Statistic

for machine learning

Tran Trong Khiem

AI lab tranning

2024/05/29

Introduction

Let $\mathbf{z} \in \mathbb{R}^L$ be an unknown vector of values, and $\mathbf{y} \in \mathbb{R}^D$ be some noisy measurement of $\mathbf{z}$.

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z} \mid \boldsymbol{\mu}_\mathbf{z}, \boldsymbol{\Sigma}_\mathbf{z})$$

$$p(\mathbf{y} \mid \mathbf{z}) = \mathcal{N}(\mathbf{y} \mid \mathbf{W}\mathbf{z} + \mathbf{b}, \boldsymbol{\Sigma}_\mathbf{y})$$

where $\mathbf{W}$ is a matrix of size $D \times L$. This is an example of a **linear Gaussian system**.
$p(\mathbf{z}, \mathbf{y}) = p(\mathbf{z})p(\mathbf{y} \mid \mathbf{z})$ is an $L + D$ dimensional Gaussian, with mean and covariance given by:

$$\mu = W\mu_z + b$$

$$\Sigma = \begin{pmatrix} \Sigma_z & \Sigma_z W^T \\ W\Sigma_z & \Sigma_y + W\Sigma_z W^T \end{pmatrix}$$

Bayes rule for Gaussians

The posterior over the latent is given by

$$p(z|y) = \mathcal{N}(z|\mu_{z|y}, \Sigma_{z|y})$$

where

$$\Sigma_{z|y}^{-1} = \Sigma_z^{-1} + W^T \Sigma_y^{-1} W$$

$$\mu_{z|y} = \Sigma_{z|y} \left[W^T \Sigma_y^{-1} (y - b) + \Sigma_z^{-1} \mu_z \right]$$

The normalization constant of the posterior is given by

$$p(y) = \int \mathcal{N}(z|\mu_z, \Sigma_z) \mathcal{N}(y|Wz + b, \Sigma_y)\, dz = \mathcal{N}(y|W\mu_z + b, \Sigma_y + W\Sigma_z W^T)$$

Example: Inferring an unknown scalar

Suppose we make N noisy measurements y_i of some underlying quantity z; let us assume the measurement noise has fixed precision $\lambda_y = \frac{1}{\sigma^2}$, so the likelihood is

$$p(y_i|z) = \mathcal{N}(y_i|z, \lambda_y^{-1})$$

We use a Gaussian prior for the value of the unknown source:

$$p(z) = \mathcal{N}(z|\mu_0, \lambda_0^{-1})$$

$=>$**We want to compute** $p(z|y_1, \ldots, y_N, \sigma^2)$**.** We define:

$$\begin{aligned} y &= (y_1, \ldots, y_N), \\ W &= 1_N \text{ (an } N \times 1 \text{ column vector of 1's)}, \\ \Sigma_y^{-1} &= \text{diag}(\lambda_y I). \end{aligned}$$

Example: Inferring an unknown scalar

We get :

$$p(z|y) = \mathcal{N}\left(z|\mu_N, \lambda_N^{-1}\right)$$
$$\lambda_N = \lambda_0 + N\lambda_y$$
$$\mu_N = \frac{N\lambda_y\bar{y} + \lambda_0\mu_0}{\lambda_N}$$

By using Bayes rule. λ_N is the prior precision λ_0 plus N units of measurement precision λ_y
The posterior mean μ_N is a convex combination of the MLE $\bar{y}$ and the prior mean μ_0 .It can rewrite as :

$$p(z|D, \sigma^2) = \mathcal{N}\left(z|\mu_N, \tau_N^2\right)$$

$$\tau_N^2 = \frac{\sigma^2\tau_0^2}{N\tau_0^2 + \sigma^2}$$

$$\mu_N = \tau_N^2\left(\frac{\mu_0}{\tau_0^2} + \frac{N\bar{y}}{\sigma^2}\right)$$

Example: Inferring an unknown scalar

If $N = 1$, we can rewrite the posterior after seeing a single observation as follows (where we define $\Sigma_y = \sigma^2$, $\Sigma_0 = \tau_0^2$ and $\Sigma_1 = \tau_1^2$ to be the variances of the likelihood, prior and posterior):

$$p(z|y) = \mathcal{N}(z|\mu_1, \Sigma_1)$$

$$\Sigma_1 = \frac{\Sigma_y \Sigma_0}{\Sigma_0 + \Sigma_y}$$

$$\mu_1 = \Sigma_1 (\frac{\mu_0}{\Sigma_0} + \frac{y}{\Sigma_y})$$

We can rewrite the posterior mean in 3 different ways:

$$\mu_1 = \mu_0 \frac{\Sigma_y}{\Sigma_y + \Sigma_0} + y \frac{\Sigma_0}{\Sigma_y + \Sigma_0}$$

$$= \mu_0 + (y - \mu_0) \frac{\Sigma_0}{\Sigma_0 + \Sigma_y}$$

$$= y - (y - \mu_0) \frac{\Sigma_y}{\Sigma_y + \Sigma_0}$$

Example: Inferring an unknown scalar

Another way to quantify the amount of shrinkage is in terms of the signal-to-noise ratio, which is defined as follows:

$$\text{SNR} = \frac{\mathbb{E}[Z^2]}{\mathbb{E}[\varepsilon^2]} = \frac{\Sigma_0 + \mu_0^2}{\Sigma_y} \tag{1}$$

where $Z \sim \mathcal{N}(\mu_0, \Sigma_0)$ is the true signal, $y = z + \varepsilon$ is the observed signal, and $\varepsilon \sim \mathcal{N}(0, \Sigma_y)$ is the noise term.

