02477 – Bayesian Machine Learning: Lecture 4

Michael Riis Andersen

Technical University of Denmark,
DTU Compute, Department of Applied Math and Computer Science

Outline

Re-cap from last week and a few words about graphical models

Bayesian linear regression model: the key equations

- Linear regression model with Gaussian noise and Gaussian priors

$$y_n = f(\phi(\boldsymbol{x}_n), \boldsymbol{w}) + e_n$$

- Given design matrix $\boldsymbol{\Phi} \in \mathbb{R}^{N \times D}$ and observations $\boldsymbol{y} \in \mathbb{R}^N$

$$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|0, \alpha^{-1}\boldsymbol{I}) \qquad (\textit{prior})$$

$$p(\boldsymbol{y}|\boldsymbol{w}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{\Phi}\boldsymbol{w}, \beta^{-1}\boldsymbol{I}) \qquad (\textit{likelihood})$$

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \beta^{-1}\boldsymbol{I} + \alpha^{-1}\boldsymbol{\Phi}\boldsymbol{\Phi}^T) \qquad (\textit{marginal likelihood})$$

$$p(\boldsymbol{w}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}_N, \boldsymbol{S}_N) \qquad (\textit{posterior})$$

  with *posterior parameters*

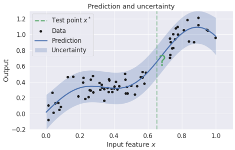
$$\boldsymbol{m}_N = \beta \boldsymbol{S}_N \boldsymbol{\Phi}^T \boldsymbol{y} \qquad (\textit{posterior mean})$$

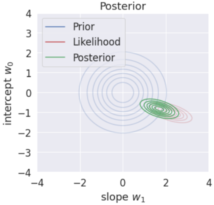
$$\boldsymbol{S}_N = \left(\alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1} \qquad (\textit{posterior covariance})$$

- *Two hyperparameters*

  α: prior precision of the regression weights
  β: precision of the measurements

- *Lazy notation*: We should actually write $p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta)$ etc., but we often suppress dependency of hyperparameter to ease notation

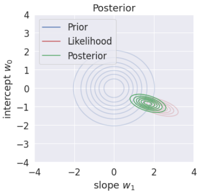




Posterior Predictive distributions

- The *posterior distribution* is $p(\boldsymbol{w}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}, \boldsymbol{S})$ with

$$\boldsymbol{m} = \beta \boldsymbol{S} \boldsymbol{\Phi}^T \boldsymbol{y}$$
$$\boldsymbol{S} = \left(\alpha \boldsymbol{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}\right)^{-1}$$

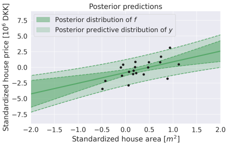


- When making predictions $\boldsymbol{x}^*$ using Bayesian methods, we *average over all possible parameters values weighted by the posterior*

$$f(\boldsymbol{x}^*|\boldsymbol{w}) = \boldsymbol{w}^T \phi(\boldsymbol{x}^*)$$
$$y(\boldsymbol{x}^*) = f(\boldsymbol{x}^*|\boldsymbol{w}) + \epsilon$$

- First, we write the likelihood corresponding to the new input $\boldsymbol{x}_*$

$$p(y^*|\boldsymbol{x}^*, \boldsymbol{w}) = \mathcal{N}(y^*|\boldsymbol{w}^T \phi(\boldsymbol{x}^*), \beta^{-1})$$



- .. and marginalize with respect to the posterior distribution (sum rule)

$$p(y^*|\boldsymbol{y}) = \int p(y^*|\boldsymbol{x}^*, \boldsymbol{w}) p(\boldsymbol{w}|\boldsymbol{y}) \mathrm{d}\boldsymbol{w}$$

- This is called *the posterior predictive distribution*

Posterior predictive distributions: how to make predictions?

