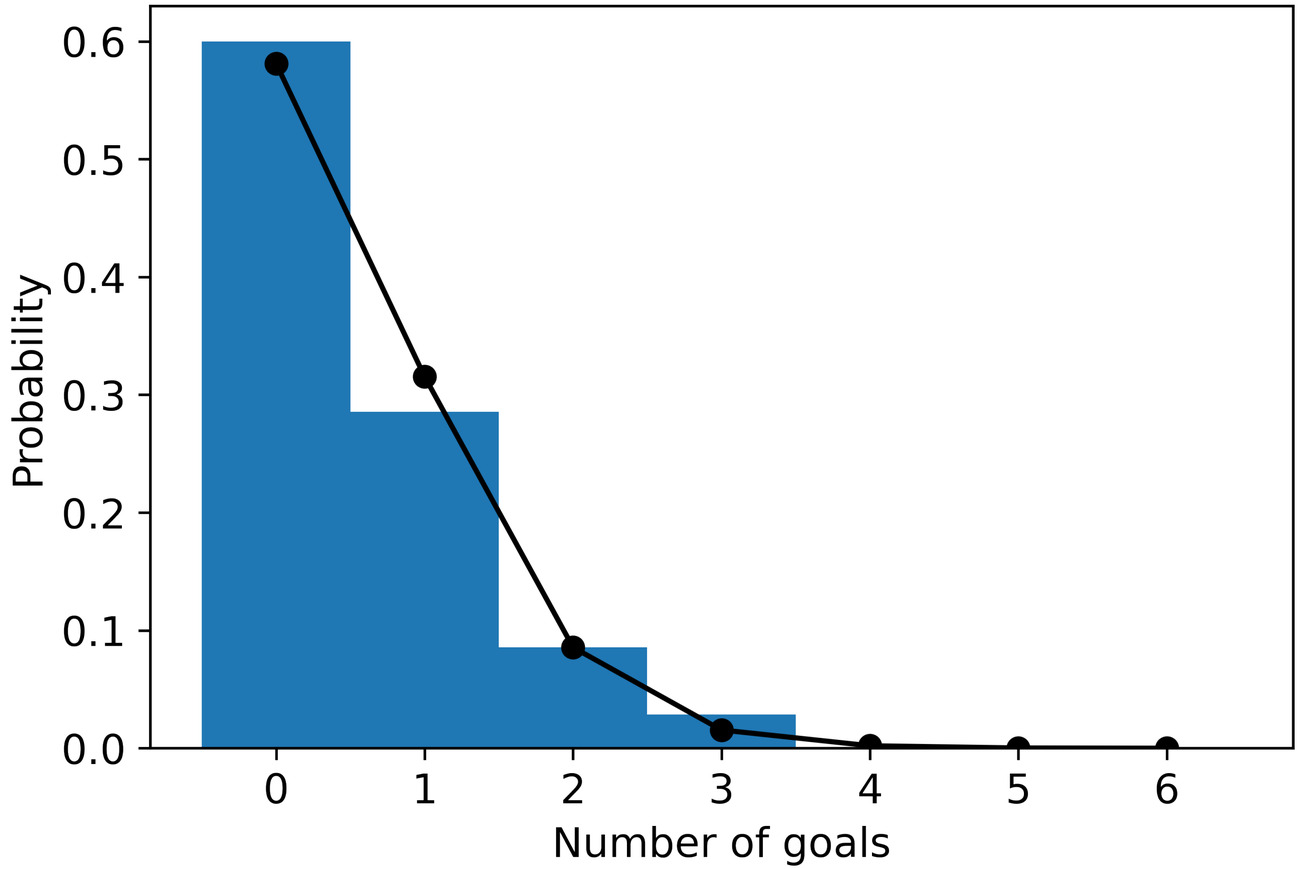
The data is:

[0,0,1,1,1,2,0,0,0,0,2,0,0,1,0,0,0,0,0,0,1,1,2,0,0,0,1,0,0,3,1,0,0,1,1]

The sample average is equal to:

$$\frac{1}{n}\sum_{i=1}^{n} X_i = 0.54 = \hat{\lambda}$$

Even for two completely different players the modelling holds well.



Least-squares regression

We assume that our model is of the form:

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon} \quad \mathbf{y}, \boldsymbol{\epsilon} \in \mathbb{R}^n, \mathbf{X} \in \mathbb{R}^{n \times p}$$

The least squared error (LSE) estimator minimizes the l_2 norm difference

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{t}} ||\mathbf{y} - \mathbf{X}\mathbf{t}||_2^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Then we can write that $\hat{\mathbf{w}} = \mathbf{w} + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{\epsilon}$ and if $\boldsymbol{\epsilon} \sim \mathcal{N}_n(0, \sigma^2\mathbf{I})$ then

$$\hat{\mathbf{w}} = \mathbf{w} + \mathcal{N}_p(0, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$$

Least-squares regression

We assume that our model is of the form:

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon} \quad \mathbf{y}, \boldsymbol{\epsilon} \in \mathbb{R}^n, \mathbf{X} \in \mathbb{R}^{n \times p}$$

The least squared error (LSE) estimator minimizes the l_2 norm difference

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{t}} ||\mathbf{y} - \mathbf{X}\mathbf{t}||_2^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Then we can write that $\hat{\mathbf{w}} = \mathbf{w} + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{\epsilon}$ and if $\boldsymbol{\epsilon} \sim \mathcal{N}_n(0, \sigma^2\mathbf{I})$ then

$$\hat{\mathbf{w}} = \mathbf{w} + \mathcal{N}_p(0, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$$

Remarks:

- We can find the estimator if $\mathbf{X}^T\mathbf{X}$ is invertible, which is not the case if $p > n$.
- The estimator is unbiased with variance proportional to $(\mathbf{X}^T\mathbf{X})^{-1}$, meaning that features with high variance are more informative.

MLE of least-squares regression

Since $\mathbf{y} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon} \sim \mathcal{N}_n(0, \sigma^2\mathbf{I})$, the probability density of $\mathbf{y}$ is $\mathcal{N}_n(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$:

$$p(\mathbf{y}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp(-\frac{||\mathbf{y} - \mathbf{X}\mathbf{w}||_2^2}{2\sigma^2})$$

Then the log likelihood is:

$$\log p(\mathbf{y}) = -n/2 \log(2\pi\sigma^2) - \frac{||\mathbf{y} - \mathbf{X}\mathbf{w}||_2^2}{2\sigma^2}$$

so that:

$$\max_t \log p(\mathbf{y}) = \min_t ||\mathbf{y} - \mathbf{X}\mathbf{t}||_2^2$$

and we have that the maximum likelihood estimator is equal to the least squares estimators if we assume that the noise is Gaussian.

Some properties

- LSE = MLE: $\hat{\mathbf{w}} \sim \mathcal{N}_p(\mathbf{w}, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$

- Prediction error: $\mathbb{E}[\,||\mathbf{y} - \mathbf{X}\hat{\mathbf{w}}||_2^2] = \sigma^2(n-p)$

- Estimator of the variance: $\hat{\sigma}^2 = \frac{1}{n-p}||\mathbf{y} - \mathbf{X}\hat{\mathbf{w}}||_2^2$

Some remarks:

- The equality between LSE and MLE holds only in the case of Gaussian noise.
- The prediction error is equal to zero if $n = p$.
- The $n - p$ factor comes from the fact that the noise lives in the space orthogonal to the span of $\mathbf{X}$, which has dimensions $n - p$.

Bayesian statistics

Up until now we have employed a frequentist approach (collect data and infer the distribution parameter using the MLE).

In Bayesian statistic, we have a two step process:

- We assume some distribution on the parameters,
- We update the distribution by collecting data.

Bayesian statistics

Up until now we have employed a frequentist approach (collect data and infer the distribution parameter using the MLE).

