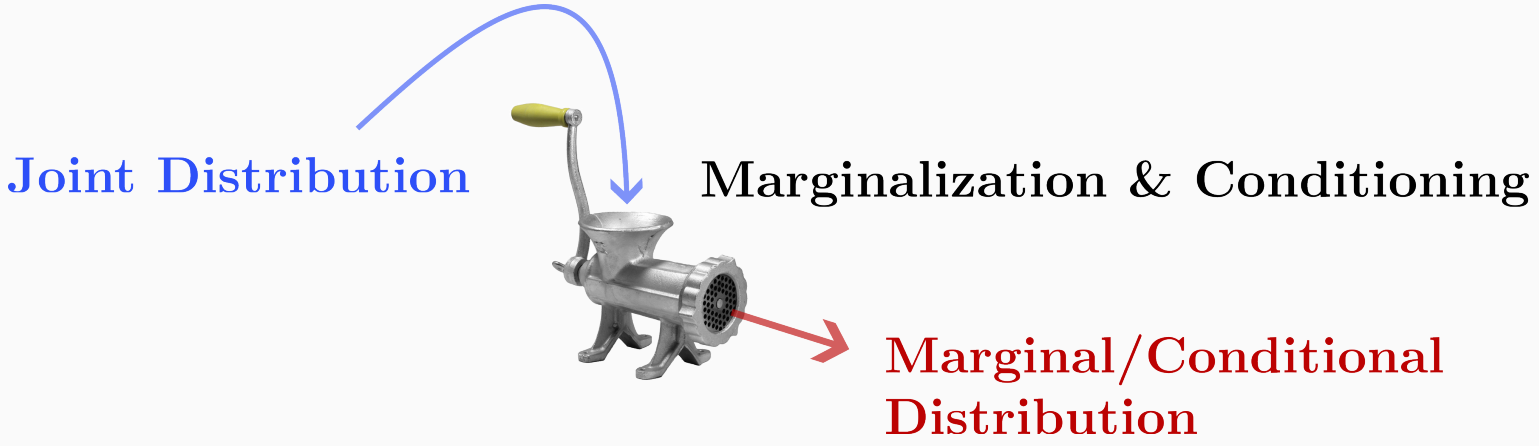
Learning Probabilistic Models (Basics)

Probabilistic Decision Making — Lecture 5

29th October 2025

Robert Peharz
Institute of Machine Learning and Neural Computation
Graz University of Technology



- probability is really about reasoning under uncertainty
- **joint distribution = knowledge base + uncertainty**
- **two inference routines to process our knowledge:**
  - **marginalization** (ignore, account for unknowns)

$$p(\boldsymbol{y}) = \int p(\boldsymbol{x}, \boldsymbol{y}) \, \mathrm{d}\boldsymbol{x} \qquad \text{(sum for discrete)}$$

  - **conditioning** (inject information, update information)

$$p(\boldsymbol{y} \mid \boldsymbol{x}) = \frac{p(\boldsymbol{y}, \boldsymbol{x})}{p(\boldsymbol{x})}$$

- let $\boldsymbol{X}$ be a (univariate or multivariate) RV
- let f be some function defined on the state space of $\boldsymbol{X}$
- let $\boldsymbol{X}_1, \boldsymbol{X}_2, \boldsymbol{X}_3, \ldots$ be iid samples from $p_{\boldsymbol{X}}$

The **Monte Carlo estimator**

$$\hat{\mathbb{E}}_N := \frac{1}{N} \sum_{k=1}^{N} f(\boldsymbol{X}_k)$$

converges with probability 1 to the **exact** expectation:

$$\hat{\mathbb{E}}_N \xrightarrow[N \to \infty]{} \mathbb{E}[f(\boldsymbol{X})] = \int p(\boldsymbol{x})\, f(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}$$

This is called the **law of large numbers**.

Monte Carlo Estimator cont'd

- the Monte Carlo estimator $\hat{\mathbb{E}}_N$ is an average of RVs
- **hence, for finite N, it is an RV itself**
- for any N its expectations is

$$\mathbb{E}\left[\hat{\mathbb{E}}_N\right] = \mathbb{E}[f(\boldsymbol{X})]$$

- an estimator with this property is called **unbiased estimator**
- if $\mathrm{var}[f(\boldsymbol{X})]$ is finite:

$$\mathrm{var}\left[\hat{\mathbb{E}}_N\right] = \frac{\mathrm{var}[f(\boldsymbol{X})]}{N} \xrightarrow[N\to\infty]{} 0$$

- interpretation of expectations as **long term averages**

- let's demonstrate the convergence behavior of Monte Carlo
- let X be Gaussian with $\mu = 1$, $\sigma = 2$; hence

$$\mathbb{E}[X] = \mu = 1, \quad \text{var}[X] = \sigma^2 = 4$$

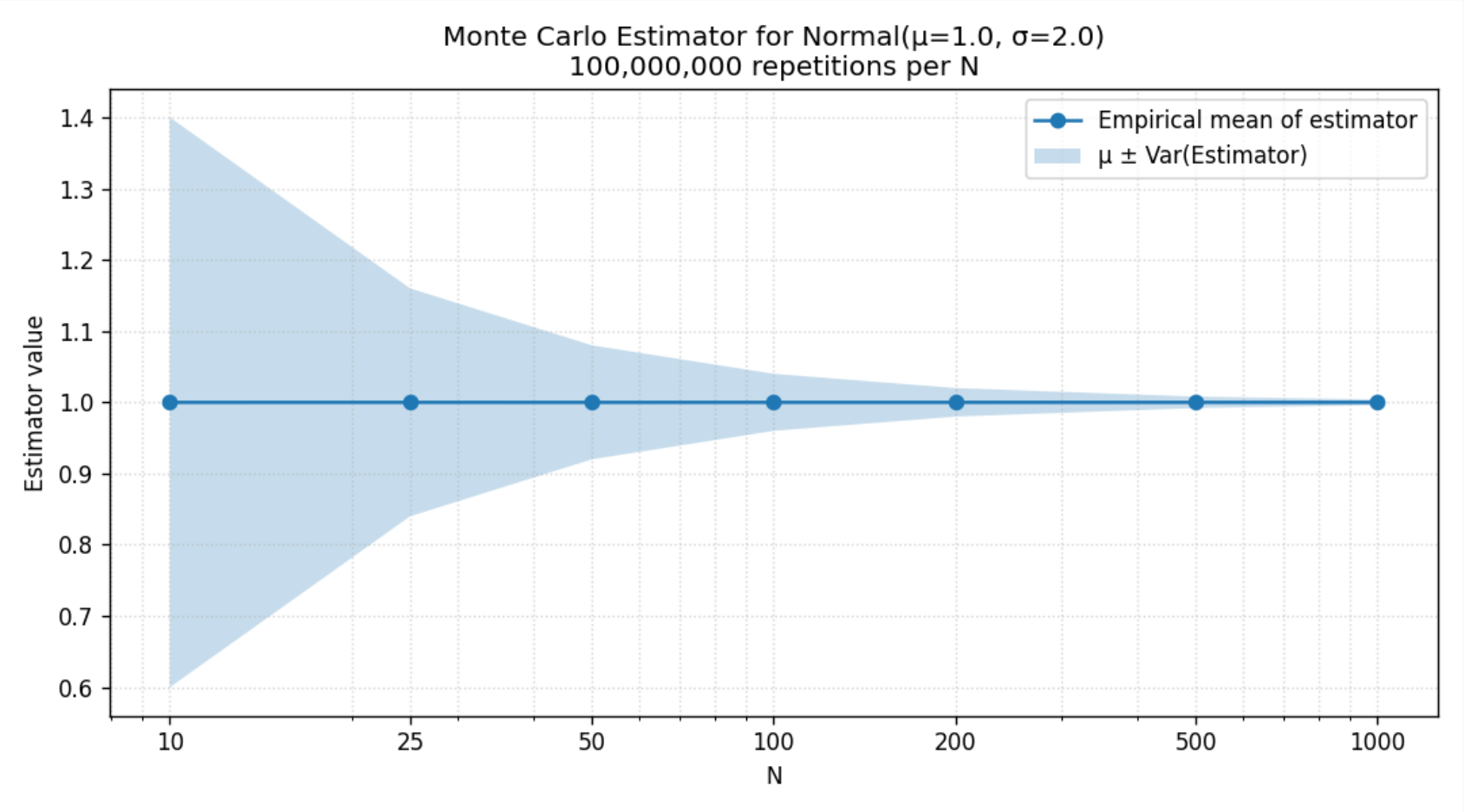
- draw N iid samples $X_1, \ldots, X_N$ and compute