- First, we write up the *likelihood* corresponding for the new input $\boldsymbol{\phi}_* = \boldsymbol{\phi}(\boldsymbol{x}_*)$:

$$p(y_*|\boldsymbol{x}_*, \boldsymbol{w}) = \mathcal{N}(y_*|\boldsymbol{w}^T\boldsymbol{\phi}_*, \beta^{-1})$$

- For MAP (and similar for MLE), we simply *plug in* the estimate of $\boldsymbol{w}$

$$p(y_*|\boldsymbol{y}, \boldsymbol{x}_*) \approx \mathcal{N}(y_*|\hat{\boldsymbol{w}}_{MAP}^T\boldsymbol{\phi}_*, \beta^{-1})$$

- Bayesian: use the *sum rule* to *marginalize* wrt. the *posterior* distribution

$$p(y_*|\boldsymbol{y}, \boldsymbol{x}_*) = \int p(y^*|\boldsymbol{x}_*, \boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{y})\mathrm{d}\boldsymbol{w} = \mathcal{N}(y_*|\hat{\boldsymbol{w}}_{MAP}^T\boldsymbol{\phi}_*, \boldsymbol{\phi}_*^T\boldsymbol{S}\boldsymbol{\phi}_* + \beta^{-1})$$

- If posterior covariance $\boldsymbol{S}$ is small, we get *approximately* the same result

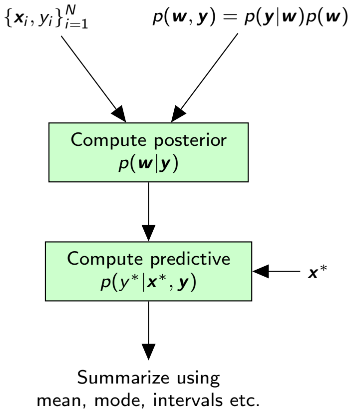
- We can think of the *MAP* solution as an *approximate* posterior distribution

$$p(\boldsymbol{w}|\boldsymbol{y}) \approx \delta(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}}) = \begin{cases} \infty & \text{if } \boldsymbol{w} = \boldsymbol{w}_{\text{MAP}} \\ \boldsymbol{0} & \text{otherwise} \end{cases}$$

- Such a distribution is called *Dirac's delta* distribution (mental picture: Gaussian with mean $\boldsymbol{w}_{\text{MAP}}$ and variance going to zero)

- MAP is sometimes called *poor man's Bayes*, but can still be a useful tool!

Bayesian inference for supervised learning



- Same principles for linear regression, logistic regression, neural networks etc. etc.

Hyperparameters and the evidence approximation

- Posterior depends on the hyperparameters α and β (but often suppressed in notation)

$$p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta) = \frac{p(\boldsymbol{y}|\boldsymbol{w}, \beta)p(\boldsymbol{w}|\alpha)}{p(\boldsymbol{y}|\alpha, \beta)}$$

- We could assign priors to α and β to get the posterior on α and β given the data

$$p(\alpha, \beta|\boldsymbol{y}) \propto p(\boldsymbol{y}|\alpha, \beta)p(\boldsymbol{\alpha})p(\beta)$$

- fully Bayesian solution: use the sum rule to marginalize over all unknowns $(\boldsymbol{w}, \alpha, \beta)$

$$p(y_*|\boldsymbol{y}, \boldsymbol{x}_*) = \int p(y^*|\boldsymbol{x}_*, \boldsymbol{w}_*, \beta)p(\boldsymbol{w}|\boldsymbol{y}, \alpha, \beta)p(\alpha, \beta|\boldsymbol{y})\, \mathrm{d}\boldsymbol{w}\mathrm{d}\alpha\mathrm{d}\beta$$

The evidence approximation

- We estimate $\hat{\alpha}, \hat{\beta}$ by *optimizing the marginal likelihood* $p(\boldsymbol{y}|\alpha, \beta)$

$$\hat{\alpha}, \hat{\beta} = \arg\max_{\alpha,\beta} \log p(\boldsymbol{y}|\alpha, \beta)$$