In Bayesian statistic, we have a two step process:

- We assume some distribution on the parameters,
- We update the distribution by collecting data.

Bayes theorem:

$$p(A \mid B) = \frac{p(B \mid A)p(A)}{p(B)}$$

so if $p(A)$ is my prior (distribution of the parameters) and $p(B \mid A)$ is the probability of the data conditioned on the parameters, then we obtain $p(A \mid B)$ the probability of the parameters conditioned on the data

Bayesian statistics

A bit more rigorously:

- $\pi(\theta)$ is the prior on the parameter θ,

- $p(X_1, \ldots, X_n \mid \theta)$ is the likelihood, the probability of the data given the parameter θ,

- $p(X_1, \ldots, X_n) = \int_\Theta p(X_1, \ldots, X_n \mid \theta)\pi(\theta)\ d\theta$ is the marginal likelihood, or the probability distribution of the data itself,

- $p(\theta \mid X_1, \ldots, X_n)$ is the posterior distribution.

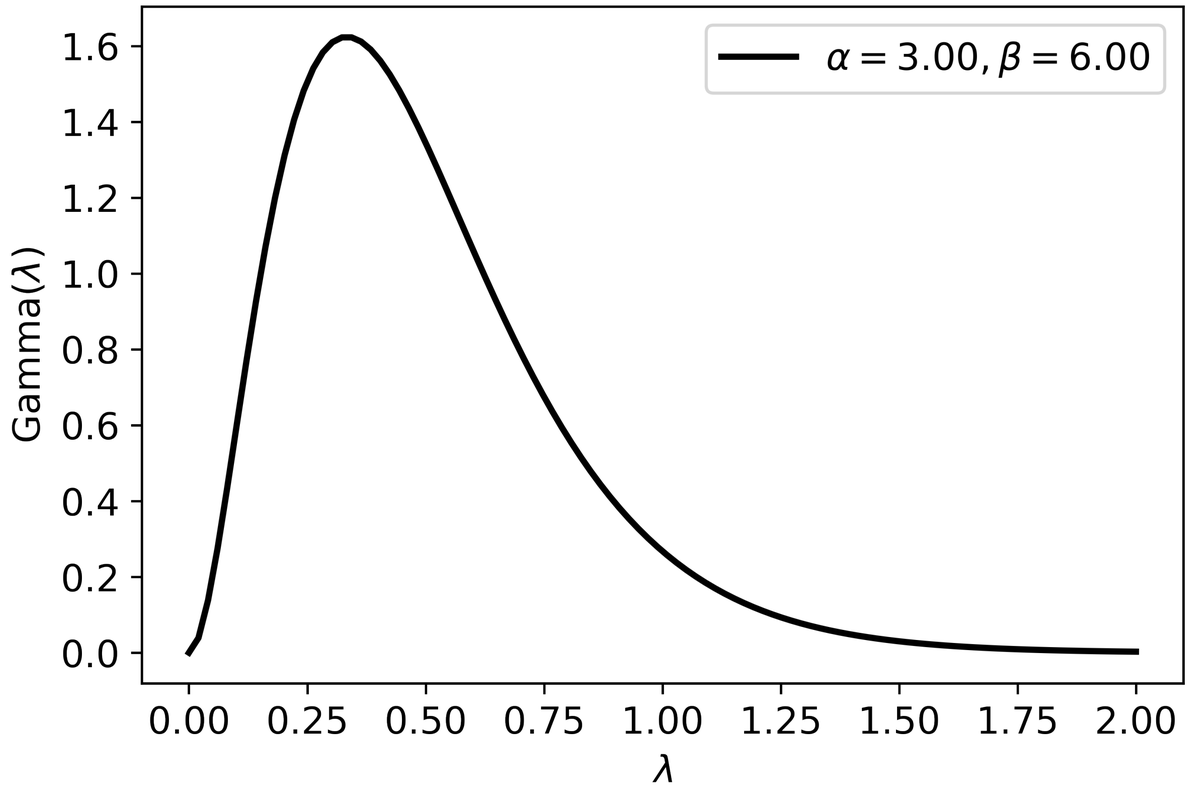
Example on Poisson distribution

Let's consider our example on the distributions of goals, but this time we put a prior distribution on the parameter λ:

- Prior: $\pi(\lambda) = \text{Gamma}(\lambda, \alpha, \beta) = \dfrac{\beta^{\alpha}\lambda^{\alpha-1}e^{-\beta\lambda}}{\Gamma(\alpha)}$
- Likelihood: $p(X_1, \ldots, X_n \mid \lambda) = \dfrac{\lambda^{\sum_{i=1}^{n} X_i} e^{-n\lambda}}{X_1! \cdots X_n!}$
- Marginal likelihood: $p(X_1, \ldots, X_n) = \displaystyle\int_{\Theta} \frac{\lambda^{\sum_{i=1}^{n} X_i} e^{-n\lambda}}{X_1! \cdots X_n!} \frac{\beta^{\alpha}\lambda^{\alpha-1}e^{-\beta\lambda}}{\Gamma(\alpha)}\, d\lambda$
- Posterior: $p(\lambda \mid X_1, \ldots, X_n) = \text{Gamma}(\lambda, \alpha + \displaystyle\sum_{i=1}^{n} X_i, \beta + n)$

Example on Poisson distribution

Prior: $\pi(\lambda) = \text{Gamma}(\lambda, \alpha = 3, \beta = 6)$

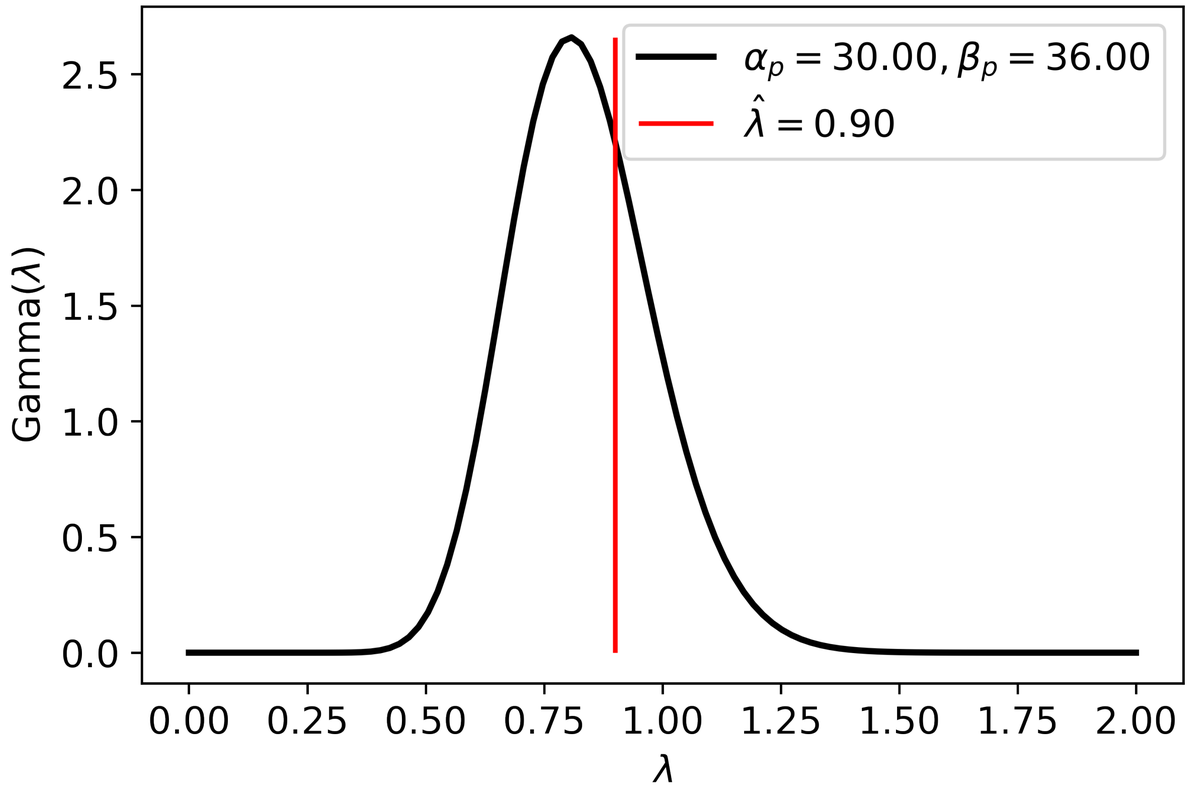


Example on Poisson distribution

Prior: $\pi(\lambda) = \text{Gamma}(\lambda, \alpha = 3,\ \beta = 6)$

Posterior:

$p(\lambda \mid X_1, \ldots, X_n) = \text{Gamma}(\lambda, \alpha = 30,\ \beta = 36)$



Example on Poisson distribution

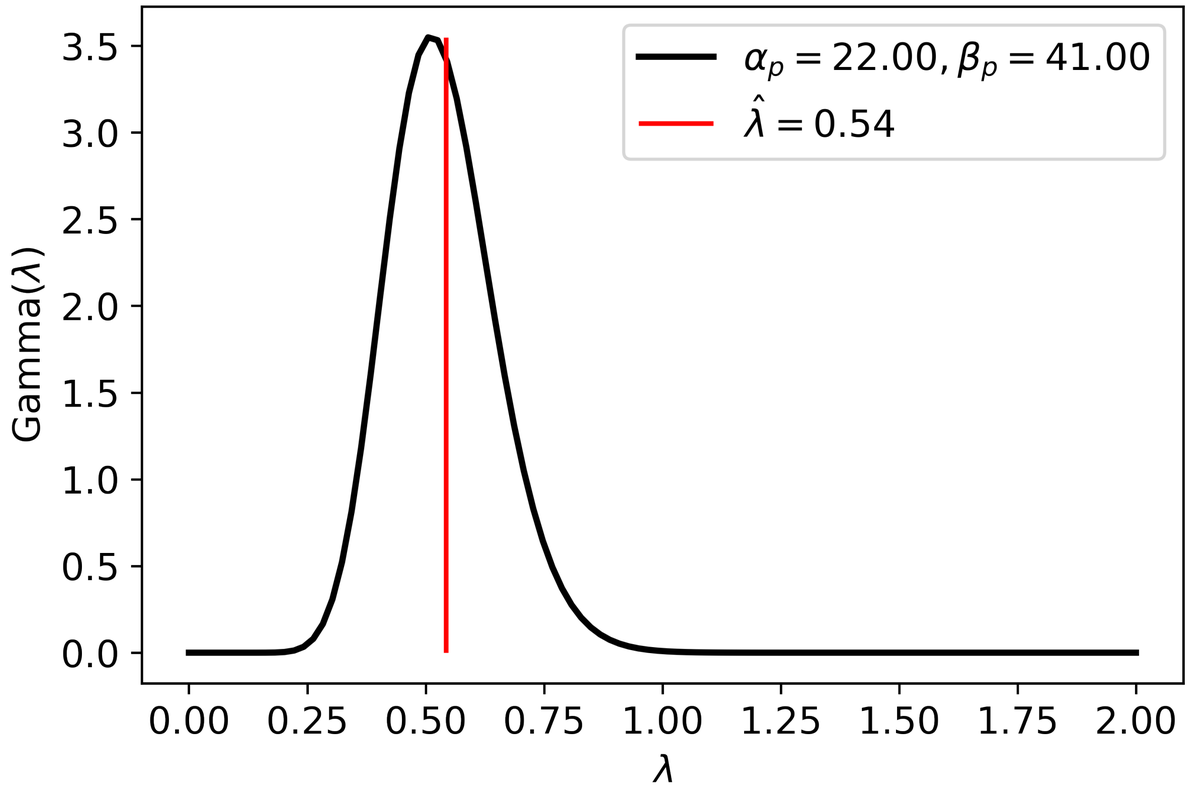
Prior: $\pi(\lambda) = \text{Gamma}(\lambda, \alpha = 3,\ \beta = 6)$

Posterior:

$p(\lambda \mid X_1, \ldots, X_n) = \text{Gamma}(\lambda, \alpha = 30,\ \beta = 36)$

Posterior:

$p(\lambda \mid X_1, \ldots, X_n) = \text{Gamma}(\lambda, \alpha = 22,\ \beta = 41)$

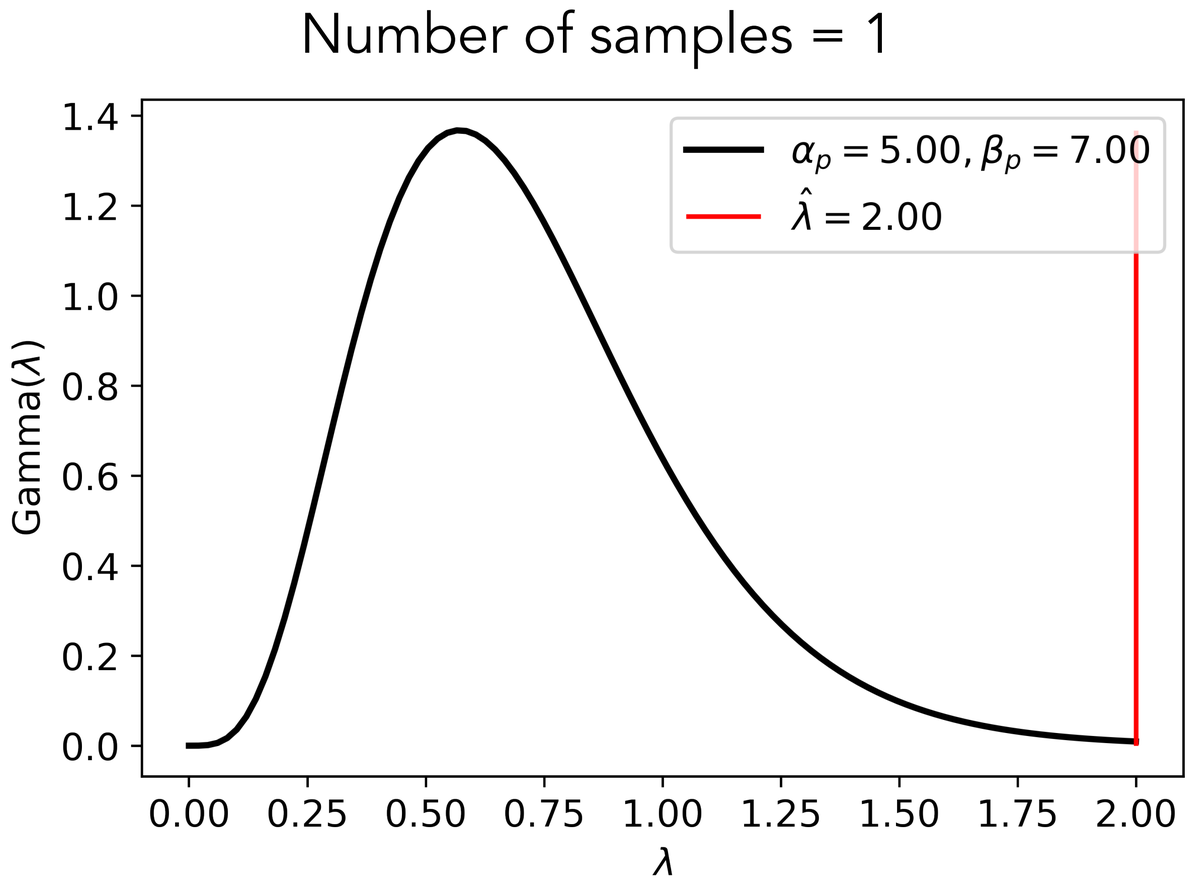


Convergence to the mean

Putting a prior on the distribution parameters let us be more conservative with the data that we need to estimate the parameters.