Example: inferring an unknown vector

Suppose we have an unknown quantity of interest, $z \in \mathbb{R}^D$, a Gauss prior $p(z) = \mathcal{N}(\mu_z, \Sigma_z)$.
If we "know nothing" about z a priori, we can set $\Sigma_z = \infty I =>$ no infor about prior.By symmetry, it seems reasonable to set $\mu_z = 0$.
Suppose we make N noisy but independent measurements of z, denoted as $y_n \sim \mathcal{N}(z, \Sigma_y)$, each of size D.

$$p(D|z) = \prod_{n=1}^{N} \mathcal{N}(y_n|z, \Sigma_y) \propto \mathcal{N}(y|z, \frac{1}{N}\Sigma_y^N)$$

Using this, and setting $W = I$ and $b = 0$, we can then use Bayes' rule for Gaussians to compute the posterior over z:

$$p(z|y_1, \ldots, y_N) = \mathcal{N}(z|\hat{\mu}, \hat{\Sigma})$$

(2)

$$\hat{\Sigma^{-1}} = \Sigma_z^{-1} + N\Sigma_y^{-1} \quad (3.66)$$

$$\hat{\mu} = \hat{\Sigma}\left(\Sigma_y^{-1}(N\bar{y}) + \Sigma_z^{-1}\mu_z\right) \quad (3.67)$$

Surrogate loss

The 0-1 loss is a non-smooth step function, making it difficult to optimize.The surrogate is usually chosen to be a maximally tight convex upper bound, which is then easy to minimize.
For example, consider a probabilistic binary classifier, which produces the following distribution over labels:

$$p(\tilde{y} \mid x, \theta) = \sigma(\tilde{y}\eta) = \frac{1}{1 + e^{-\tilde{y}\eta}} \tag{3}$$

Where $\eta = f(x;\theta) = log(\frac{p(y|x,\theta)}{1-p(y|x,\theta)})$. Hence the log loss is given by

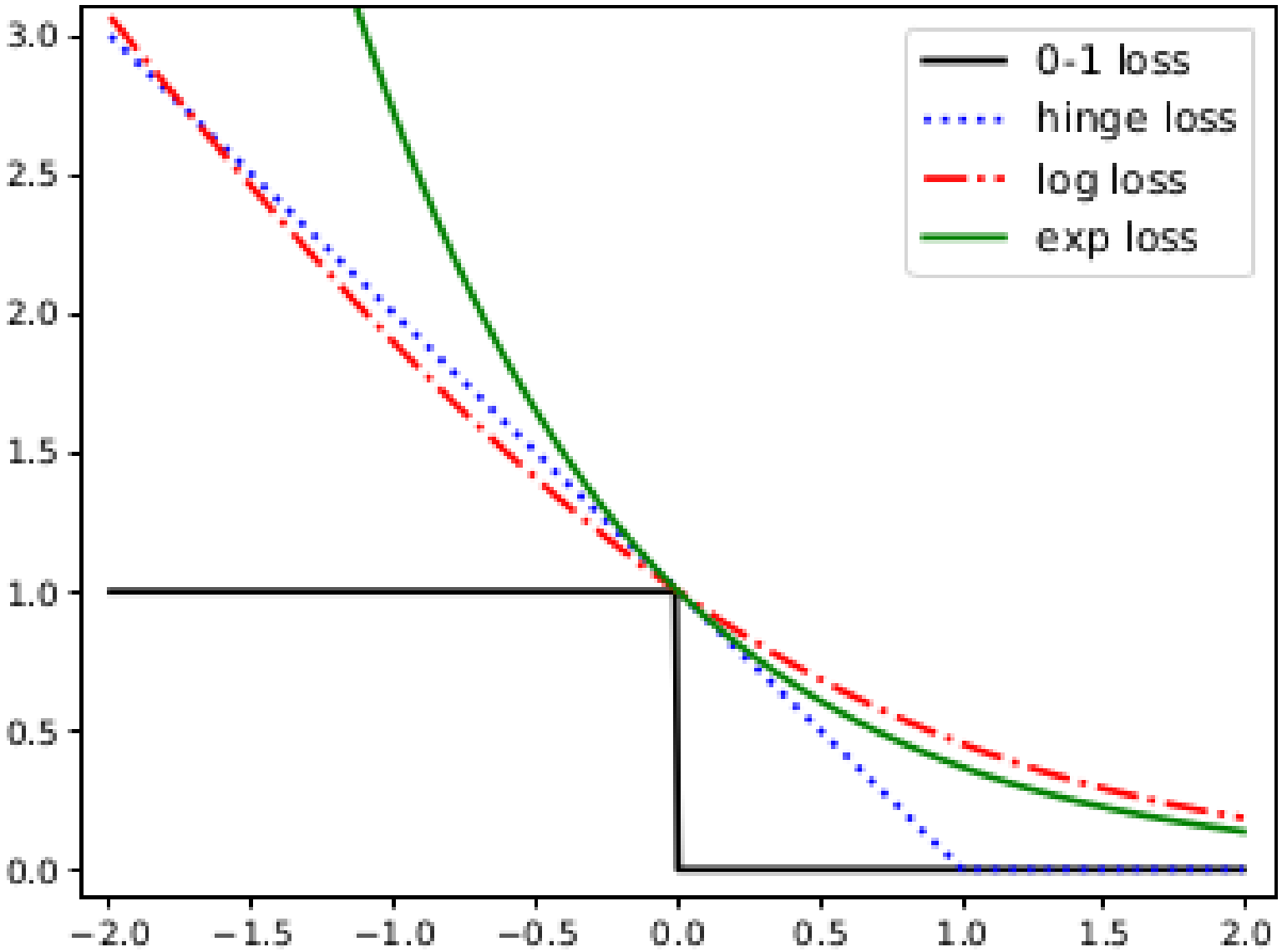
$$\ell(\tilde{y}, \eta) = -\log p(\tilde{y} \mid \eta) = \log(1 + e^{-\tilde{y}\eta}) \tag{4}$$

The hinge loss is defined as follows:

$$\text{hinge}(\tilde{y}, \eta) = \max(0, 1 - \tilde{y}\eta) = (1 - \tilde{y}\eta)_+, \tag{5}$$

Thus we see that minimizing the negative log likelihood is equivalent to minimizing a (fairly tight) upper bound on the empirical 0-1 loss.