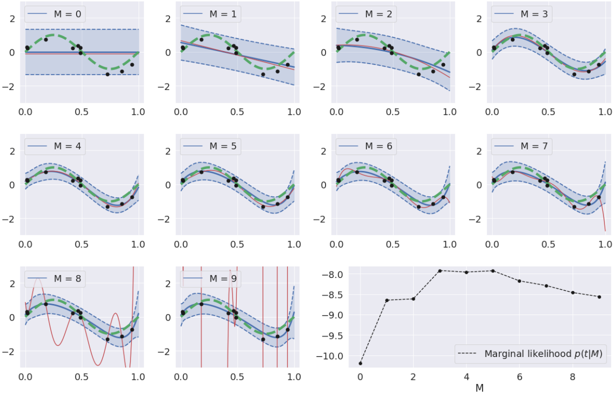
- Equivalent to maximizing the posterior $p(\alpha, \beta|\boldsymbol{y})$ assuming *flat priors* for α and β

$$p(\alpha, \beta|\boldsymbol{y}) \propto p(\boldsymbol{y}|\alpha, \beta)$$

- Equivalent to *poor man's Bayes* on hyperparameter level

Sinusoidal example revisited using the evidence approximation

- Also useful for model selection: $\alpha, \beta, M^* = \arg\max_{\alpha,\beta,M} p(\boldsymbol{y}|\alpha, \beta, M)$



- Implements "Occam's razor": choose the "simplest" model that explain the data
- Often works well for many models, but we should always assess the generalization error

A more general probabilistic perspective on supervised learning

Product rule
$p(\boldsymbol{a}, \boldsymbol{b}) = p(\boldsymbol{b}|\boldsymbol{a})p(\boldsymbol{a})$

Sum rule
$p(\boldsymbol{b}) = \int p(\boldsymbol{a}, \boldsymbol{b})\mathrm{d}\boldsymbol{a}$

Conditional
$p(\boldsymbol{a}|\boldsymbol{b}) = \frac{p(\boldsymbol{a},\boldsymbol{b})}{p(\boldsymbol{b})}$

Conditional independence
$p(\boldsymbol{a}, \boldsymbol{b}|\boldsymbol{c}) = p(\boldsymbol{a}|\boldsymbol{c})p(\boldsymbol{b}|\boldsymbol{c})$

Supervised learning: Given some data $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$, what can we say about a new test point $y^* = y(\boldsymbol{x}^*)$?

- Step 1: Formulate joint distribution for all variables of interests

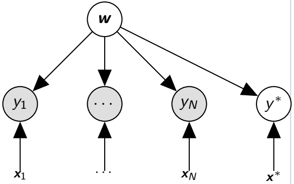
$$p(y^*, \boldsymbol{y}, \boldsymbol{w}) = p(y^*, \boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}) = p(y^*|\boldsymbol{w})p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})$$

- Step 2: Conditioned on the observed data $\boldsymbol{y}$

$$p(y^*, \boldsymbol{w}|\boldsymbol{y}) = \frac{p(y^*|\boldsymbol{w})p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y})}$$

- Step 3: Marginalize over all parameters $\boldsymbol{w}$ using sum rule

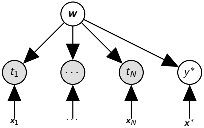
$$p(y^*|\boldsymbol{y}) = \int \frac{p(y^*|\boldsymbol{w})p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y})}\mathrm{d}\boldsymbol{w} = \int p(y^*|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{y})\mathrm{d}\boldsymbol{w}$$



A more complete graphical model

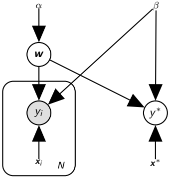
- Lazy, but common notation for Bayesian linear regression

$$p(y^*, \boldsymbol{y}, \boldsymbol{w}) = p(y^*|\boldsymbol{w})p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})$$