In general, Bayesian methods are more efficient (if we use a good prior).

The prior is also a good place where we can put physical information about our process.

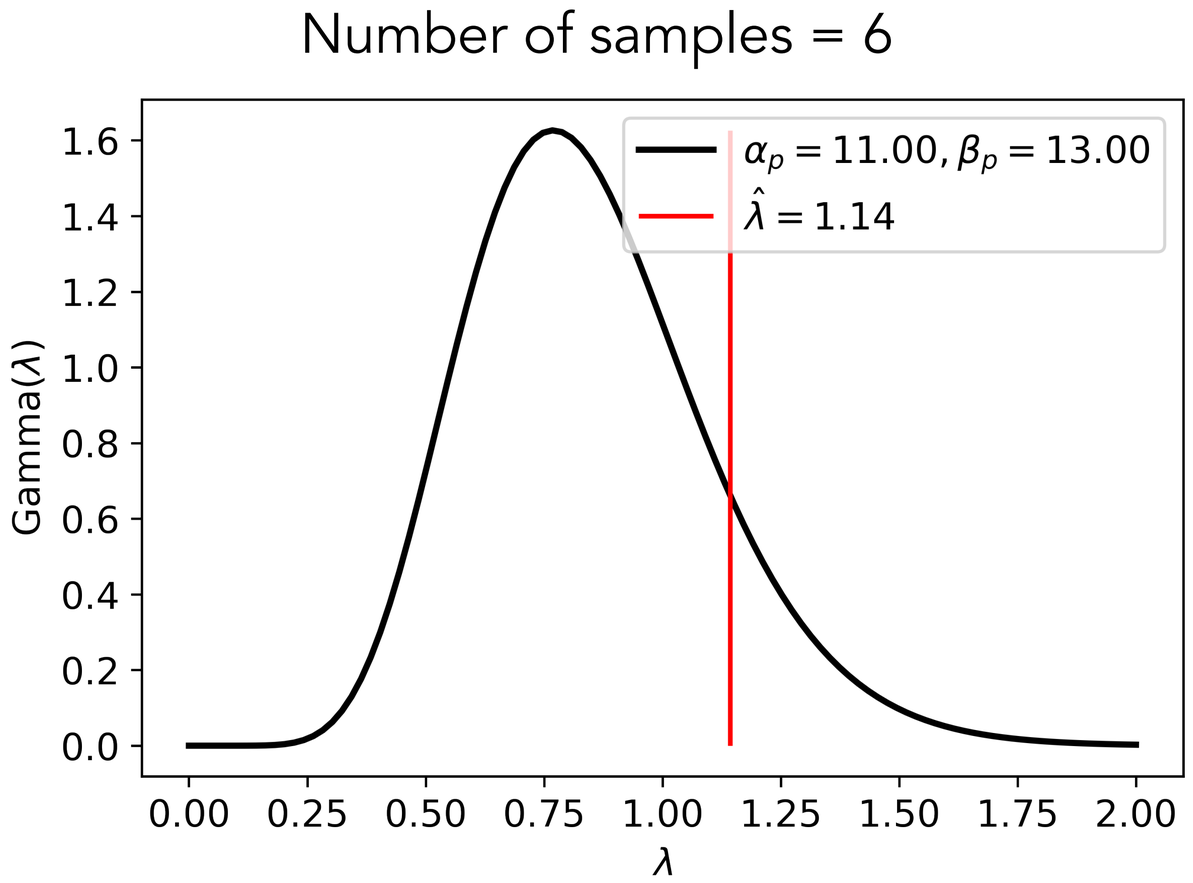


Convergence to the mean

Putting a prior on the distribution parameters let us be more conservative with the data that we need to estimate the parameters.

In general, Bayesian methods are more efficient (if we use a good prior).

The prior is also a good place where we can put physical information about our process.

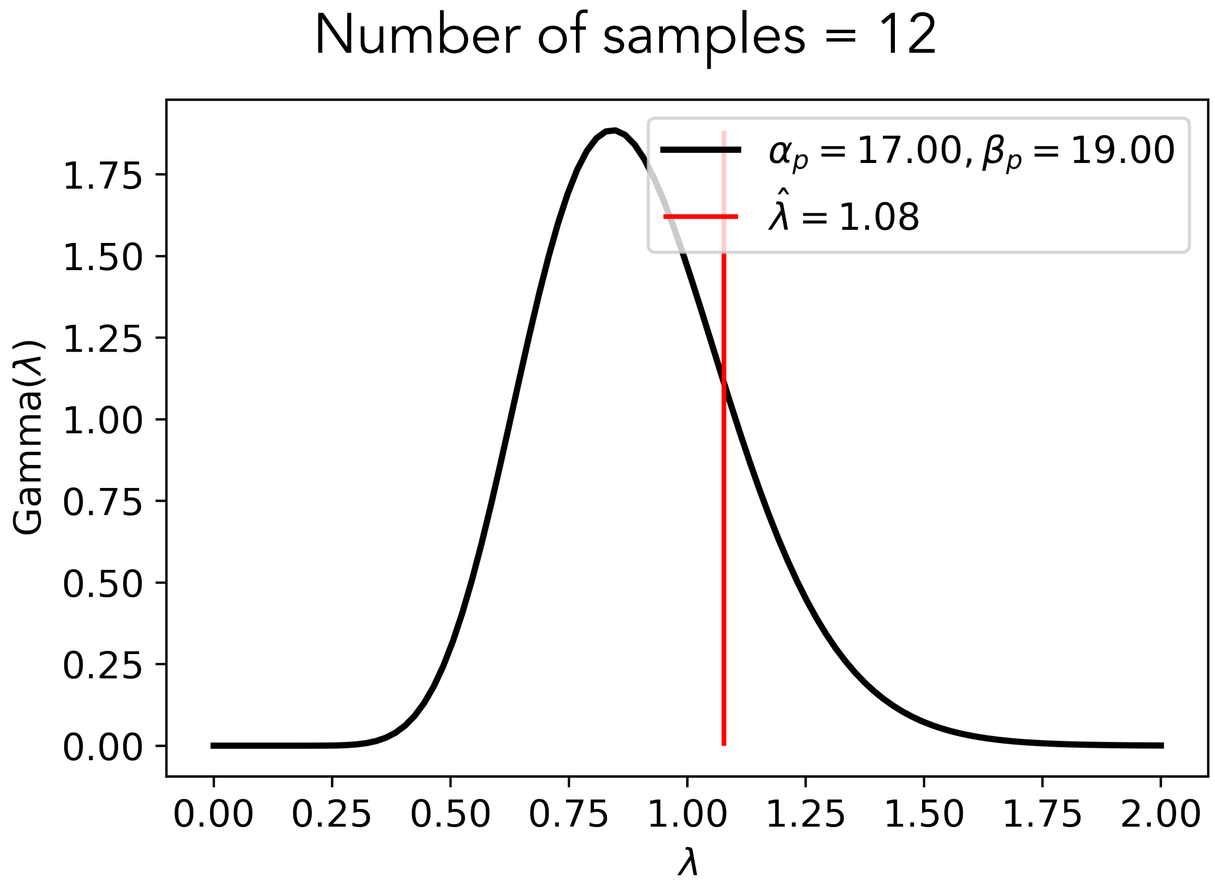


Convergence to the mean

Putting a prior on the distribution parameters let us be more conservative with the data that we need to estimate the parameters.

In general, Bayesian methods are more efficient (if we use a good prior).

The prior is also a good place where we can put physical information about our process.