Surrogate loss



Shrinkage estimate

Univariate case

If σ^2 is a known constant, the likelihood for μ has form (iid assumtion):

$$p(D|\mu) \propto \exp\left(-\frac{1}{2\sigma^2}\sum_{n=1}^{N}(y_n-\mu)^2\right) \tag{6}$$

The conjugate prior is another Gaussian, $\mathcal{N}(\mu|m,\tau^2)$.Apply Bayes's rule for Gaussians :

$$p(\mu|D,\sigma^2) = \mathcal{N}(\mu|\hat{m},\hat{\tau^2})$$

$$\hat{\tau^2} = \frac{\sigma^2\tau^2}{N\tau^2+\sigma^2}$$

$$\hat{m} = \hat{\tau^2}(\frac{m}{\tau^2}+\frac{N\bar{y}}{\sigma^2})$$

Where $\bar{y}\hat{=}\frac{1}{N}\sum_{n=1}^{N}y_n$ is the empirical mean.

Univariate case

We can then rewrite the posterior as follows:

$$p(\mu|D,\kappa) = \mathcal{N}(\mu|\hat{m}, \hat{\lambda^{-1}}) \tag{7}$$

$$\hat{\lambda} = \lambda + N\kappa$$

$$\hat{m} = \frac{N\kappa\bar{y} + m\lambda}{\hat{\lambda}}$$

Posterior after seeing N = 1 examples

Consider the posterior after seeing a single data point y(so $N = 1$).Then the posterior mean can be written in the following:

$$\hat{m} = \frac{\lambda}{\hat{\lambda}}m + \frac{\kappa}{\hat{\lambda}}y$$

$$= m + \frac{\kappa}{\hat{\lambda}}(y - m)$$

$$= y - \frac{\lambda}{\hat{\lambda}}(y - m)$$

The first equation is a convex combination of the prior mean and the data. The second equation is the prior mean adjusted towards the data y.

Posterior variance

The square root of this is called the standard error of the mean:

$$se(\mu) \hat{=} \mathrm{V}[\mu|\mathrm{D}]^{1/2}$$

Suppose we use an uninformative prior for μ by setting $\lambda = 0$.In this case, the posterior mean is equal to the MLE, $\hat{m} = \bar{y}$.
We approximate σ^2 by the sample variance :

$$s^2 \hat{=} \frac{1}{N} \sum_{n=1}^{N} (y_n - \bar{y})^2$$

Hence $\hat{\lambda} = N\hat{\kappa} = \frac{N}{s^2}$, so SEM becomes:

$$se(\mu) = \mathbb{V}[\mu|\mathbb{D}]^{1/2} = \frac{1}{\hat{\lambda}^{1/2}}$$

In addition, we can use the fact that 0.95 of a Gaussian distribution is contained within 2 standard deviations of the mean to approximate the 0.95 credible interval for μ using

$$I_{95}(\mu|D) = y \pm 2\sqrt{\frac{s^2}{N}}$$

Multivariate case

For D-dimensional data, the likelihood has the following form

$$p(\mathbf{D}|\boldsymbol{\mu}) = \prod_{n=1}^{N} \left(\frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y}_n - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{y}_n - \boldsymbol{\mu}) \right) \right)$$

$$= \left(\frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \right)^N \exp\left(-\frac{1}{2} \sum_{n=1}^{N} (\mathbf{y}_n - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{y}_n - \boldsymbol{\mu}) \right)$$

$$\propto N(\bar{\mathbf{y}}|\boldsymbol{\mu}, \frac{1}{N}\Sigma)$$

Where $\bar{y} = \frac{1}{N}\sum_{n=1}^{N} y_n$.Thus we replace the set of observations with their mean, and scale down the covariance by a factor of N.

Multivariate case

For simplicity, we will use a conjugate prior, which in this case is a Gaussian :

$$p(\mu) = \mathcal{N}(\mu|m, V)$$

Base on Bayses's rule for gaussian, we have :

$$p(\mu|\mathcal{D}, \Sigma) = \mathcal{N}(\mu|\hat{m}, \hat{V})$$

$$\hat{V}^{-1} = V^{-1} + N\Sigma^{-1}$$

$$\hat{m} = \hat{V}(\Sigma^{-1}(N\bar{y}) + V^{-1}m)$$

Noninformative priors

When we have little or no domain specific knowledge, it is desirable to use an uninformative, noninformative or objective priors.
For example, if we want to infer a real valued quantity, such as a location parameter $\mu \in \mathcal{R}$, we can use a flat prior $p(\mu) \propto 1$.
Unfortunately, there is no unique way to define uninformative priors, and they all encode some kind of knowledge.

Hierarchical priors

Bayesian models require specifying a prior $p(\theta)$ for the parameters.
The parameters of the prior are called hyperparameters, and will be denoted by ξ.
If these are unknown, we can put a prior on them; this defines a **hierarchical Bayesian model**, or **multi-level model**

$$\xi \rightarrow \theta \rightarrow D$$

Assume the prior hyper-parameters is fixed(minimally informative prior).

$$p(\xi, \theta, D) = p(\xi)p(\theta \mid \xi)p(D \mid \theta)$$

The hope is that we can learn the hyperparameters by treating the parameters themselves as datapoints.