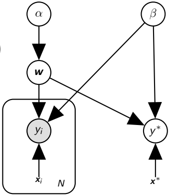
- More complete notation and corresponding graphical model

$$p(y^*, \boldsymbol{y}, \boldsymbol{w}|\alpha, \beta, \boldsymbol{X}, \boldsymbol{x}^*) = p(y^*|\boldsymbol{w}, \beta, \boldsymbol{x}^*)p(\boldsymbol{y}|\boldsymbol{w}, \beta, \boldsymbol{X})p(\boldsymbol{w}|\alpha)$$



- Fully Bayesian inference on hyperparameter level

$$p(y^*, \boldsymbol{y}, \boldsymbol{w}, \alpha, \beta|\boldsymbol{X}, \boldsymbol{x}^*) = p(y^*|\boldsymbol{w}, \beta, \boldsymbol{x}^*)p(\boldsymbol{y}|\boldsymbol{w}, \beta, \boldsymbol{X})p(\boldsymbol{w}|\alpha)p(\alpha)p(\beta)$$



Bayesian vs. classical statistics

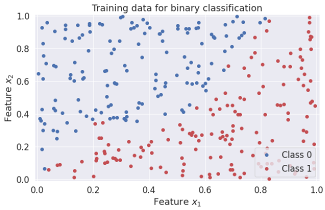
Bayesian vs. classical statistics

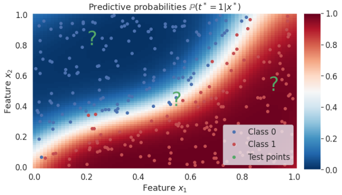
	Frequentist/classical	**Bayesian**
Probability interpretation	Long run frequencies	Degrees of belief
Parameters	Deterministic, but unknown Cannot make probabilistic statement about parameters	Random variables Probabilistic reasoning at levels: models, parameters and observations
	Point estimates	Probability distributions
Intepretation of intervals	*Confidence intervals* If the experiment is repeated infinitely many times, 95% of the intervals will contain the true population value	*Credibility intervals* The interval will contain the population value with 95% probability given the data
Sources of information	Data only	Data & prior knowledge
Computation	Often less computationally expensive	Often more computationally expensive

Bayesian methods for classification

Probabilistic approaches for classification

- Dataset $\mathcal{D} = \{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_N}, y_N)\}$
  - Input features: $\boldsymbol{x}_i \in \mathbb{R}^D$
  - Targets: $y_i \in \{0, 1\}$
- How to predict label for test point $\boldsymbol{x}^* \in \mathbb{R}^D$?
- *Predictive distributions*: what is the probability that $\boldsymbol{x}^*$ belong to class 1, i.e. $p(y^* = 1 | \mathcal{D}, \boldsymbol{x}^*)$?
- Two probabilistic approaches
  1. *Discriminative methods*
  2. *Generative methods*





Discriminative vs generative methods

- The *generative approach* models the *joint distribution* $p(\boldsymbol{x}_n, y_n)$, e.g. via Bayes rule

$$p(y_n = k|\boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n|y_n = k)p(y_n = k)}{p(\boldsymbol{x}_n)}$$

- Joint distribution over inputs $\boldsymbol{x}_n$ and labels y_n allow sampling (*generating*) from the model

$$\boldsymbol{x}^{(i)}, y^{(i)} \sim p(\boldsymbol{x}, y)$$

- Pros and cons for generative models
  - \+ Optimal if the assumptions are correct
  - \+ Can easily handle *missing data*
  - \+ Can reason about input data
  - \- Assumptions are often hard to get correct

- The *discriminative approach* models the *conditional distribution* $p(y_n|\boldsymbol{x}_n)$ directly by assuming some parametric form for the posterior

$$p(y_n|\boldsymbol{x}_n) = f(\boldsymbol{x}_n|\boldsymbol{w})$$

- The function $f(\boldsymbol{x}_n|\boldsymbol{w})$ can be based on a linear model, a neural network etc.