$$\hat{\mathbb{E}}_N = \frac{1}{N} \sum_{k=1}^{N} X_k \approx \mathbb{E}[X] = \mu$$

- for each N, repeat this procedure **100 million times** to compute high-fidelity MC estimates for $\mathbb{E}\left[\hat{\mathbb{E}}_N\right]$ and $\text{var}\left[\hat{\mathbb{E}}_N\right]$ (i.e., MC-Estimators of mean and variance of the MC-Estimator)

Monte Carlo Estimator I cont'd

Example



Note that the mean of the estimator is μ for all N and that the tube plot shrinks as σ^2/N.

- **plot twist: probabilities are also expectations!**
- for any random variable $\boldsymbol{X}$

$$\mathbb{P}_{\boldsymbol{X}}(A) = \mathbb{E}[\mathbb{1}[\boldsymbol{X} \in A]]$$

where $\mathbb{1}$ is the **indicator function**:

$$\mathbb{1}[\text{arg}] = \begin{cases} 1 & \text{if arg is True} \\ 0 & \text{if arg is False} \end{cases}$$

Monte Carlo Estimator II

Example

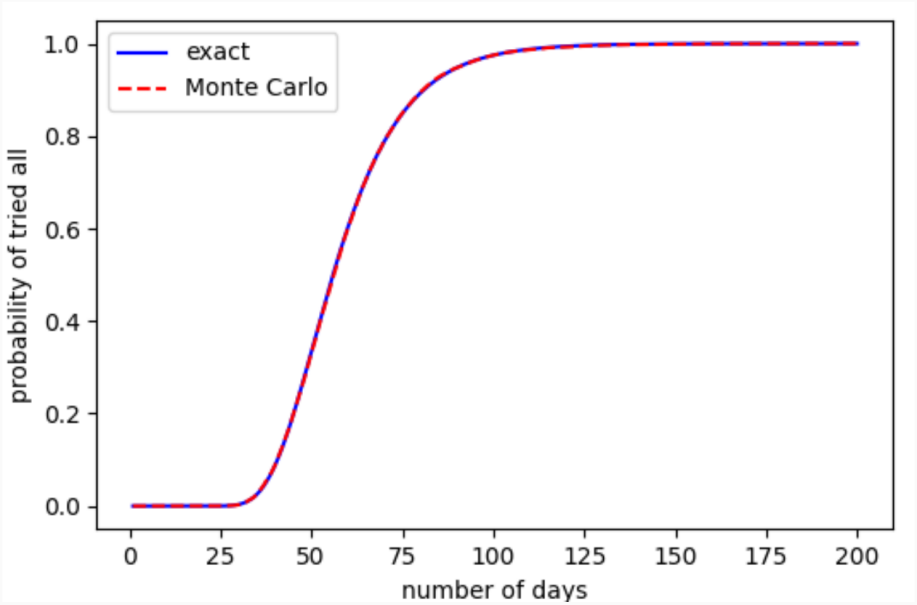
- your favorite ice cream parlor has 42 different ice cream flavors
- every day you go there and ask for 3 random flavors
- what is the probability, as a function of days, that you have tried all flavors?
- you could now invest substantial brain power to find that the exact answer is ...

$$p(d) = \sum_{k=0}^{39} (-1)^k \binom{42}{k} \left(\frac{\binom{42-k}{3}}{\binom{42}{3}} \right)^d$$

- ...or invest 5 minutes to write a Monte Carlo estimator:
  - simulate **trials** over some number of days
  - increase counter for each day when all flavors have been seen
  - normalize by number of trials

Monte Carlo Estimator II

```python
def estimate_icecream(days=200, trials=10000, n=42, r=3):
    hits = np.zeros(days)
    for _ in range(trials):
        seen = set()
        for d in range(days):
            seen.update(random.sample(range(n), r))
            if len(seen) == n:
                hits[d:] += 1
                break
    return hits / trials
```



Maximum Likelihood

Setting

- say we have a dataset $\mathcal{D} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}\}$ drawn **independent and identically distributed** (iid) from some unknown distribution p^*
- further, assume we have a **parametric model class** $\{p_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}}$, i.e., a set of distributions $p_{\boldsymbol{\theta}}$ described by a parameter vector $\boldsymbol{\theta}$
- $\boldsymbol{\Theta}$ is the **parameter space**, i.e. the set of all possible parameters
- how do we set $\boldsymbol{\theta}$ such that

$$p_{\boldsymbol{\theta}} \approx p^* \quad ?$$

Recall, multiple random variables are **independent** if their distribution factorizes.

Hence, due to the **iid assumption**, the probability density of the whole dataset $p(\mathcal{D} \mid \boldsymbol{\theta})$ factorizes into **sample-wise densities**:

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$$

The quantity

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$$

depends on the dataset $\mathcal{D}$ and the parameters $\boldsymbol{\theta}$.

- When interpreted as function of $\mathcal{D}$: probability density of $\mathcal{D}$
- When interpreted as function of $\boldsymbol{\theta}$: **likelihood** of $\boldsymbol{\theta}$

Maximum Likelihood Estimator (MLE): Parameters $\hat{\boldsymbol{\theta}}$ which maximize the likelihood

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \; p(\mathcal{D} \mid \boldsymbol{\theta}),$$

i.e., the parameters $\hat{\boldsymbol{\theta}}$ under which the data is most likely.

Instead of likelihood, we can maximize the **log-likelihood**:

$$\mathcal{L}(\boldsymbol{\theta}) := \log \prod_{i=1}^{N} p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) = \sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$$

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \ \mathcal{L}(\boldsymbol{\theta})$$

Since the log is a **strictly increasing** function, the log-likelihood has exactly the same maxima as the likelihood, i.e. **maximizing log-likelihood is equivalent to maximizing likelihood**.

The log-likelihood is expressed in **bits** (logarithm of base 2) or **nats** (logarithm of base e).

Why the Log?

Optimizing the log-likelihood (sum of terms) is usually easier than optimizing the likelihood (product of factors). Moreover, when using numerical optimizers, the log-likelihood is **numerically stable**, while the likelihood, a product of many small factors, becomes quickly numerical zero.

Why is maximizing the (log-)likelihood meaningful?