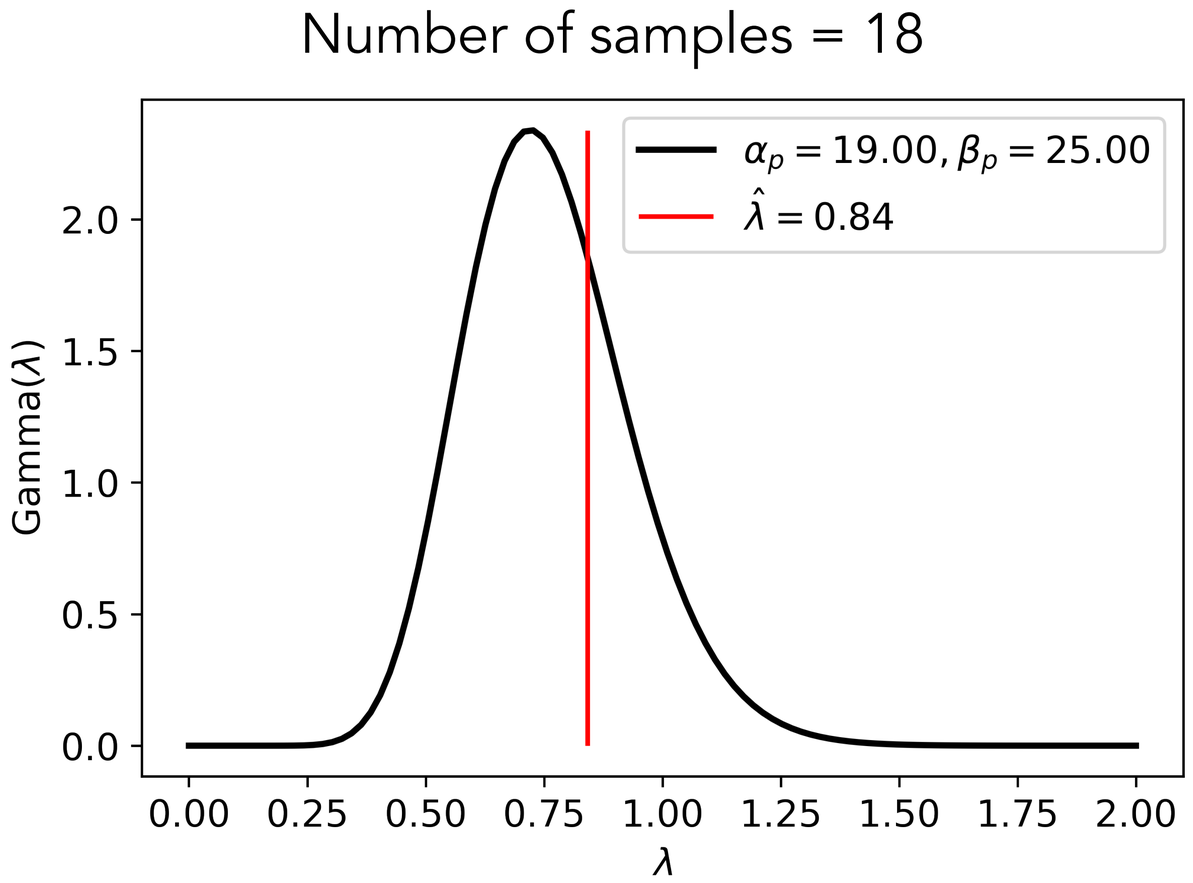


Convergence to the mean

Putting a prior on the distribution parameters let us be more conservative with the data that we need to estimate the parameters.

In general, Bayesian methods are more efficient (if we use a good prior).

The prior is also a good place where we can put physical information about our process.

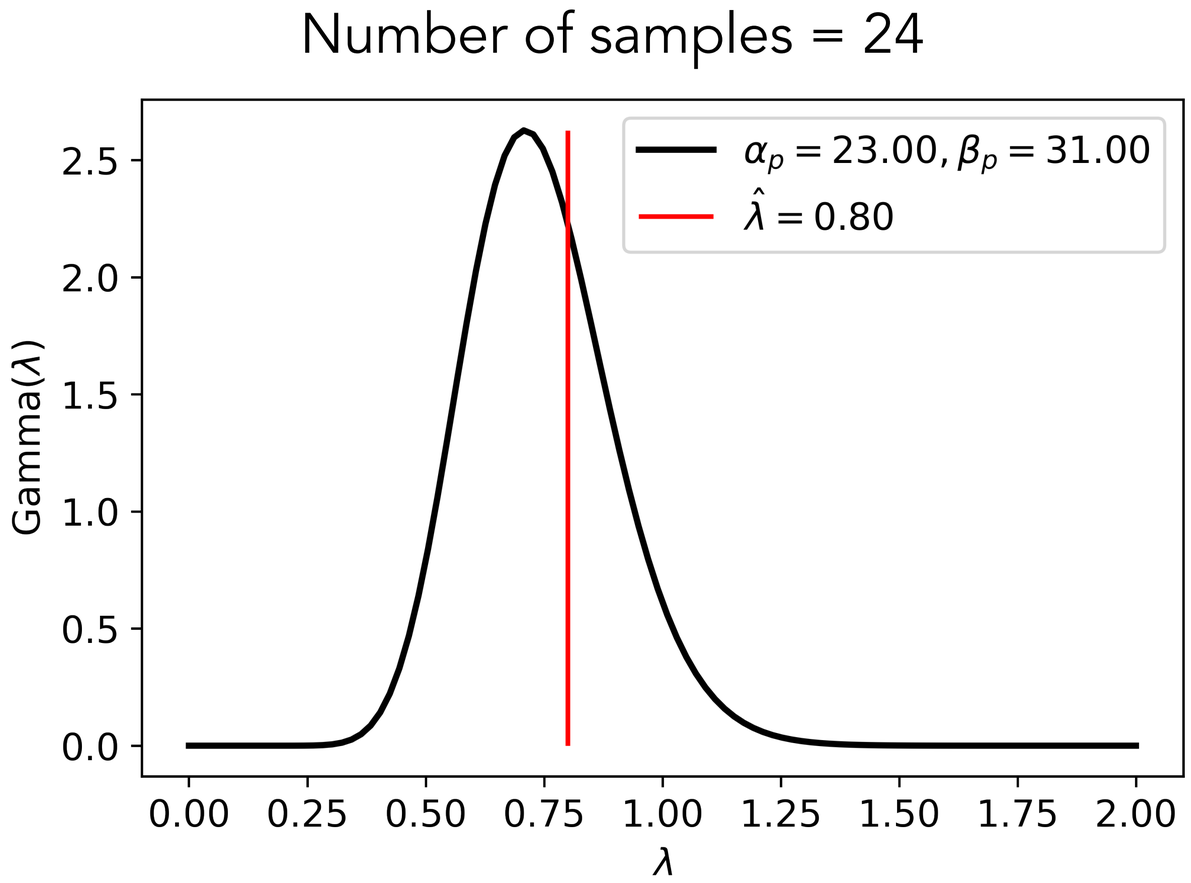


Convergence to the mean

Putting a prior on the distribution parameters let us be more conservative with the data that we need to estimate the parameters.

In general, Bayesian methods are more efficient (if we use a good prior).

The prior is also a good place where we can put physical information about our process.