- Pros and cons for discriminative models
  - \+ Often superior when the assumptions for generative models are wrong
  - \+ Often better calibrated (compared to e.g. generative methods like Naive Bayes etc)
  - \+ Easy to make flexible
  - \- Difficult to handle missing data
  - \- Cannot reason about input data

Bayesian methods for classification: Generative modeling

The generative approach I

- Binary classification $y_n \in \{0, 1\}$

$$p(y_n = 1|\boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n|y_n = 1)p(y_n = 1)}{p(\boldsymbol{x}_n)}$$

- Terminology
  - *Class-conditional distribution* $p(\boldsymbol{x}_n|y_n)$
  - *Prior probabilities* $p(y_n = k) = \pi_k$
  - *Marginal data density* $p(\boldsymbol{x}_n)$

- The marginal density of $\boldsymbol{x}_n$ is a *mixture distribution* and is obtained using the *sum rule*

$$p(\boldsymbol{x}_n) = \sum_{k \in \{0,1\}} p(\boldsymbol{x}_n|y_n = k)p(y_n = k) = \pi_0 p(\boldsymbol{x}_n|y_n = 0) + \pi_1 p(\boldsymbol{x}_n|y_n = 1)$$

- Let's plug the result into Bayes' rule

The generative approach II

- The posterior of y_n given the input $\boldsymbol{x}_n$

$$p(y_n = 1|\boldsymbol{x}_n) = \frac{\pi_1 p(\boldsymbol{x}_n|y_n = 1)}{\pi_0 p(\boldsymbol{x}_n|y_n = 0) + \pi_1 p(\boldsymbol{x}_n|y_n = 1)}$$

- Divide by numerator

$$p(y_n = 1|\boldsymbol{x}_n) = \frac{1}{1 + \frac{\pi_0 p(\boldsymbol{x}_n|y_n=0)}{\pi_1 p(\boldsymbol{x}_n|y_n=1)}}$$

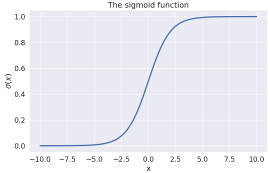
- Define

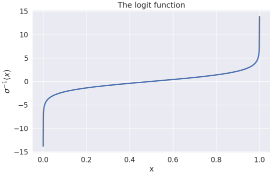
$$a = \ln \frac{\pi_1 p(\boldsymbol{x}_n|y_n = 1)}{\pi_0 p(\boldsymbol{x}_n|y_n = 0)}$$

- then

$$p(y_n = 1|\boldsymbol{x}_n) = \frac{1}{1 + \exp(-a)} = \sigma(a)$$

- Recall $\sigma(a)$ is the *logistic sigmoid* function and its inverse is called the *logit* function $a = \ln\left(\frac{\sigma}{1-\sigma}\right)$





The generative approach III: multi-class problems and softmax

- Assume we have K different classes, where $k = 1, \dots, K$

- Define a_k

$$a_k = \ln p(\boldsymbol{x}_n | y_n = k) p(y_n = k)$$

- Using similar line of reasoning for K classes

$$p(y_n = k | \boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n | y_n = k) P(y_n = k)}{\sum_{i=1}^{K} p(\boldsymbol{x}_n | y_n = i) p(y_n = i)} = \frac{\exp(a_k)}{\sum_{j=1}^{K} \exp(a_j)}$$

- The *normalized exponentials* is the known as the *softmax* function

Example: Gaussian class conditionals

- Binary classification, normal class conditionals with common variance in 1D

$$p(x_n|y_n = 0) = \mathcal{N}(x_n|\mu_0, \sigma^2)$$
$$p(x_n|y_n = 1) = \mathcal{N}(x_n|\mu_1, \sigma^2)$$