We want to find parameters $\boldsymbol{\theta}$ such that

$$p_{\boldsymbol{\theta}} \approx p^*$$

To quantify the approximation quality ("$\approx$") we require a distance or **divergence measure** for comparing $p_{\boldsymbol{\theta}}$ and p^*. The **Kullback-Leibler divergence** is the most widely used divergence:

$$\mathrm{KL}(p^* || p_{\boldsymbol{\theta}}) = \mathbb{E}_{p^*}\left[\log \frac{p^*(\boldsymbol{X})}{p_{\boldsymbol{\theta}}(\boldsymbol{X})}\right] = \int p^*(\boldsymbol{x}) \log \frac{p^*(\boldsymbol{x})}{p_{\boldsymbol{\theta}}(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$

It holds that

- $\mathrm{KL}(p^* || p_{\boldsymbol{\theta}}) \geq 0$ and
- $\mathrm{KL}(p^* || p_{\boldsymbol{\theta}}) = 0$ iff $p^* \equiv p_{\boldsymbol{\theta}}$ almost everywhere.

Kullback-Leibler Divergence and Log-likelihood

$$\begin{aligned}
\mathrm{KL}(p^* || p_\theta) &= \mathbb{E}_{p^*}\left[\log \frac{p^*(\boldsymbol{X})}{p_\theta(\boldsymbol{X})}\right] \\
&= \mathbb{E}_{p^*}\left[\log p^*(\boldsymbol{X}) - \log p_\theta(\boldsymbol{X})\right] \\
&= \underbrace{\mathbb{E}_{p^*}\left[\log p^*(\boldsymbol{X})\right]}_{\text{neg. entropy } -H[p^*], \text{const.}} - \mathbb{E}_{p^*}\left[\log p_\theta(\boldsymbol{X})\right]
\end{aligned}$$

The **Monte Carlo** estimator of $-\mathbb{E}_{p^*}\left[\log p_\theta(\boldsymbol{X})\right]$ is just the negative log-likelihood up to factor $1/N$:

$$-\mathbb{E}_{p^*}\left[\log p_\theta(\boldsymbol{X})\right] \approx -\frac{1}{N} \overbrace{\sum_{i=1}^{N} \log p_\theta(\boldsymbol{x}^{(i)})}^{\text{log-likelihood } \mathcal{L}(\theta)}$$

Kullback-Leibler Divergence and Log-likelihood cont'd

- maximum likelihood is equivalent to optimizing a **Monte Carlo estimate of the Kullback-Leibler divergence**
- for $N \to \infty$:

$$\textbf{maximizing (log-)likelihood} \Leftrightarrow \textbf{minimizing } \mathrm{KL}(p^* || p_\theta)$$

- hence, MLE is a **consistent distribution estimator**
- for finite data, of course, there will be random deviations and overfitting

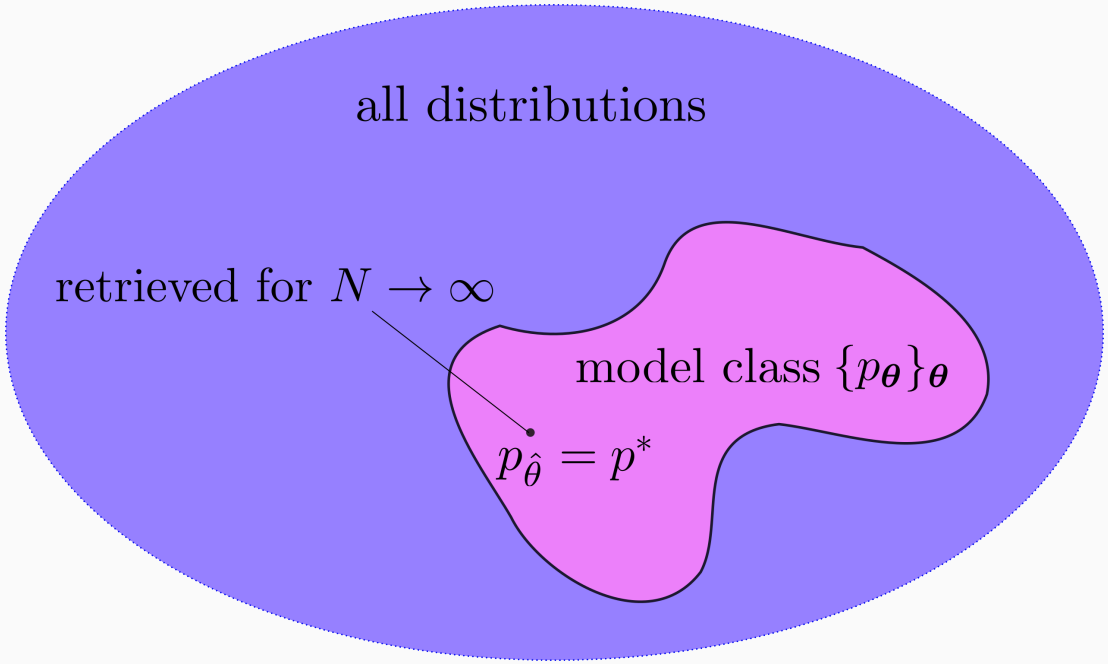
Kullback-Leibler Divergence and Log-likelihood cont'd

- **if** p^* is contained in the model class $\{p_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}}$,
- **if** we can actually find the **global** maximum likelihood solution

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \; \mathcal{L}(\boldsymbol{\theta})$$

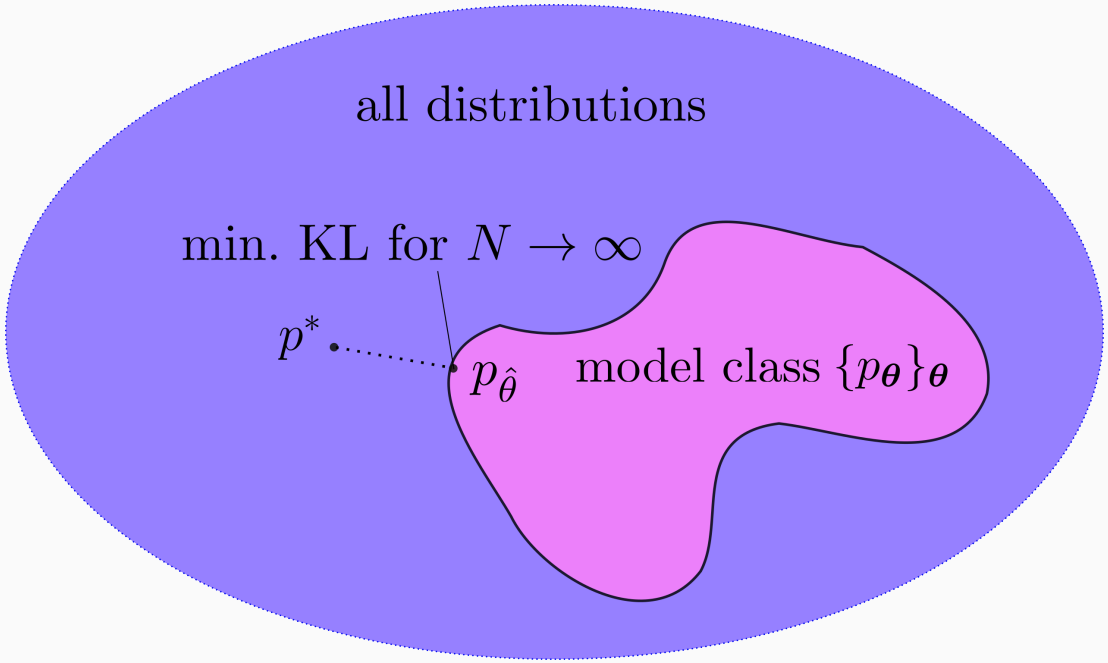
- and $N \to \infty$,

then MLE will converge to p^*!



Kullback-Leibler Divergence and Log-likelihood cont'd

When p^* is **not** contained in the model class, maximum likelihood still converges to the best approximation in KL-sense.