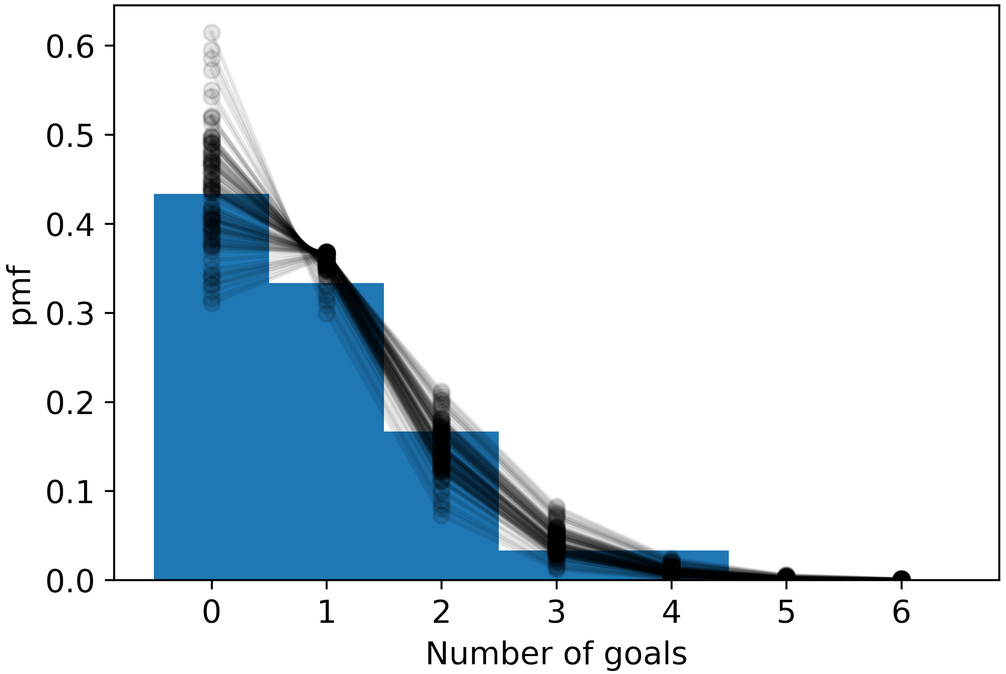
Distribution of distributions

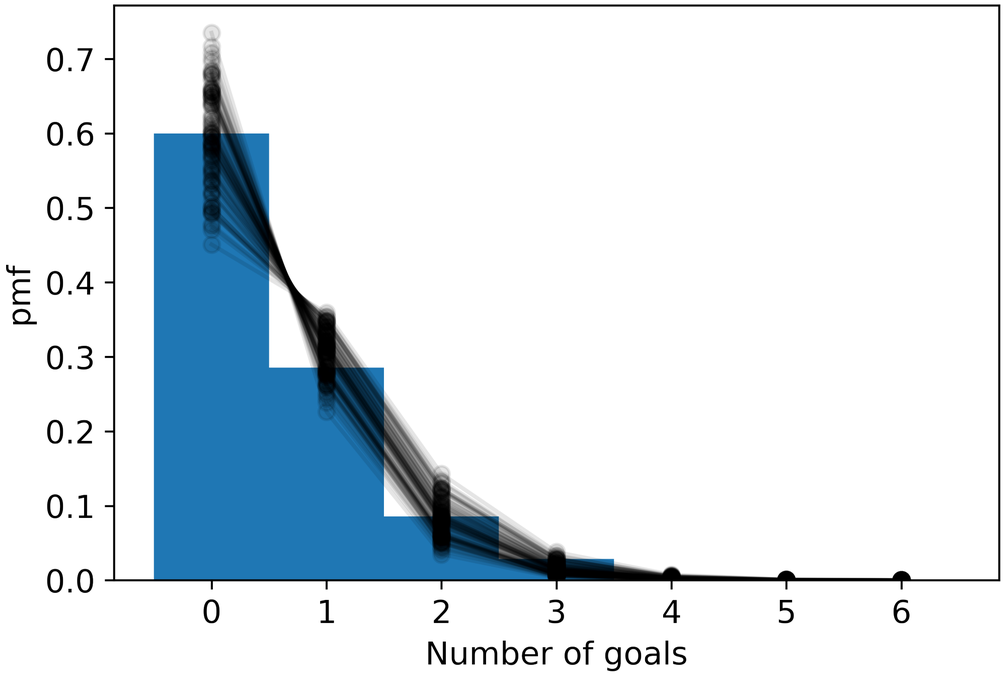
Now that we have a posterior distribution on λ, we can sample from it:

$$\lambda_i \sim p(\lambda \mid X_1, \ldots, X_n)$$

$$p(x, \lambda_i) = \frac{\lambda^x e^{-\lambda_i}}{x!} \quad i = 1, \ldots, m$$

This information is more powerful than knowing only the MLE, because it provides a distribution of distributions.





If we do not have any idea

$$p(\theta \mid X_1, \ldots, X_n) = \frac{p(X_1, \ldots, X_n \mid \theta)\pi(\theta)}{\int_\Theta p(X_1, \ldots, X_n \mid \theta)\pi(\theta)\ d\theta}$$

The equation still works, we just assume that $\pi(\theta)$ is equal to one.

This way we do not put any weights on θ (non informative prior) but we can still obtain a posterior distribution which can give us an idea of the uncertainty estimation.

Technically, this prior is not a true distribution (improper prior), but we can still use it to find the posterior as the ratio between the likelihood and its integration.

Gaussian model

$$p(\theta \mid X_1, \ldots, X_n) = \frac{p(X_1, \ldots, X_n \mid \theta)\pi(\theta)}{\int_\Theta p(X_1, \ldots, X_n \mid \theta)\pi(\theta)\ d\theta}$$

For example, the posterior of a Gaussian likelihood with uninformative prior:

$X_1, \ldots, X_n \sim \mathcal{N}(\theta, 1)$

$\pi(\theta) = 1$

$$p(X_1, \ldots, X_n \mid \theta) = \frac{1}{\sqrt{2\pi}^n}\ exp(-\frac{1}{2}\sum_{i=1}^{n}(X_i - \theta)^2)$$

$$p(\theta, X_1, \ldots, X_n) = \mathcal{N}(\bar{X}_n, \frac{1}{n})$$

Bayesian treatment of OLS

Let's consider again the linear regression model with Gaussian noise:

$$y = \mathbf{w}^T\mathbf{x} + \epsilon \quad \epsilon \sim \mathcal{N}(0,\, \sigma^2)$$

We already know what is the likelihood of this model:

$$p(\mathbf{y}\,|\,\mathbf{X}, \mathbf{w}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp(-\frac{||\,\mathbf{Y} - \mathbf{X}\mathbf{w}\,||_2^2}{2\sigma^2}) = \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

We assume that the prior is a Gaussian distribution $p(\mathbf{w}\,|\,\mathbf{X}) = \mathcal{N}(0,\mathbf{\Sigma}_p)$

Bayesian treatment of OLS

Let's consider again the linear regression model with Gaussian noise:

$$y = \mathbf{w}^T\mathbf{x} + \epsilon \quad \epsilon \sim \mathcal{N}(0,\, \sigma^2)$$

We already know what is the likelihood of this model:

$$p(\mathbf{y}\,|\,\mathbf{X}, \mathbf{w}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp(-\frac{||\,\mathbf{Y} - \mathbf{X}\mathbf{w}\,||_2^2}{2\sigma^2}) = \mathcal{N}(\mathbf{X}\mathbf{w}, \sigma^2\mathbf{I})$$

We assume that the prior is a Gaussian distribution $p(\mathbf{w}\,|\,\mathbf{X}) = \mathcal{N}(0,\boldsymbol{\Sigma}_p)$

Applying Bayes we obtain that the posterior is equal to:

$$p(\mathbf{w}\,|\,\mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y}\,|\,\mathbf{X}, \mathbf{w})p(\mathbf{w}\,|\,\mathbf{X})}{p(\mathbf{y}\,|\,\mathbf{X})}$$

Turns out that $p(\mathbf{w}\,|\,\mathbf{y}, \mathbf{X}) = \mathcal{N}(\bar{\mathbf{w}}, \mathrm{cov}(\mathbf{w}))$ with

$$\bar{\mathbf{w}} = \frac{1}{\sigma^2}(\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} + \boldsymbol{\Sigma}_p^{-1})^{-1}\mathbf{X}^T\mathbf{y}$$

$$\mathrm{cov}(\mathbf{w}) = (\frac{1}{\sigma^2}\mathbf{X}^T\mathbf{X} + \boldsymbol{\Sigma}_p^{-1})^{-1}$$

Gaussian process regression

In the case of linear regression we have assumed that we can model the observed data as a sum of a deterministic part ($\mathbf{Xw}$) plus a random part ($\boldsymbol{\epsilon}$).