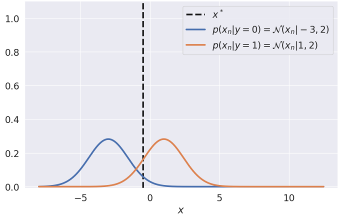
- We know

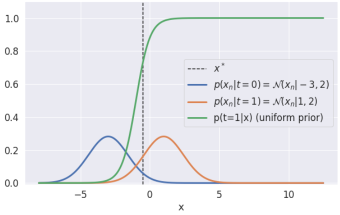
$$p(y_n = 1|\boldsymbol{x}_n) = \frac{1}{1 + \exp(-a)} = \sigma(a) = \sigma(w_0 + w_1 x_n)$$

- The quantity a has a simple expression

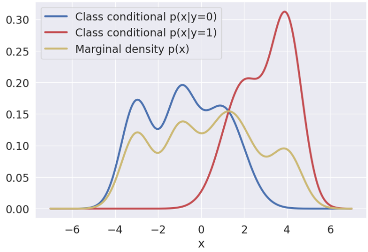
$$a = \ln \frac{\pi_1 p(\boldsymbol{x}_n|y_n = 1)}{\pi_0 p(\boldsymbol{x}_n|y_n = 0)} = \ln \frac{\pi_1 \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x_n - \mu_1)^2}{2\sigma^2})}{\pi_0 \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x_n - \mu_0)^2}{2\sigma^2})}$$
$$= \ln \frac{\pi_1}{\pi_0} - \frac{\mu_1^2 - \mu_0^2}{2\sigma^2} + \frac{\mu_1 - \mu_0}{\sigma^2} x_n$$
$$= w_0 + w_1 x_n$$

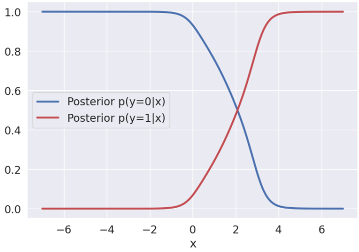
where $w_0 = \ln \frac{\pi_1}{\pi_0} - \frac{\mu_1^2 - \mu_0^2}{2\sigma^2}$ and $w_1 = \frac{\mu_1 - \mu_0}{\sigma^2}$





Example 2: More complex distributions





Bayesian methods for classification: Discriminative modelling

Discriminative modelling for binary classification

- In the generative model, we defined *priors* $p(y_n = 1) = \pi_1$ and a set of *class-conditionals* $p(\mathbf{x}_n | y_n = 1)$, applied Bayes rule and ended up with

$$a(\mathbf{x}_n) = \ln \frac{\pi_1 p(\mathbf{x}_n | y_n = 1)}{\pi_0 p(\mathbf{x}_n | y_n = 0)} = \ln \frac{p(y_n = 1 | \mathbf{x}_n)}{p(y_n = 0 | \mathbf{x}_n)}$$

  and

$$p(y_n = 1 | \mathbf{x}_n) = \frac{1}{1 + \exp(-a)} = \sigma(a(\mathbf{x}_n))$$

- The distributional assumptions gave the specific functional form for $a(\mathbf{x})$, but in *discriminative modelling*, we directly assume a functional form for $a(\mathbf{x})$

- Example: logistic regression

$$p(y_n = 1 | \mathbf{x}_n, \mathbf{w}) = \frac{1}{1 + \exp(-a)} = \sigma(\phi(\mathbf{x}_n)^T \mathbf{w})$$

- We model each observation with a *Bernoulli* distribution with probability $\sigma(\phi(\mathbf{x}_n)^T \mathbf{w})$

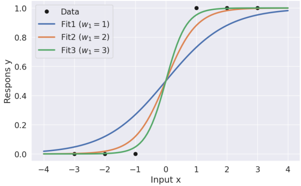
- We *estimate* $\mathbf{w}$ using maximum likelihood, MAP or Bayesian inference with the likelihood function

$$p(\mathbf{y} | \mathbf{w}, \mathbf{X}) = \prod_{n=1}^{N} \sigma(\phi(\mathbf{x}_n)^T \mathbf{w})^{y_n} \left(1 - \sigma(\phi(\mathbf{x}_n)^T \mathbf{w})\right)^{1 - y_n}$$