Assume data $\mathcal{D} = \{x^{(1)}, \ldots, x^{(N)}\}$ drawn iid from a Bernoulli distribution

$$x^{(i)} \sim \text{Bernoulli}(\theta), \quad x^{(i)} \in \{0, 1\}, \quad \theta \text{ unknown}$$

The likelihood of the dataset is

$$p(\mathcal{D} \mid \theta) = \prod_{i=1}^{N} \theta^{x^{(i)}} (1-\theta)^{1-x^{(i)}} = \theta^{\sum_i x^{(i)}} (1-\theta)^{N - \sum_i x^{(i)}}$$

Taking the log-likelihood:

$$\mathcal{L}(\theta) = \left(\sum_{i=1}^{N} x^{(i)} \right) \log \theta + \left(N - \sum_{i=1}^{N} x^{(i)} \right) \log(1-\theta)$$

Maximum Likelihood for Bernoulli cont'd

Differentiate and set to zero:

$$\frac{\partial \mathcal{L}}{\partial \theta} = \frac{\sum_i x^{(i)}}{\theta} - \frac{N - \sum_i x^{(i)}}{1-\theta} = 0$$

$$\sum_i x^{(i)} - \theta \sum_i x^{(i)} - \theta N + \theta \sum_i x^{(i)} = 0$$

$$\Rightarrow \quad \hat{\theta} = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}$$

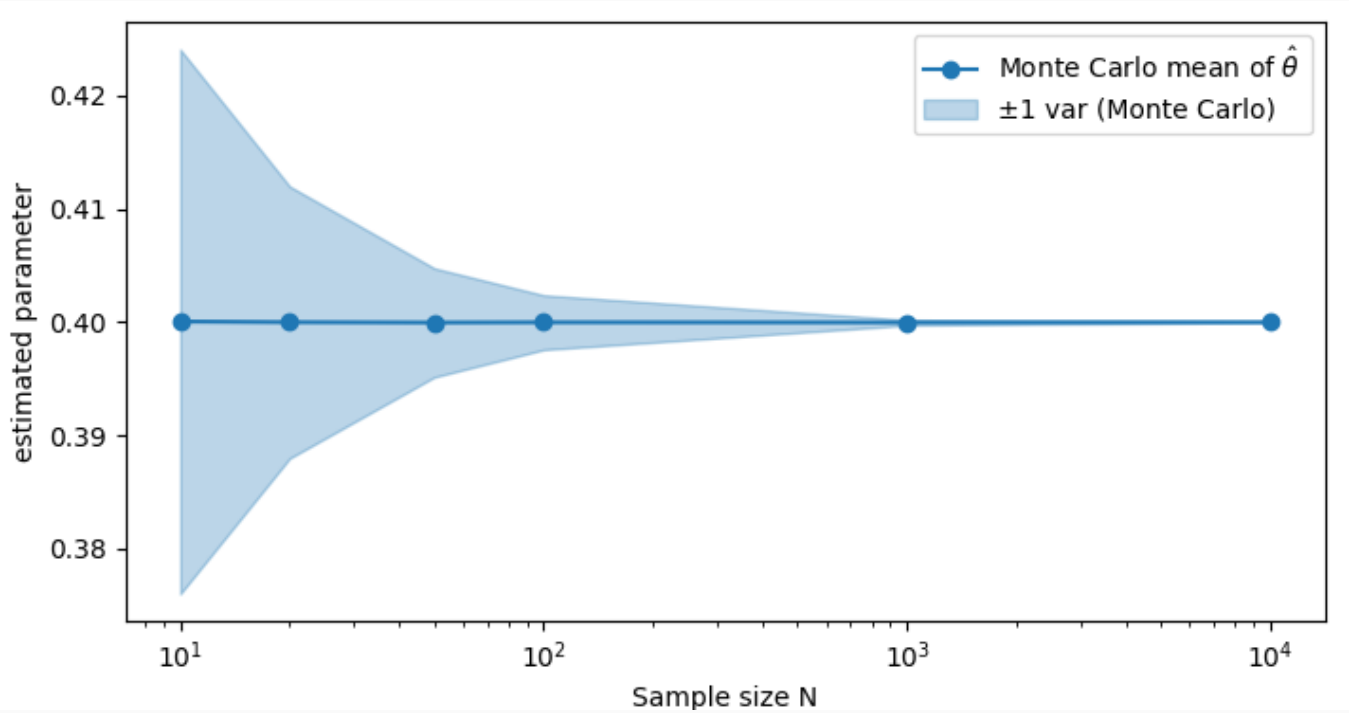
Hence: the MLE for θ is the empirical frequency of "successes" (normalized counts).

Maximum Likelihood for Bernoulli cont'd

Example

Assume a Bernoulli with success parameter $\theta = 0.4$:

- draw datasets of size $N \in \{10, 20, 50, 100, 1000, 10000\}$
- compute $\hat{\theta}$
- repeat this experiment 1 million time for each N
- plot mean and variance of $\hat{\theta}$ over 1 million trials



Recall the **Categorical distribution** over states $x_1, x_2, \ldots, x_K$:

$$p(x \mid \boldsymbol{\theta}) = \begin{cases} \theta_1 & \text{if } x = x_1 \\ \theta_2 & \text{if } x = x_2 \\ \vdots & \\ \theta_K & \text{if } x = x_K \end{cases}$$

where $\theta_k \geq 0$, $\sum_k \theta_k = 1$.

Akin to the Bernoulli, the MLE is just the normalized counts:

$$\hat{\theta}_k = \frac{\sum_{i=1}^{N} \mathbb{1}[x^{(i)} = x_k]}{N}$$

Assume data $\mathcal{D} = \{x^{(1)}, \ldots, x^{(N)}\}$ drawn iid from a Gaussian:

$$x^{(i)} \sim \mathcal{N}(\mu, \sigma)$$

The likelihood of the dataset is

$$p(\mathcal{D} \mid \mu, \sigma) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x^{(i)} - \mu)^2}{2\sigma^2}\right)$$

Taking the log-likelihood:

$$\mathcal{L}(\mu, \sigma) = -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(x^{(i)} - \mu)^2$$

Maximum Likelihood for Univariate Gaussian cont'd Example

To find the MLEs, differentiate the log-likelihood.

1. Derivative w.r.t. μ:

$$\frac{\partial \mathcal{L}}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^{N} (x^{(i)} - \mu) = 0 \quad \Rightarrow \quad \hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} x^{(i)}$$

Note $\hat{\mu}$ does not depend on σ. Hence, we can readily use $\hat{\mu}$ below.

2. Derivative w.r.t. σ:

$$\frac{\partial \mathcal{L}}{\partial \sigma} = -\frac{N}{\sigma} + \frac{1}{\sigma^3} \sum_{i=1}^{N} (x^{(i)} - \hat{\mu})^2 = 0$$

$$\Rightarrow \hat{\sigma} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x^{(i)} - \hat{\mu})^2}$$

empirical mean $\hat{\mu}$ and **empirical standard deviation** $\hat{\sigma}$.