In GPR instead we directly assume that the entire function is a sample from a Gaussian process:

$$f(x) \sim GP(\mu(x), k(x, x'))$$

Similar to the normal distribution, a Gaussian process is a distribution over function, which is parametrized by a mean function $\mu(x)$ and a covariance (or kernel) function $k(x, x')$.

In practise, we approximate functions with arrays, and so we approximate the Gaussian process with a multivariate normal distribution:

$$\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K})$$

GPR is a Bayesian method

We collect n observations $\{\mathbf{x}_i, y_i\}_n$ and we assemble them into the input matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ and the array of observations $\mathbf{y} \in \mathbb{R}^n$.

We start by selecting our prior, which is usually a normal distribution with zero mean:

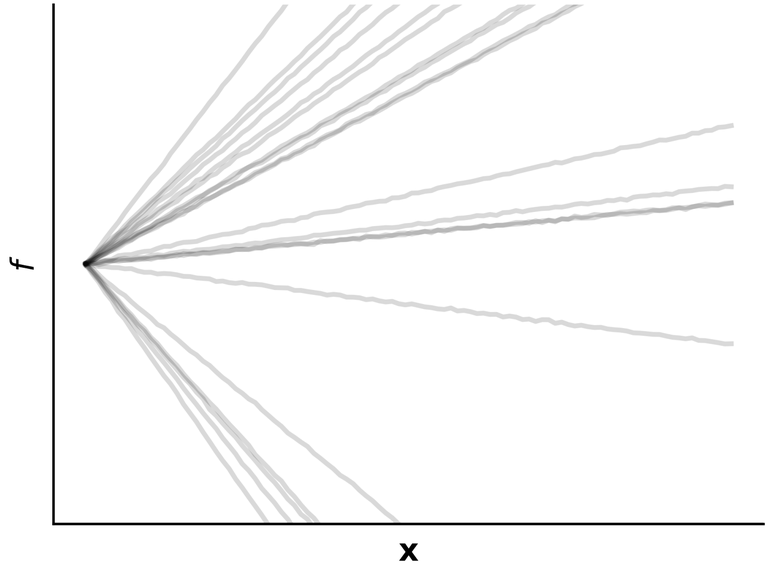
$$p(\mathbf{f} \,|\, \mathbf{X}, \phi) = \mathcal{N}(\mathbf{X}; 0, \mathbf{K}(\phi))$$

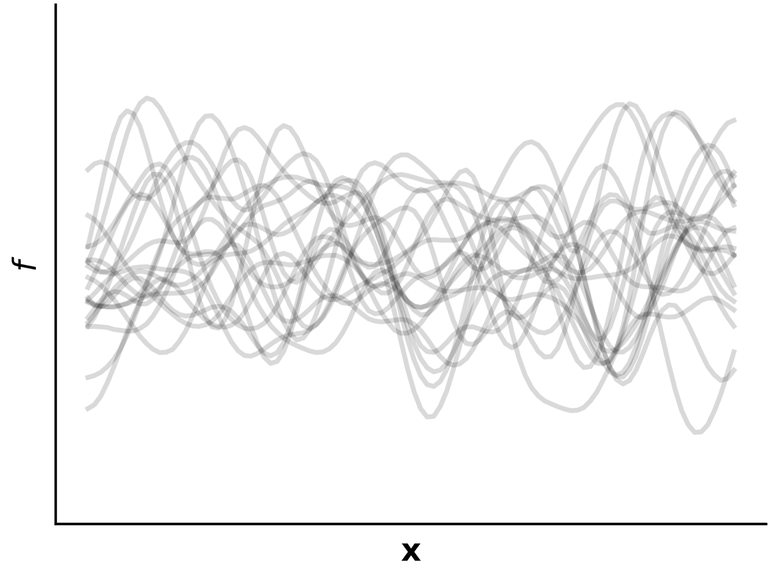
To compute the covariance matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$, we use a kernel function $\mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j; \phi)$.
The kernel function measures the similarity between points and it is controlled by the hyperparameter ϕ.
The simplest kernel function is the linear kernel, which assumes that the similarity between points decrease linearly with the distance:

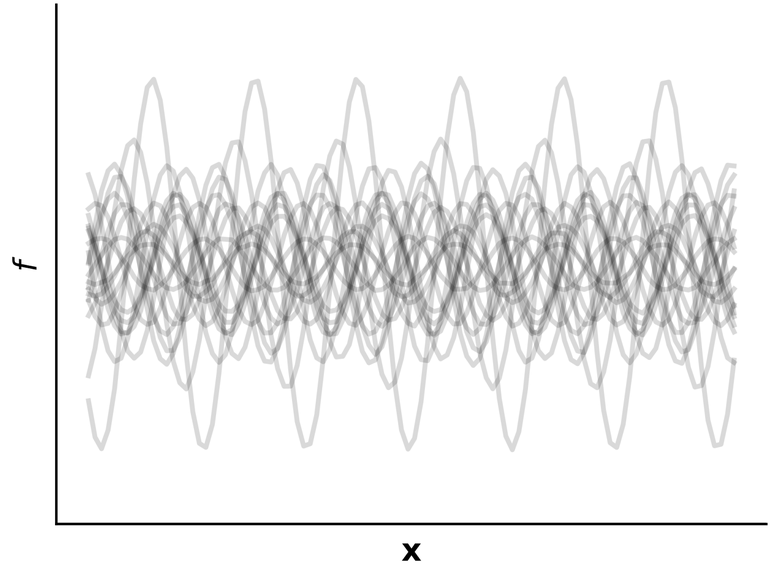
$$k_l(\mathbf{x}_i, \mathbf{x}_j; \phi) = \phi \mathbf{x}_i^T \mathbf{x}_j$$

Some kernels

Let's plot some samples from the prior with different kernels:







$$k_l(\mathbf{x}_i, \mathbf{x}_j; \phi) = \phi \mathbf{x}_i^T \mathbf{x}_j$$

$$k_c(\mathbf{x}_i, \mathbf{x}_j; \phi) = \cos(\pi ||\mathbf{x}_i - \mathbf{x}_j||_2 / \phi)$$

$$k_{RBF}(\mathbf{x}_i, \mathbf{x}_j; \phi) = \exp(-\frac{(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j)}{2\phi^2})$$

How to chose the kernel

The choice of the kernels depends on the type of regression problem.

In general, the RBF is a good choice because it is very flexible and it can adapt to different types of models.

The kernel can be used also to include physical information on the system. For example, if we know that the system is periodic we can use a mixture of cosine kernels.

A kernel can be also created by composing different kernel functions (adding a linear kernel with a periodic kernel for example).

Compute the posterior

We want to compute the posterior $p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}, \phi)$

So we need:

- the likelihood $p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi)$
- the marginal likelihood $p(\mathbf{y} \mid \mathbf{X}, \phi) = \int p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi) p(\mathbf{f} \mid \mathbf{X}, \phi) d\mathbf{f}$

Compute the posterior

We want to compute the posterior $p(\mathbf{f} \mid \mathbf{y}, \mathbf{X}, \phi)$

So we need:

- the likelihood $p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi)$
- the marginal likelihood $p(\mathbf{y} \mid \mathbf{X}, \phi) = \int p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi) p(\mathbf{f} \mid \mathbf{X}, \phi) d\mathbf{f}$

The likelihood maps the model to the observation, i.e. is the noise:

$$p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi) = \mathcal{N}(\mathbf{f}, \sigma^2 \mathbf{I})$$

The marginal likelihood represents the probability of observing the data given my choice of kernel hyperparameters.