Maximum likelihood estimator for logistic regression: Quiz

Set-up

- Consider a simple dataset with $N = 6$, $\phi(x) = x$



- Logistic regression likelihood

$$p(\boldsymbol{y}|w_1) = \prod_{n=1}^{N} \sigma(w_1 x_n)^{y_n} \left(1 - \sigma(w_1 x_n)\right)^{1-y_n}$$

- One parameter: w_1

Questions (5mins)

- Spend 5 minutes DTU Learn quiz: "Lecture 4: Logistic regression"

Bayesian logistic regression

Bayesian supervised learning

- For conjugate models, both the posterior and predictive distributions can be computed analytically

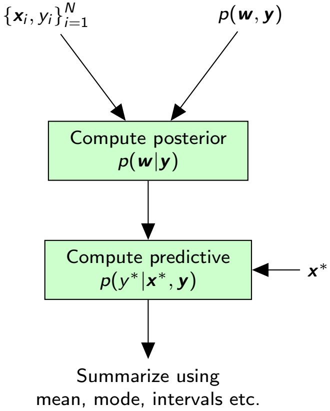
- The real strength of the Bayesian framework lies in the modelling flexibility

- ... but for even rather simple models like *Bayesian logistic regression*, we cannot compute

  1. the posterior distribution
  2. the predictive distribution

- Today we will see how to use the *Laplace approximation* to approximate the posterior

- ... and we will discuss different strategies to evaluate the predictive distribution



Bayesian logistic regression

- Likelihood for logistic regression

$$p(\boldsymbol{y}|\boldsymbol{w}) = \prod_{n=1}^{N} \sigma(\phi(\boldsymbol{x}_n)^T \boldsymbol{w})^{y_n} \left(1 - \sigma(\phi(\boldsymbol{x}_n)^T \boldsymbol{w})\right)^{1-y_n}$$

- Let's impose a prior distribution on the weights $\boldsymbol{w}$ assuming the individual weights w_i are *independent and identically distributed (i.i.d)* a priori

$$p(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{0}, \alpha^{-1}\boldsymbol{I}) = \prod_{i=1}^{D} \mathcal{N}(w_i|0, \alpha^{-1})$$

- The posterior follows from Bayes' theorem

$$p(\boldsymbol{w}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y})}$$

- Let's calculate the posterior mean

$$\mathbb{E}_{p(\boldsymbol{w}|\boldsymbol{y})}[\boldsymbol{w}] = \int \boldsymbol{w} p(\boldsymbol{w}|\boldsymbol{y}) \mathrm{d}\boldsymbol{w} = \frac{1}{p(\boldsymbol{y})} \int \boldsymbol{w} p(\boldsymbol{y}|\boldsymbol{w}) p(\boldsymbol{w}) \mathrm{d}\boldsymbol{w}$$

- Clearly, we need $p(\boldsymbol{y})$ as well

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{w}) p(\boldsymbol{w}) \mathrm{d}\boldsymbol{w}$$

Bayesian logistic regression II

- General problem: we *cannot* compute the posterior mean analytically

$$\mathbb{E}_{p(\boldsymbol{w}|\boldsymbol{y})}[\boldsymbol{w}] = \frac{1}{p(\boldsymbol{y})} \int \boldsymbol{w} p(\boldsymbol{y}|\boldsymbol{w}) p(\boldsymbol{w}) \mathrm{d}\boldsymbol{w}$$

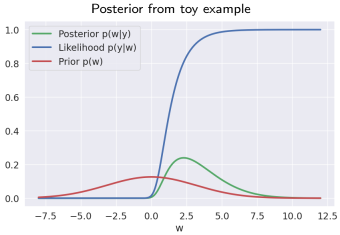
- This *roadblock* occurs for almost any other interesting posterior summary

- The posterior distribution and the marginal likelihood for most Bayesian models is *analytically intractable*

- Posterior distribution from our toy example is asymmetric, but almost resembles a Gaussian

- *Bernstein von Mises theorem*
  Assuming certain regularity conditions, the posterior distribution of a parametric model becomes more and more Gaussian as N increases

- Let's approximate $p(\boldsymbol{w}|\boldsymbol{y})$ with a Gaussian!