Similarly, for **multivariate** Gaussians we get the **empirical mean (vector)** and **empirical covariance matrix** as MLE solution:

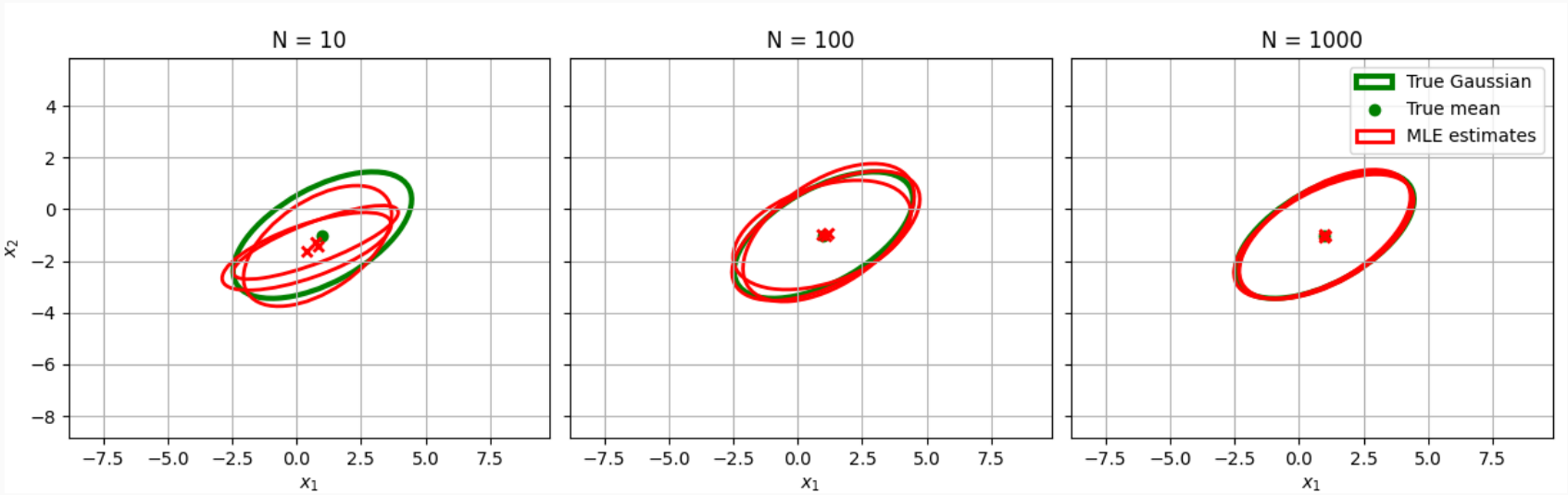
$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}^{(i)}$$

$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}^{(i)} - \hat{\boldsymbol{\mu}})(\boldsymbol{x}^{(i)} - \hat{\boldsymbol{\mu}})^T$$

Assume a multivariate Gaussian with

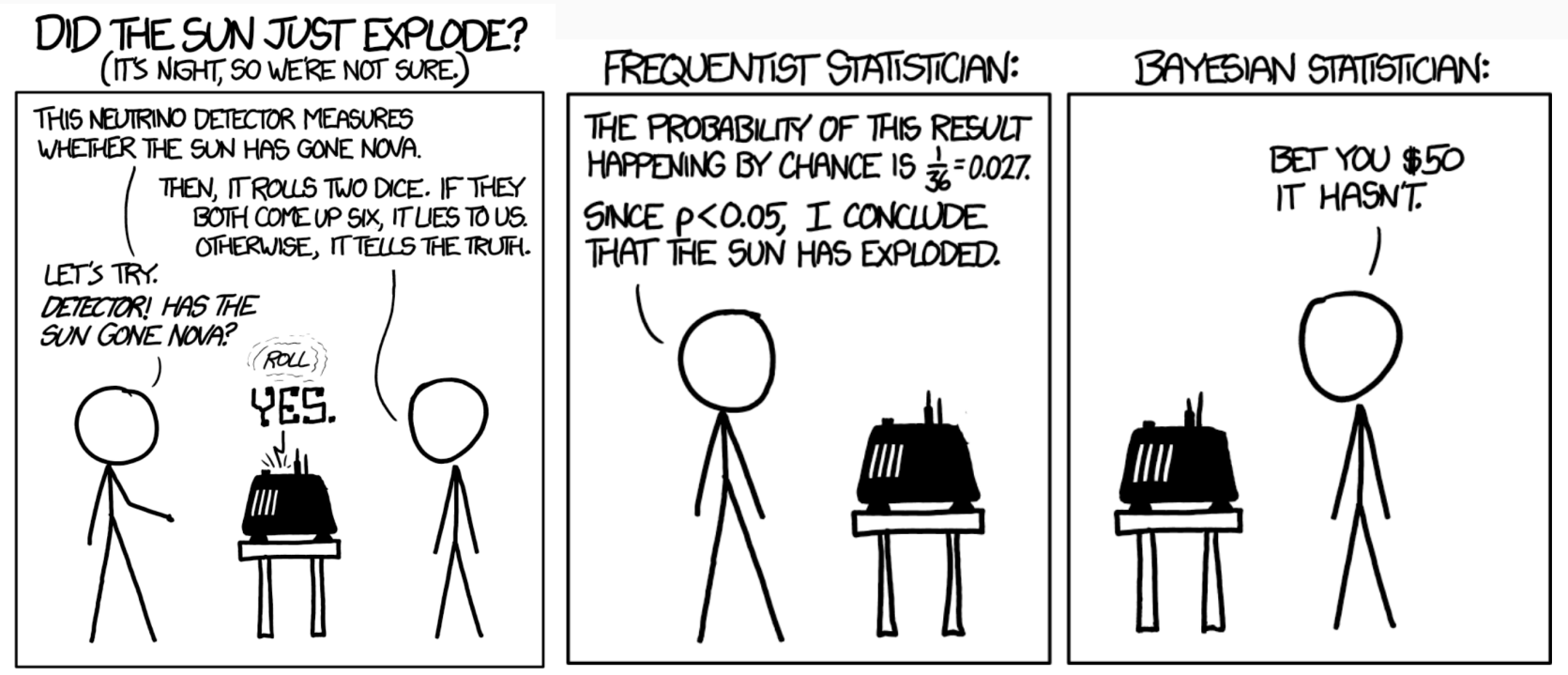
$$\boldsymbol{\mu} = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad \Sigma = \begin{pmatrix} 2 & 0.8 \\ 0.8 & 1 \end{pmatrix}$$

3 independent MLE estimations for $N \in \{10, 100, 1000\}$:



Bayesian Modeling and Inference

Bayesian vs. Frequentist Debate



The Bayesian vs. frequentist debate was a foundational dispute on the "correct" interpretation of probability.

- **frequentist:** $\mathbb{P}(A)$ is the **relative frequency** of $\omega_i \in A$
- **Bayesian:** $\mathbb{P}(A)$ is a **relaxed logical value** of $\omega \in A$

On the level of events, the conditional probability is

$$\overbrace{\mathbb{P}(A \mid B)}^{\text{A given B}} = \frac{\overbrace{\mathbb{P}(A \cup B)}^{\text{A and B}}}{\underbrace{\mathbb{P}(B)}_{\text{B}}}$$

Conversely,

$$\mathbb{P}(B \mid A) = \frac{\mathbb{P}(A \cup B)}{\mathbb{P}(A)}$$

Hence

$$\mathbb{P}(A \mid B)\,\mathbb{P}(B) = \mathbb{P}(B \mid A)\,\mathbb{P}(A)$$

leading to **Bayes theorem**:

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(B \mid A)\,\mathbb{P}(A)}{\mathbb{P}(B)}$$

Bayes Theorem cont'd

Essentially, Bayes theorem allows us to "invert the direction of conditioning"—hence also called the **law of inverse probability**.