Empirical priors

We discussed hierarchical Bayes as a way to infer parameters from data. Unfortunately, posterior inference in such models can be computationally challenging.
In this section, we discuss a computationally convenient approximation:
Step 1: compute a point estimate of the hyperparameters, $\hat{\xi}$
Step 2: compute the conditional posterior, $p(\theta|\hat{\xi}, D)$
To estimate the hyper-parameters, we can maximize the marginal likelihood:

$$\hat{\xi}_{\text{mml}}(D) = \arg\max_{\xi} p(D|\xi) = \arg\max_{\xi} \int p(D|\theta)p(\theta|\xi)\, d\theta \tag{8}$$

$=>$**Type II maximum likelihood**

Credible intervals

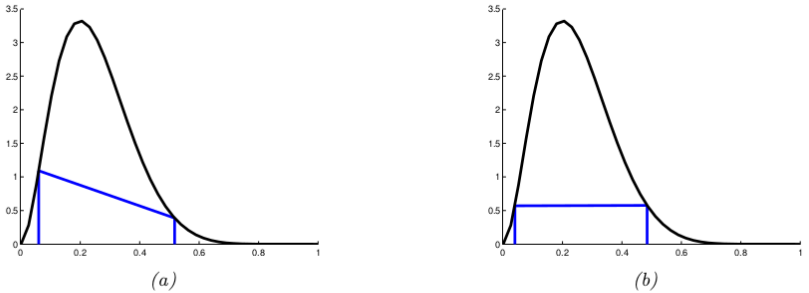
We define a $100(1-\alpha)\%$ credible interval to be a (contiguous) region $C = (\ell, u)$ which contains $1-\alpha$ of the posterior probability mass.

$$C_\alpha(D) = (\ell, u) : P(\ell \leq \theta \leq u \mid D) = 1 - \alpha$$

We usually choose one such that there is $(1-\alpha)/2$ mass in each tail; this is called a **central interval**.
If the posterior has a known functional form, we can compute the posterior central interval using $\ell = F^{-1}(\alpha/2)$ and $u = F^{-1}(1-\alpha/2)$, where F is the cdf of the posterior, and F^{-1} is the inverse cdf.
E.g: If the posterior is Gaussian, $p(\theta|D) = \mathcal{N}(0,1)$, and $\alpha = 0.05$, then we have $\ell = \Phi^{-1}(\alpha/2) = -1.96$, and $u = \Phi^{-1}(1-\alpha/2) = 1.96$,
In general, it is often hard to compute the inverse cdf of the posterior.

Credible intervals

A problem with **central intervals** is that there might be **points outside the central interval which have higher probability than points that are inside**



The highest posterior density or **HPD** region, which is the set of points which have a probability above some threshold.We find the threshold p^* on the pdf such that

$$1 - \alpha = \int_{\theta : p(\theta|D) > p^*} p(\theta|D)\, d\theta \tag{9}$$

and then define the HPD as

$$C_\alpha(D) = \{\theta : p(\theta|D) \geq p^*\} \tag{10}$$

Plugin approximation

Once we have computed the posterior over the parameters, we can compute the posterior predictive distribution over outputs given :

$$p(y|x,D) = \int p(y|x,\theta)p(\theta|D)\,d\theta \tag{11}$$

A very simple approximation is to assume there is just a single best model, $\hat{\theta}$ such as the MLE.

$$p(\theta|D) = \delta(\theta - \hat{\theta})$$

If we use this approximation, then the predictive distribution can be obtained by simply "plugging in" the point estimate into the likelihood:

$$p(y \mid x,D) = \int p(y \mid x,\theta)p(\theta \mid D)\,d\theta \quad \approx \int p(y \mid x,\theta)\delta(\theta - \hat{\theta})\,d\theta = p(y \mid x,\hat{\theta})$$

This called a **plug-in approximation**.This approach is equivalent to the standard approach used in most of machine learning, in which we first fit the model (i.e.compute a point estimate $\hat{\theta}$ and then use it to make predicitons $=>$ **can suffer from overfitting and overconfidence.**

Example: scalar input, binary output

Suppose we want to perform binary classification, so $y \in 0, 1$. We will use a model of the form

$$p(y|x;\theta) = Ber(y|\sigma(w^T x + b))$$

Where

$$\sigma(a) \hat{=} \frac{e^a}{1+e^a}$$

is the sigmoid or logistic function map $\mathcal{R} -> [0,1]$. Other words,

$$p(y=1|x;\theta) = \sigma(w^T x + b) = \frac{1}{1+e^{-(w^T x+b)}}$$

=>Called **logistic regression**

Example: scalar input, binary output

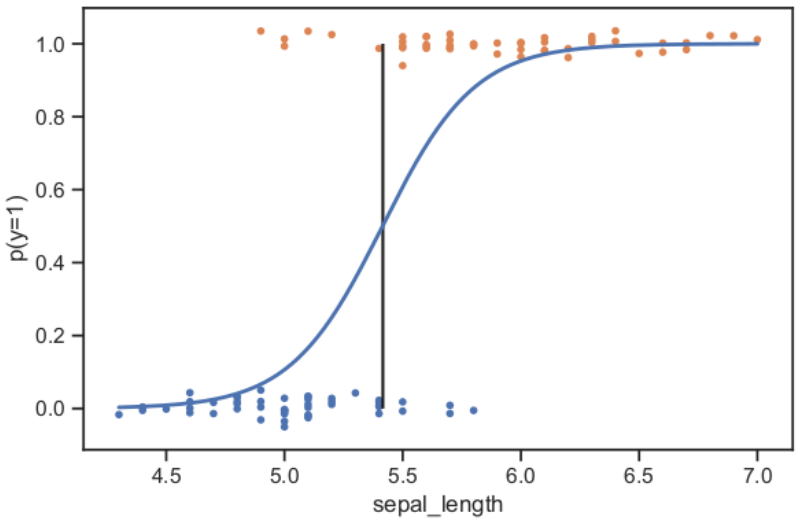
We first fit a 1d logistic regression model of the following form:

$$p(y = 1 \mid x; \theta) = \sigma(b + wx)$$

The **decision boundary** is defined to be the input value x^* where $p(y = 1|x^*; \theta) = 0.5$. We can solve for this value as follows

$$\sigma(b + wx^*) = \frac{1}{1 + e^{-(b+wx^*)}} = 1/2$$

$$x^* = -\frac{b}{w}$$



Example: scalar input, binary output

The above approach **does not model the uncertainty in our estimate of the parameters.**
To capture this additional uncertainty, we can use a Bayesian approach to approximate the posterior $p(\theta|D)$. Using Monte Carlo approimation:

$$p(y = 1 \mid x, D) \approx \frac{1}{S} \sum_{s=1}^{S} p(y = 1 \mid x, \theta^s)$$

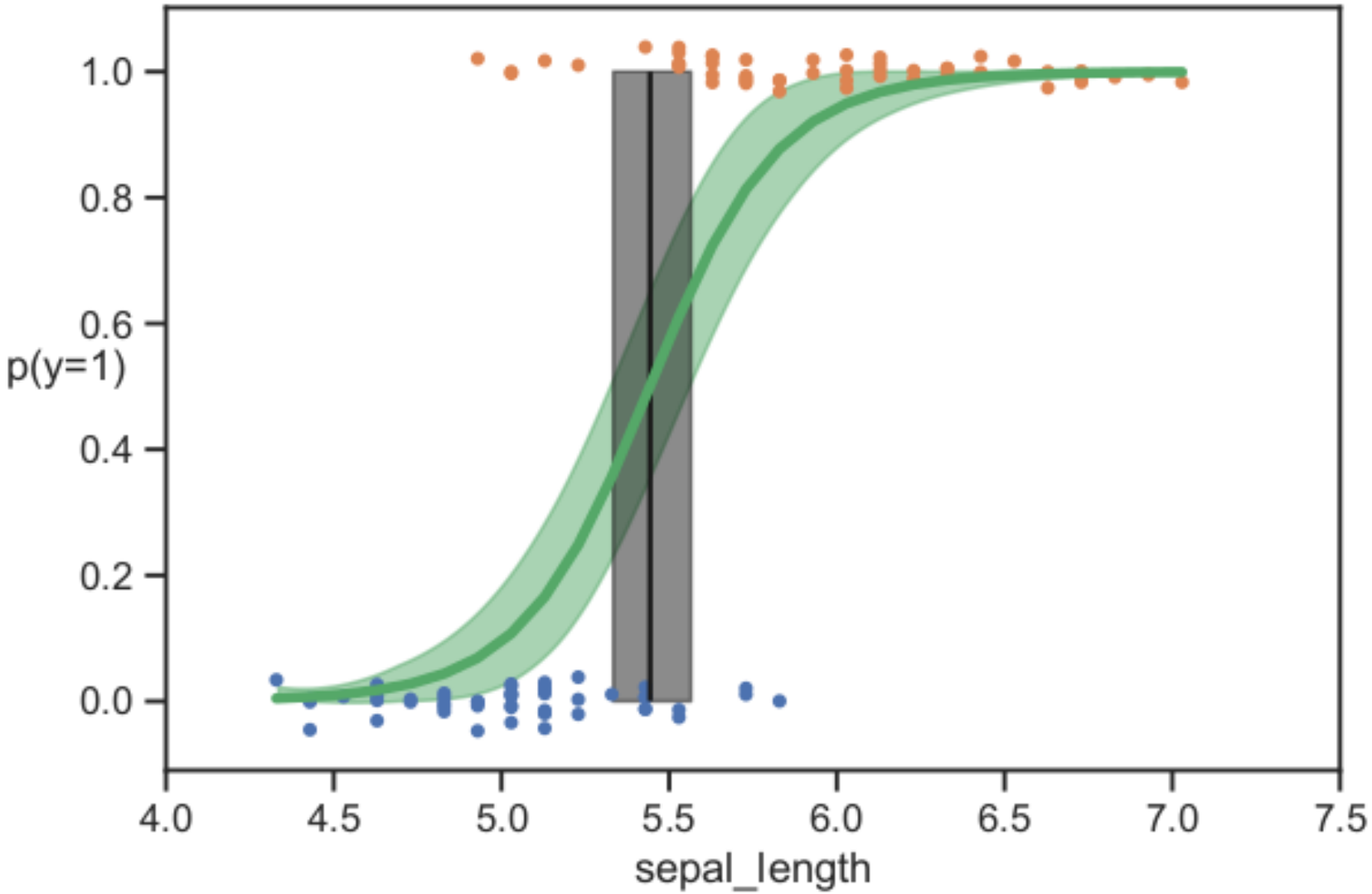
Where $\theta_s \sim p(\theta \mid D)$ is posterior sample. We see that there is now a range of predicted probabilities for each input.
Compute a distribution over the location of the decision boundary by using the Monte Carlo approximation :

$$p(x^* \mid D) \approx \frac{1}{S} \sum_{s=1}^{S} \delta \left(x^* - \frac{-b_s}{w_s} \right)$$

where $(b_s, w_s) = \theta_s$.