The marginal likelihood

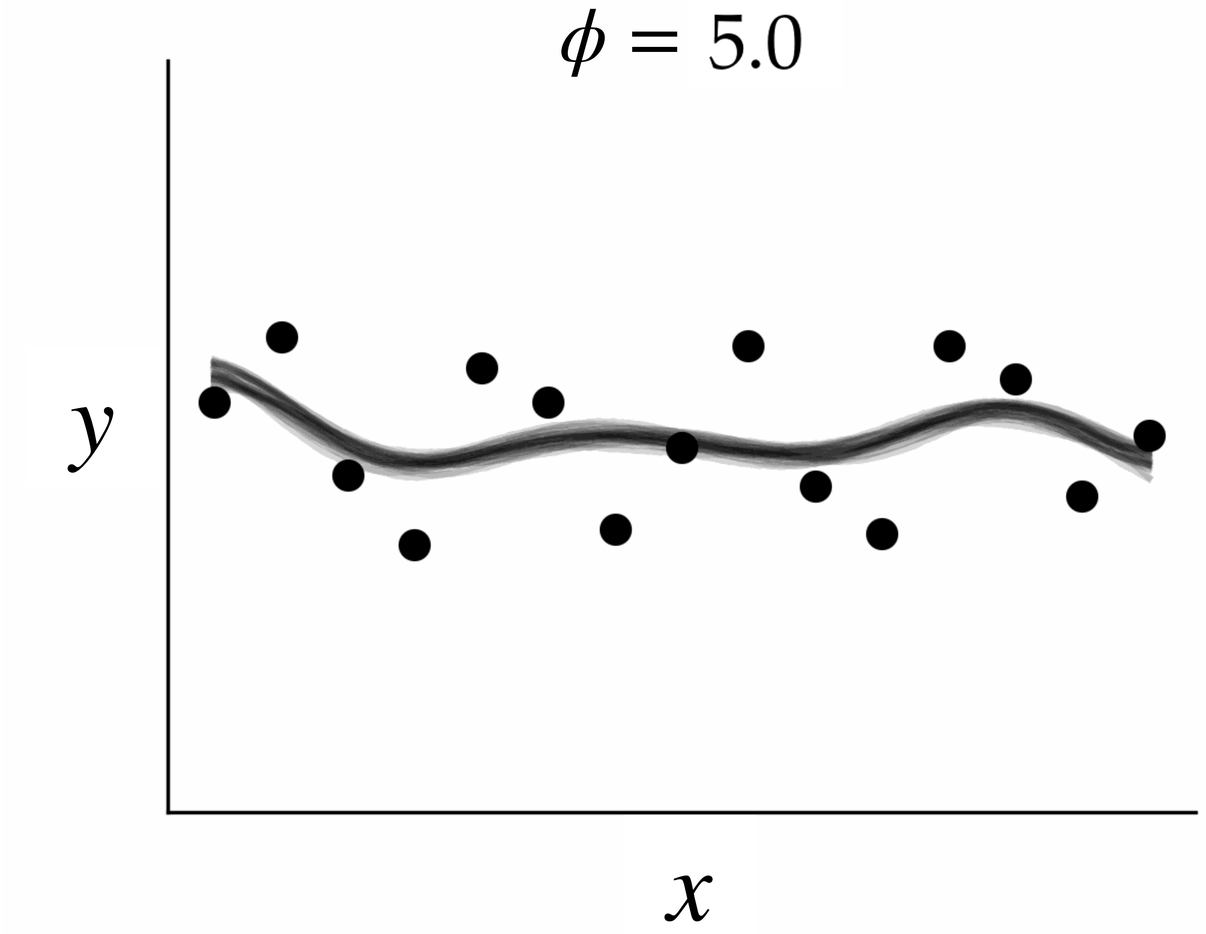
In general, it is very hard to compute directly the marginal likelihood.
However, if both the likelihood and the prior are Gaussian, then the marginal likelihood has a closed-form solution:

$$\log p(\mathbf{y} \mid \mathbf{X}, \phi) = \log \int p(\mathbf{y} \mid \mathbf{f}, \mathbf{X}, \phi) p(\mathbf{f} \mid \mathbf{X}, \phi) d\mathbf{f} = -\frac{1}{2}\mathbf{y}^T(\mathbf{K}(\phi) + \sigma^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}(\phi) + \sigma^2\mathbf{I}| - n/2\log 2\pi$$

To find the optimal kernel parameters, we want to maximize $\log p(\mathbf{y} \mid \mathbf{X}, \phi)$, which is equivalent to minimize $-\log p(\mathbf{y} \mid \mathbf{X}, \phi)$.

The negative log marginal likelihood is not a convex function, and so we need to employ methods (such as gradient descent) to find its minimum.

The marginal likelihood



$$\log p(\mathbf{y} \mid \mathbf{X}, \phi) = -\frac{1}{2}\mathbf{y}^T(\mathbf{K}(\phi) + \sigma^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K}(\phi) + \sigma^2\mathbf{I}| - n/2\log 2\pi = \text{-24.14}$$

The posterior

$$p(\mathbf{f}\,|\,\mathbf{y},\mathbf{X},\phi^o) = \frac{p(\mathbf{y}\,|\,\mathbf{f},\mathbf{X},\phi^o)p(\mathbf{f}\,|\,\mathbf{X},\phi^o)}{p(\mathbf{y}\,|\,\mathbf{X},\phi^o)}$$

Using Bayes we obtain the posterior on the training data.

To obtain the posterior on the prediction points $\mathbf{X}^*$(predictive posterior) we need to condition the joint probability between $\mathbf{f}^*$ and $\mathbf{y}$:

Joint probability $p(\mathbf{f}^*,\mathbf{y}\,|\,\mathbf{X},\mathbf{X}^*,\phi^o) = \mathcal{N}(0,\begin{bmatrix}\mathbf{K}+\sigma^2\mathbf{I} & \mathbf{K}_* \\ \mathbf{K}_* & \mathbf{K}_{**}\end{bmatrix})$

Predictive posterior $p(\mathbf{f}^*\,|\,\mathbf{y},\mathbf{X},\mathbf{X}^*,\phi^o) = \frac{p(\mathbf{f}^*\,|\,\mathbf{y},\mathbf{X},\mathbf{X}^*,\phi^o)}{p(\mathbf{y}\,|\,\mathbf{X},\mathbf{X}^*,\phi^o)} = \mathcal{N}(\bar{\mathbf{f}}^*,\mathrm{cov}(\mathbf{f}^*))$

$\bar{\mathbf{f}}^* = \mathbf{K}_*(\mathbf{K}+\sigma^2\mathbf{I})^{-1}\mathbf{y}$

$\mathrm{cov}(\mathbf{f}^*) = \mathbf{K}_{**} - \mathbf{K}_*(\mathbf{K}+\sigma^2\mathbf{I})^{-1}\mathbf{K}_*$

An example

We generate data from the model $y = x\sin(x) + \mathcal{N}(0,1) \quad x \in [\pi, 3\pi]$

Since the function to regress is nonlinear, we opt for the RBF kernel with a scaling parameter:

$$k_{RBF}(\mathbf{x}_i, \mathbf{x}_j; \phi) = s\exp\left(-\frac{(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j)}{2\phi^2}\right)$$

We train the model by minimizing the negative log marginal likelihood to find the set of optimal parameters $[s, \phi, \sigma] = [37.2,\ 1.53,\ 0.9]$.