Often it is straightforward to specify prior probabilities $\mathbb{P}(A)$, $\mathbb{P}(A^c) = 1 - \mathbb{P}(A)$ and conditional probabilities in one direction

- $\mathbb{P}(B \mid A)$
- $\mathbb{P}(B \mid A^c)$

but we need to reason in the other direction. Bayes theorem allows us to compute this:

$$\mathbb{P}(A \mid B) = \frac{\mathbb{P}(B \mid A)\,\mathbb{P}(A)}{\mathbb{P}(B)} = \frac{\mathbb{P}(B \mid A)\,\mathbb{P}(A)}{\underbrace{\mathbb{P}(B \mid A)\,\mathbb{P}(A)}_{\mathbb{P}(A \cup B)} + \underbrace{\mathbb{P}(B \mid A^c)\,\mathbb{P}(A^c)}_{\mathbb{P}(A^c \cup B)}}$$

- a dangerous illness has a prevalence of 1 in $100,000$
- a test for this illness is 99% accurate
- you are tested **positive**
- what is the probability that you have the illness?

Condition: ill	
$\mathbb{P}(\text{ill})$	0.00001
$\mathbb{P}(\text{positive} \mid \text{ill})$	0.99
$\mathbb{P}(\text{ill} \wedge \text{positive})$	$0.00001 \times 0.99 = 0.0000099$
Condition: not ill	
$\mathbb{P}(\text{ill}^c)$	0.99999
$\mathbb{P}(\text{positive} \mid \text{ill}^c)$	0.01
$\mathbb{P}(\text{ill}^c \wedge \text{positive})$	$0.99999 \times 0.01 = 0.0099999$

$$\begin{aligned}\mathbb{P}(\text{ill} \mid \text{positive}) &= \frac{\mathbb{P}(\text{positive} \mid \text{ill})\,\mathbb{P}(\text{ill})}{\mathbb{P}(\text{positive} \mid \text{ill})\,\mathbb{P}(\text{ill}) + \mathbb{P}(\text{positive} \mid \text{ill}^c)\,\mathbb{P}(\text{ill}^c)} \\ &= \frac{0.99 \times 0.00001}{0.99 \times 0.00001 + 0.01 \times 0.99999} = 0.00099.\end{aligned}$$

Less than 1 per mille chance of having the illness!

Bayesian vs. Frequentist Approaches to Learning

Situation: data $\mathcal{D}$ is given and we want to learn $\boldsymbol{\theta}$.

Frequentist:

- data $\mathcal{D}$ is random and can be equipped with probability
- however, there is only **one fixed** unknown $\boldsymbol{\theta}$, for which probability cannot be specified
- hence we need to **estimate** it

Bayesian:

- probability is an extension of logic
- data $\mathcal{D}$ is given, but $\boldsymbol{\theta}$ is **unknown**
- hence, we should include $\boldsymbol{\theta}$ in our model and just perform probabilistic inference about it

Bayesian Inference

- specify what you (don't) know about $\boldsymbol{\theta}$ in a **prior distribution**

$$p(\boldsymbol{\theta})$$

- specify how data is generated given $\boldsymbol{\theta}$ (**sample-wise likelihood**):

$$p(\boldsymbol{x} \mid \boldsymbol{\theta})$$

- for a dataset, under iid assumption, we get the **likelihood**

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{x}^{(i)} \mid \boldsymbol{\theta})$$

- by the product rule, we get a **joint of data and parameters**:

$$p(\mathcal{D}, \boldsymbol{\theta}) = p(\mathcal{D} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})$$

Bayesian Inference

- data $\mathcal{D}$ is observed, parameters $\boldsymbol{\theta}$ shall be inferred, hence we compute conditional

$$\overbrace{p(\boldsymbol{\theta} \mid \mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D} \mid \boldsymbol{\theta})}^{\text{likelihood}} \overbrace{p(\boldsymbol{\theta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{marginal likelihood}}} = \frac{\overbrace{p(\mathcal{D} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}^{\text{joint}}}{\int p(\mathcal{D} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta}}$$

- **Bayes' rule**, also called **law of inverse probability**
- the marginal likelihood $p(\mathcal{D})$ is also called the **evidence**
- since $p(\mathcal{D})$ does not depend on $\boldsymbol{\theta}$ we can also write

$$p(\boldsymbol{\theta} \mid \mathcal{D}) \propto p(\mathcal{D} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta}) = p(\mathcal{D}, \boldsymbol{\theta})$$

Recall the Bernoulli distribution for a binary RV:

$$p(x \mid \theta) = \theta^x (1-\theta)^{1-x} = \begin{cases} 1-\theta & \text{if } x = 0 \\ \theta & \text{if } x = 1 \end{cases}$$

For an iid dataset $\mathcal{D} = \{x^{(1)}, \ldots, x^{(N)}\}$ we get the likelihood

$$p(\mathcal{D} \mid \theta) = \prod_i p(x^{(i)} \mid \theta) = \theta^{\sum_i x^{(i)}} (1-\theta)^{N - \sum_i x^{(i)}}$$

$\sum_i x^{(i)}$ and $N - \sum_i x^{(i)}$ are the counts of 1's and 0's, respectively.

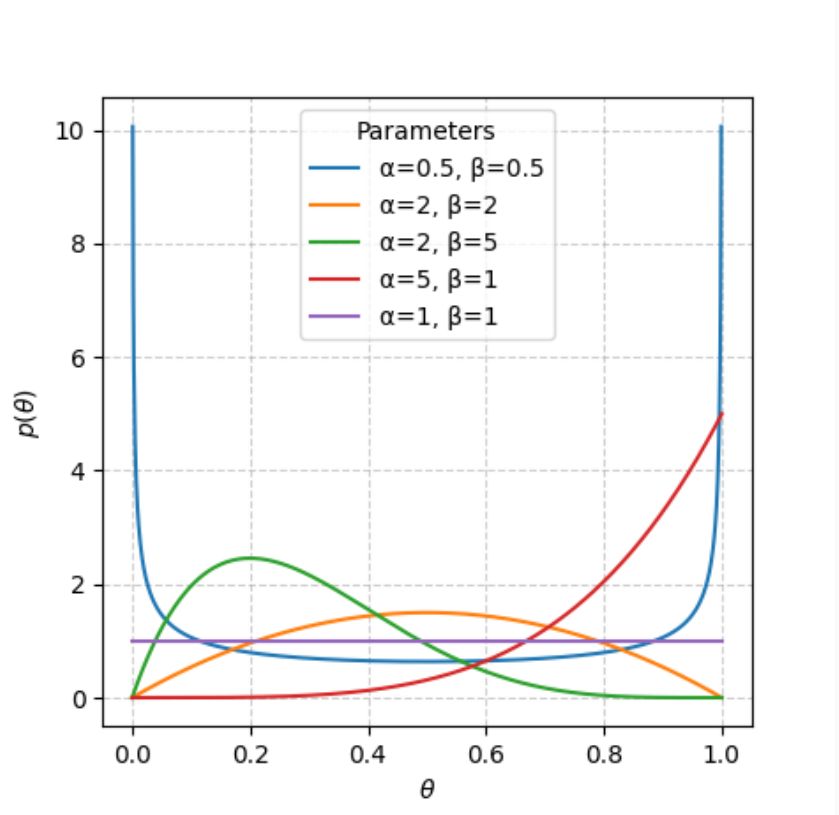
We want to equip θ with a prior distribution p_θ. Evidently, $0 \leq \theta \leq 1$ must hold, i.e. p_θ should be defined on $[0, 1]$. A suitable model is the **Beta** distribution.

$$p(\theta \mid \alpha, \beta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$

The normalization constant is the so-called **Beta function**

$$B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

The **Gamma function** $\Gamma(\cdot)$ is a continuous extension of the factorial function. In particular, $\Gamma(n) = (n-1)!$ for any $n \in \mathbb{N}$. Readily implemented in `NumPy` and co.



The parameters α, β must be strictly positive. For $\alpha = \beta = 1$, the Beta distribution is equal to the uniform distribution.