Example: scalar input, binary output

plots the mean and 0.95 credible interval of this function

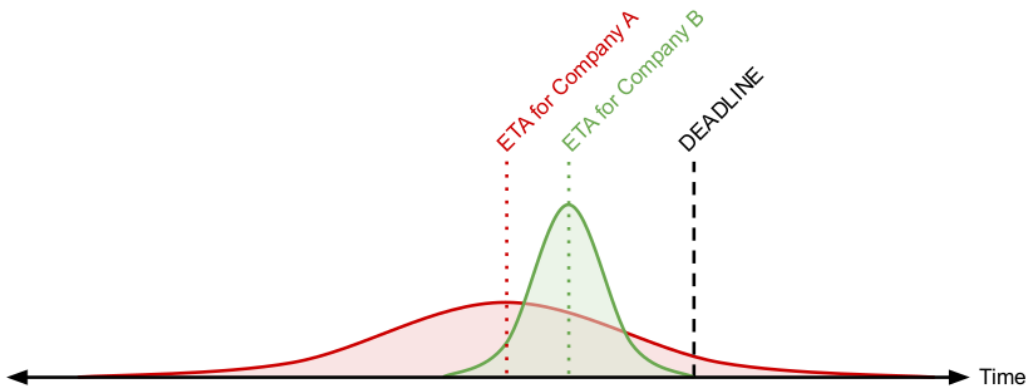


Example: binary input, scalar output

Now suppose we want to predict the delivery time for a package, $y \in R$, if shipped by company A vs B.
Using binary feature $x \in 0, 1$, where $x = 0$ means A, $x = 1$ mean B.

$$p(y \mid x, \theta) = N(y \mid \mu_x, \sigma_x^2)$$

$\theta = (\mu_0, \mu_1, \sigma_0, \sigma_1)$ are the parameters of the model.
Suppose we have only used each company once, so our training set has the form $D = \{(x_1 = 0, y_1 = 15), (x_2 = 1, y_2 = 20)\}$
The MLE for the means will be the empirical means, $\hat{\mu}_0 = 15$ and $\hat{\mu}_1 = 20$, but the MLE for the standard deviations will be zero, $\hat{\sigma}_0 = \hat{\sigma}_1 = 0$, since we only have a single sample from each "class".



Computational issues

Given a likelihood $p(D|\theta)$ and a prior $p(\theta)$, we can compute the posterior $p(\theta|D)$ using Bayes' rule. However, actually performing this computation is usually intractable.
$=>$**need to approximate the posterior**(trade off accuracy, simplicity, and speed)
Example: Use the problem of approximating the posterior of a beta-Bernoulli model. Specifically, the goal is to approximate.

$$p(\theta \mid D) \propto [\prod_{n=1}^{N} \text{Bin}(y_n \mid \theta)]\text{Beta}(\theta \mid 1, 1)$$

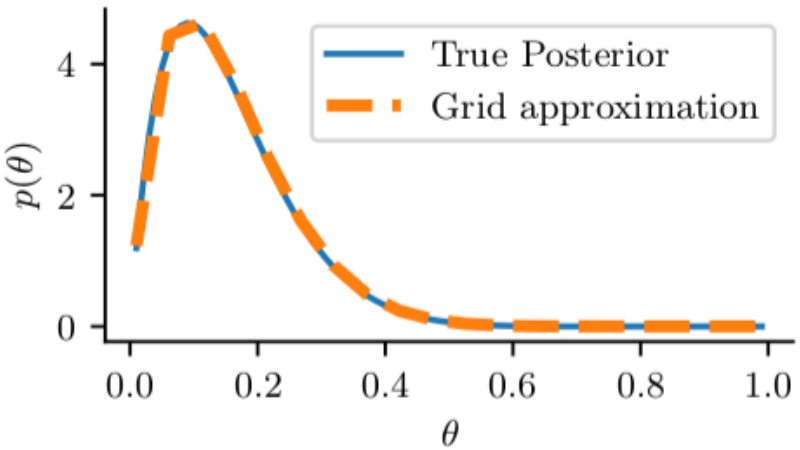
where D consists of 10 heads and 1 tail (so the total number of observations is $N = 11$), and we use a uniform prior.

Grid approximation

Partition the space of possible values for the unknowns into a finite set of possibilities, call them $\theta_1, \ldots, \theta_K$, and then to approximate the posterior by brute-force enumeration, as follows:

$$p(\theta = \theta_k \mid D) \approx \frac{p(D \mid \theta_k)p(\theta_k)}{\sum_{k'=1}^{K} p(D \mid \theta_{k'})p(\theta_{k'})}$$

$=>$ This called **grid approximation**.
This approach does not scale to problems in more than 2 or 3 dimensions, because the number of grid points grows exponentially with the number of dimensions.



Quadratic (Laplace) approximation

We write the posterior as follows :

$$p(\theta \mid D) = \frac{1}{Z} e^{-\mathcal{E}(\theta)}$$

Where $\mathcal{E}(\theta) = -p(\theta, D)$ is the energy function, $Z = p(D)$ is the normalization constant.
Performing a Taylor series expansion around the mode $\hat{\theta}$ (i.e., the lowest energy state) we get :