Bayesian Bernoulli cont'd

- let's start with a Beta prior

$$p(\theta \mid \alpha, \beta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)}$$

- standard trick: **work up to proportionality** ($\propto$ notation); here, this means we drop the normalization constant and write

$$p(\theta \mid \alpha, \beta) \propto \theta^{\alpha-1}\,(1-\theta)^{\beta-1}$$

- the likelihood is

$$p(\mathcal{D} \mid \theta) = \theta^{\sum_i x^{(i)}}\,(1-\theta)^{N-\sum_i x^{(i)}}$$

- prior and likelihood are of similar form, so the joint is

$$\begin{aligned} p(\mathcal{D}, \theta \mid \alpha, \beta) &= p(\mathcal{D} \mid \theta)\, p(\theta \mid \alpha, \beta) \\ &\propto \theta^{\sum_i x^{(i)}+\alpha-1}\,(1-\theta)^{N-\sum_i x^{(i)}+\beta-1} \end{aligned}$$

$$p(\mathcal{D}, \theta \mid \alpha, \beta) \propto \theta^{\sum_i x^{(i)} + \alpha - 1} (1 - \theta)^{N - \sum_i x^{(i)} + \beta - 1}$$

Proportionality trick again; (posterior is proportional to joint):

$$p(\theta \mid \mathcal{D}, \alpha, \beta) \propto p(\mathcal{D}, \theta \mid \alpha, \beta)$$

Hence the posterior is

$$p(\theta \mid \mathcal{D}, \alpha, \beta) \propto \theta^{\sum_i x^{(i)} + \alpha - 1} (1 - \theta)^{N - \sum_i x^{(i)} + \beta - 1}$$

Compare this with the Beta distribution:

$$p(\theta \mid \alpha, \beta) \propto \theta^{\alpha - 1} (1 - \theta)^{\beta - 1}$$

Same form! Hence, the posterior is again Beta with:

$$\alpha' = \sum_i x^{(i)} + \alpha \qquad \beta' = N - \sum_i x^{(i)} + \beta$$

Bayesian Bernoulli model

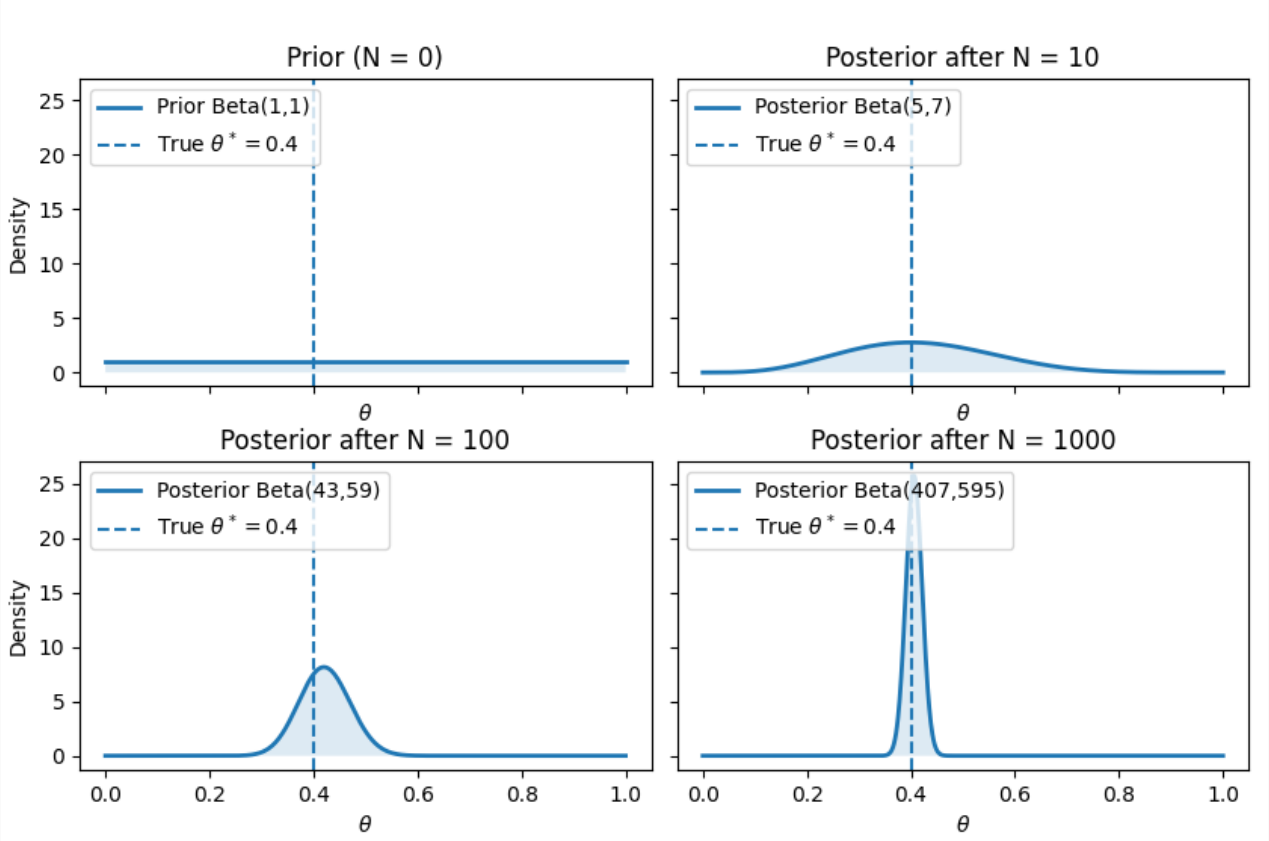
- start with Beta prior: $\theta \sim \text{Beta}(\alpha, \beta)$
- count 0's and 1's in $\mathcal{D}$: $N_1 = \sum_i x^{(i)}$ $N_0 = N - \sum_i x^{(i)}$
- posterior: $\theta \mid \mathcal{D} \sim \text{Beta}(\alpha + N_1, \beta + N_0)$

Note:

- like the MLE solution, the Bayesian solution is based entirely on **data counts** N_0, N_1
- α, β can be interpreted is **prior pseudo counts**
- for Beta prior and Bernoulli likelihood, the posterior is also Beta—we say that the Beta is the **conjugate prior** for Bernoulli