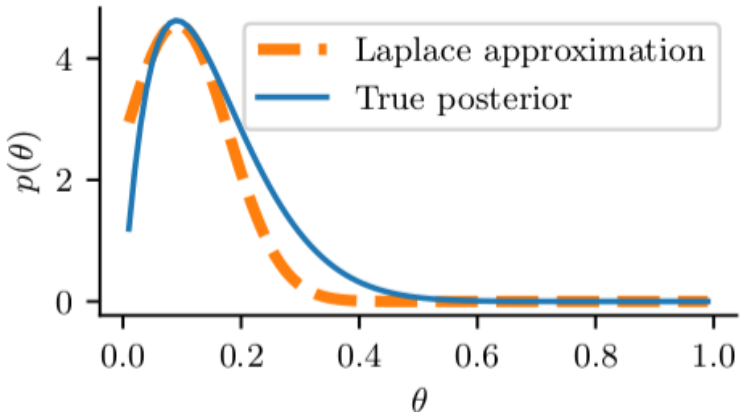
$$\mathcal{E}(\theta) \approx \mathcal{E}(\hat{\theta}) + (\theta - \hat{\theta})^T g + \frac{1}{2}(\theta - \hat{\theta})^T H (\theta - \hat{\theta})$$

where g is the gradient at the mode, and H is the Hessian. Since $\hat{\theta}$is the mode, the gradient term is zero. Hence :

$$p(\hat{\theta}, D) = e^{-\mathcal{E}(\theta)} = e^{\frac{1}{2}(\theta - \hat{\theta})^T H (\theta - \hat{\theta})}$$

$$\hat{p}(\theta \mid D) = \frac{1}{Z}\hat{p}(\theta, D) = \mathcal{N}(\theta \mid \hat{\theta}, H^{-1})$$

Quadratic approximation



The Laplace approximation is easy to apply, since we can leverage existing optimization algorithms to compute the MAP estimate, and then we just have to compute the Hessian at the mode.

not a particularly good approximation

The parameter of interest lies in the constrained interval $\theta \in [0, 1]$

Whereas the Gaussian assumes an unconstrained space, $\theta \in R$.

$=>$ Using change of variable $\alpha = logit(\theta)$

Variational approximation

VI attempts to approximate an intractable probability distribution, such as $p(\theta \mid D)$, with one that is tractable, $q(\theta)$, so as to minimize some discrepancy D between the distributions:

$$q^* = \arg\min_{q \in \mathcal{Q}} D(q, p)$$

Where $\mathcal{Q}$ is some tractable family of distributions(Gauss, Exponent,..)
If we define D to be the KL divergence,we can derive a lower bound to the log marginal likelihood(ELBO).
By maximizing the ELBO, we can improve the quality of the posterior approximation.
VI is a fast, optimization-based method, it can give a biased approximation to the posterior, since it is restricted to a specific function form $q \in \mathcal{Q}$.

Markov Chain Monte Carlo (MCMC) approximation

Using a non parametric approximation in terms of a set of samples $q(\theta) \approx \frac{1}{S} \sum_{s=1}^{S} \delta(\theta - \theta^s)$
This is called a **Monte Carlo approximation to the posterior**.
The key issue is how to create the posterior samples $\theta_s \sim p(\theta \mid D)$ efficiently, without having to evaluate the normalization constant $p(D)$.A common approach to this problem is known as **Markov chain Monte Carlo or MCMC**.

Frequentist statistics *

Attempts have been made to devise approaches to **statistical inference** that **avoid treating parameters like random variables**, and which thus **avoid the use of priors and Bayes rule**. This alternative approach is known as **frequentist statistics, classical statistics** or **orthodox statistics**

The **basic idea** is to **represent uncertainty by calculating how a quantity estimated from data** (such as a parameter or a predicted label) **would change if the data were changed**.

Sampling distributions

In frequentist statistics, uncertainty is not represented by the posterior distribution of a random variable, but instead by the **sampling distribution** of an **estimator**.
"Estimator" is defined $\delta : D \to A$ is a decision procedure that specifies what action to take given some observed data
Use $\hat{\Theta}$ to denote estimator.
The output of this function, when applied to a specific dataset of size N, is denoted $\hat{\theta} = \hat{\Theta}(D)$, where $D = \{x_1, \ldots, x_N\}$.
The key idea in frequentist statistics is to **view the data D as a random variable**, and the **parameters** from which the data are drawn, θ^* , as a **fixed but unknown constant**.
Thus $\hat{\theta} = \hat{\Theta}(D)$ is a random variable, and its distribution is known as the sampling distribution of the estimator.
Suppose we create S different datasets, each of the form

$$D^{(s)} = \{x_n \sim p(x_n \mid \theta^*) : n = 1 : N\}$$

We denote this by $D^{(s)} \sim \theta^*$ for brevity.

Sampling distributions

Now we apply the estimator to each $D^{(s)}$ to get a set of estimates, $\{\hat{\theta}(D^{(s)})\}$.
As we let $S \to \infty$, the distribution induced by this set is the sampling distribution of the estimator.

$$\text{SamplingDist}(\hat{\Theta}, \theta^*) = \text{PushThrough}(p(\tilde{D} \mid \theta^*), \hat{\Theta})$$

where we push the data distribution through the estimator function to induce a distribution of estimates.

Gaussian approximation of the sampling distribution of the MLE

The most common estimator is the MLE. When the sample size becomes large, the sampling distribution of the MLE for certain models becomes Gaussian.

Theorem 4.7.1. If the parameters are identifiable, then

$$\text{SamplingDist}(\hat{\Theta}_{\text{MLE}}, \theta^*) \to \mathcal{N}\left(\cdot \mid \theta^*, (NF(\theta^*))^{-1}\right)$$

where $F(\theta^*)$ is the Fisher information matrix.

The above result says that the distribution of $NF(\theta^*)^{1/2}(\hat{\theta} - \theta^*)$ approaches $\mathcal{N}(0, I)$, where $\hat{\theta} = \hat{\Theta}_{\text{MLE}}(\tilde{D})$.

The Fisher information matrix measures the amount of curvature of the log-likelihood surface at its peak.

The Fisher information matrix (FIM) is defined to be the covariance of the gradient of the log likelihood :

$$F(\theta) = \mathbb{E}_{x \sim p(x|\theta)}[\nabla \log p(x|\theta) \nabla \log p(x|\theta)^T]$$

Gaussian approximation of the sampling distribution of the MLE

Hence the (i, j)'th entry has the form

$$F_{ij} = \mathbb{E}_{x\sim\theta}\left[\frac{\partial}{\partial\theta_i}\log p(x|\theta)\frac{\partial}{\partial\theta_j}\log p(x|\theta)\right] \tag{4.221}$$

Theorem 4.7.2. If $\log p(x|\theta)$ is twice differentiable, and under certain regularity conditions, the Fisher information matrix (FIM) is equal to the expected Hessian of the negative log-likelihood (NLL).

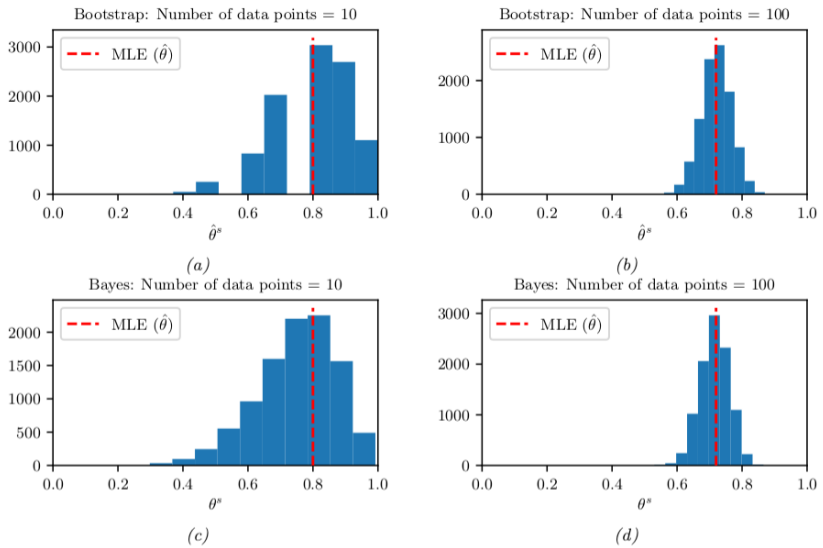
$$F_{ij} = -\mathbb{E}_{x\sim\theta}\left[\frac{\partial^2}{\partial\theta_i\partial\theta_j}\log p(x|\theta)\right]$$

A log-likelihood function with high curvature (large Hessian) will result in a low variance estimate, since the parameters are "well determined" by the data, and hence robust to repeated sampling.
In the scalar case, we have that $V(\hat{\theta} - \theta^*) \to \frac{1}{NF(\theta^*)}$.

The distribution of $\frac{\hat{\theta}-\theta^*}{se}$ approaches $\mathcal{N}(0, 1)$

Bootstrap approximation of the sampling distribution of any estimator

If we knew the true parameters θ^* , we could generate many fake datasets(S), each of size N , from the true distribution, using $\tilde{D}^{(s)} = \{x_n \sim p(x_n \mid \theta^*) : n = 1 : N\}$.
We could then compute our estimate from each sample, $\theta^s = \hat{\Theta}(\tilde{D}^{(s)})$, use the empirical distribution of the resulting θ^s as our estimate of the sampling distribution.
More precisely, the idea is **to generate each** $\tilde{D}^{(s)}$ by **sampling** N **data points with replacement** from the **original dataset.**



Bootstrap is a "poor man's" posterior

what is the connection between the parameter estimates $\hat{\theta}^s = \Theta(\hat{D}(s))$ computed by boostrap and parameter values sampled from the posterior ?

- quite different concept
- quite similar if the estimator is MLE and the prior is not very strong(not heavily influencing the posterior).

Bootstrap can be slower than posterior sampling. The reason is that the bootstrap has to generate S sampled datasets, and then fit a model to each one.

Confidence intervals

A $100(1-\alpha)\%$ **confidence interval** for parameter θ is defined as an estimator that returns an interval that captures the true parameter with probability at least $1-\alpha$.
Denote the estimator by $I(D) = (l(D), u(D))$.
The sampling distribution of this estimator is the distribution that is induced by sampling $\tilde{D} \sim \theta^*$ and then computing $I(\tilde{D})$. We require that:

$$\Pr(\theta^* \in I(\tilde{D}) | \tilde{D} \sim \theta^*) \geq 1-\alpha$$

Example: Suppose that $\hat{\theta} = \hat{\Theta}(D)$ is an estimator for some parameter with true but unknown value θ^*. The sampling distribution of $\Delta = \theta^* - \hat{\theta}$ is known.
Let δ_+ and δ_- denote its $\alpha/2$ and $1-\alpha/2$ quantiles :

$$\Pr(\delta_- \leq \Delta \leq \delta_+) = \Pr(\delta_- \leq \theta^* - \hat{\theta} \leq \delta_+) = 1-\alpha$$

$$\Pr(\hat{\theta} + \delta_- \leq \theta^* \leq \hat{\theta} + \delta_+) = 1-\alpha$$

$$I(D) = (L, U) = (\hat{\theta_-}(D) + \delta(D), \hat{\theta_+}(D) + \delta(D))$$

Common to assume a Gaussian approximation to the sampling distribution $\hat{\theta} \approx N(\theta^*, \hat{\text{se}}^2)$

Caution: Confidence intervals are not credible

- The frequentist approach(0.95 confidence intervals) : the procedure for generating confidence intervals (CIs) will contain the true value 95% of the time. We repeatedly sample datasets $\tilde{D}$ from θ^*, and compute their CIs to get $I(\tilde{D})$, have $\Pr(\theta^* \in I(\tilde{D})) = 0.95$.

These concepts are quite different :

- In the frequentist approach, θ is treated as an unknown fixed constant, and the data is treated as random.
- In the Bayesian approach, we treat the data as fixed and the parameter as random