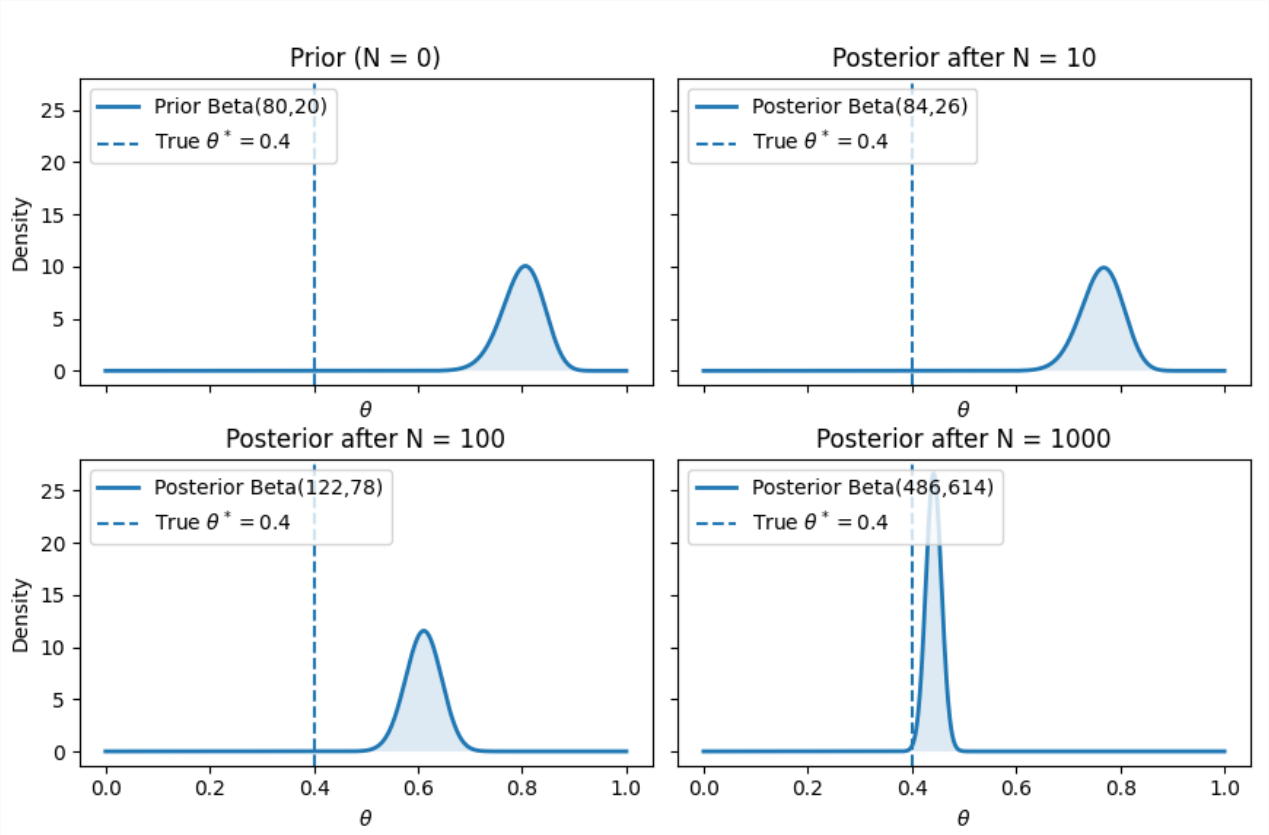
Bayesian Bernoulli cont'd

Example



- drawing samples from Bernoulli with $\theta^* = 0.4$; prior $\alpha = 1$, $\beta = 1$
- when observing more data, the posterior **concentrates** around θ^*
- for $N \to \infty$, the posterior collapses to a **point mass** on θ^*

When the Prior is Wrong



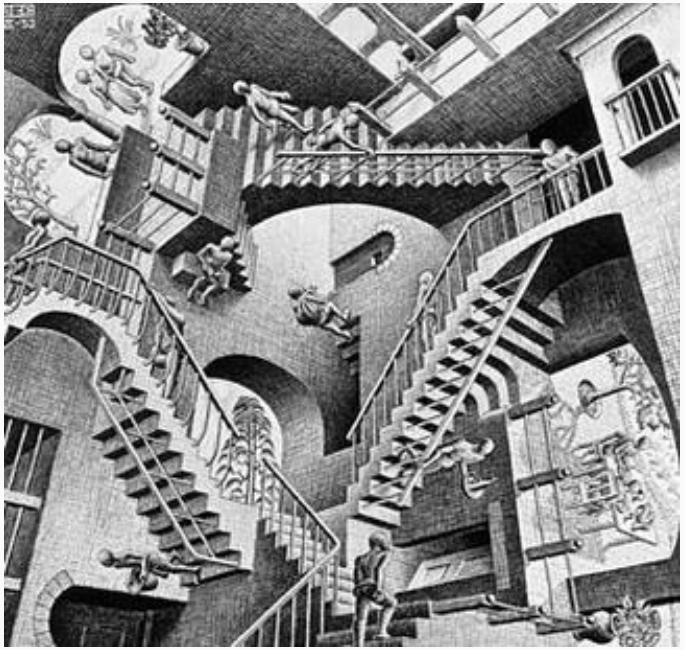
- when starting with a wrong prior (here $\alpha = 80$, $\beta = 20$) it can take very long until the posterior is "convinced otherwise"
- **"garbage in—garbage out"** (universal principle, not restricted to Bayesian inference)

When Parameters Are Not Identifiable

Non-identifiable parameters are not distinguished by data:

$$\exists\, \theta_1 \neq \theta_2 : \quad p(\mathcal{D} \mid \theta_1) = p(\mathcal{D} \mid \theta_2) \quad \forall\, \mathcal{D}$$

- likelihood $p(\mathcal{D} \mid \theta)$ is flat or **multimodal**
- Bayesian posterior also might remain flat or multimodal
- represents **ambiguity**



Bayesian approach: infer parameters from data.

But what shall we do with the posterior $p(\theta \mid \mathcal{D})$?

- **ultimately, we want to do predictions**
- the joint of **test data** $\boldsymbol{x}^*$ and $\boldsymbol{\theta}$ given training data $\mathcal{D}$ is

$$p(\boldsymbol{x}^*, \boldsymbol{\theta} \mid \mathcal{D}) = p(\boldsymbol{x}^* \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathcal{D})$$

- we are ultimately not interested in $\boldsymbol{\theta}$, hence **marginalize**:

$$p(\boldsymbol{x}^* \mid \mathcal{D}) = \int p(\boldsymbol{x}^* \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathcal{D})\, \mathrm{d}\boldsymbol{\theta}$$

- this is the **Bayes predictive distribution**—average of all possible models $p(\boldsymbol{x}^* \mid \boldsymbol{\theta})$ weighted with their posterior $p(\boldsymbol{\theta} \mid \mathcal{D})$
- incorporates **model uncertainty** and proceeds more careful than a point estimate like MLE

For the Bernoulli model with Beta posterior:

$$\theta \,|\, \mathcal{D} \sim \text{Beta}(\alpha', \beta') \quad \text{with} \quad \alpha' = \alpha + N_1, \quad \beta' = \beta + N_0$$

We want to predict a test sample $x^* \in \{0, 1\}$:

$$p(x^* = 1 \,|\, \mathcal{D}) = \int \overbrace{p(x^* = 1 \,|\, \theta)}^{= \theta} \, p(\theta \,|\, \mathcal{D}) \, \mathrm{d}\theta$$

$$= \mathbb{E}_{p(\theta|\mathcal{D})}[\theta] = \frac{\alpha'}{\alpha' + \beta'}$$

Furthermore,

$$p(x^* = 0 \,|\, \mathcal{D}) = 1 - p(x^* = 1 \,|\, \mathcal{D}) = \frac{\beta'}{\alpha' + \beta'}$$

Hence $x^* \,|\, \mathcal{D} \sim \text{Bernoulli}\left(\frac{\alpha'}{\alpha' + \beta'}\right)$ (equivalent to MLE regularized with pseudo-counts α and β)

Conjugate Priors

Conjugacy: prior and posterior belong to the same distribution family (when using a suitable likelihood) $\Rightarrow$ **analytically tractable** Bayesian inference.

Likelihood	**Conjugate Prior**
Bernoulli / Binomial	Beta(α, β)
Categorical	Dirichlet$(\boldsymbol{\alpha})$
Gaussian (known σ)	$\mathcal{N}(\mu_0, \sigma_0^2)$
Gaussian	Normal–Inverse-Gamma$(\mu_0, \lambda, \alpha, \beta)$
Poisson	Gamma(α, β)
Exponential	Gamma(α, β)

Summary

- **maximum likelihood estimation (MLE)**
- MLE = minimization of a Monte Carlo estimator of the KL between true data distribution and model
- **Bayesian inference**: include parameters into the model and perform probabilistic inference
- Bayesian inference not restricted to parameters, but can in principle be applied to **any** unknown quantity