Laplace approximations

Laplace approximations I

- The *Laplace approximation* is a method for approximating intractable probability densities

- Assume we have a posterior distribution of interest $p(\boldsymbol{w}|\boldsymbol{y})$

$$p(\boldsymbol{w}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y})} = \frac{1}{Z}f(\boldsymbol{w}) \approx \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}, \boldsymbol{S})$$

- The log density for Gaussians is quadractic wrt. $\boldsymbol{w}$

$$\ln \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}, \boldsymbol{S}) = -\frac{D}{2}\ln(2\pi) - \frac{1}{2}\log|\boldsymbol{S}| - \frac{1}{2}(\boldsymbol{w} - \boldsymbol{m})^T\boldsymbol{S}^{-1}(\boldsymbol{w} - \boldsymbol{m})$$

- Let's make a second order Taylor expansion of $f(\boldsymbol{w})$ around the mode $\boldsymbol{w}_{\text{MAP}}$

$$\ln f(\boldsymbol{w}) \approx \ln f(\boldsymbol{w}_{\text{MAP}}) - \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})^T\boldsymbol{A}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}}),$$

  where $\boldsymbol{A}$ is the Hessian at the mode, i.e. $\boldsymbol{A} = -\nabla\nabla \ln f(\boldsymbol{w})\big|_{\boldsymbol{w}=\boldsymbol{w}_{\text{MAP}}}$

- The Laplace approximation is defined

$$q(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{w}_{\text{MAP}}, \boldsymbol{A}^{-1})$$

- That is, we approximate the posterior mean using the MAP and the posterior covariance using the curvature at the MAP solution.

Laplace approximation II

- Suppose we want to approximate $p(\boldsymbol{w}|\boldsymbol{y})$ using the Laplace approximation

$$p(\boldsymbol{w}|\boldsymbol{y}) \approx q(\boldsymbol{w}) = \mathcal{N}(\boldsymbol{w}|\boldsymbol{w}_{\text{MAP}}, \boldsymbol{A}^{-1})$$

- Computational steps

  1. Locate the mode of $p(\boldsymbol{w}|\boldsymbol{y})$

$$\boldsymbol{w}_{\text{MAP}} = \arg\max_{\boldsymbol{w}} p(\boldsymbol{w}|\boldsymbol{y}) = \arg\max_{\boldsymbol{w}} p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})$$

  2. Evaluate the Hessian at $\boldsymbol{w}_{\text{MAP}}$

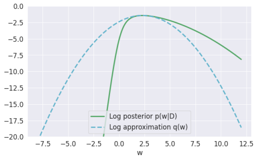
$$\boldsymbol{A} = -\nabla\nabla \ln p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})\Big|_{\boldsymbol{w}=\boldsymbol{w}_{\text{MAP}}}$$

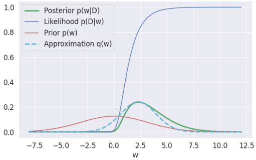
- Advantages

  1. Simple and well-understood
  2. Very fast to compute
  3. Gives good results for many problems

- Limitations

  1. Only applies to continuous parameters
  2. Gaussian (symmetric distribution, thin tails)
  3. Only capture local properties of $p(\boldsymbol{w}|\boldsymbol{y})$ near $\boldsymbol{w}_{\text{MAP}}$
  4. Does not work for hierarchical models in general





Laplace approximations III: Approximating the marginal likelihood

- Our second order Taylor approximation for $\ln f(\boldsymbol{w})$

$$\ln f(\boldsymbol{w}) \approx \ln f(\boldsymbol{w}_{\text{MAP}}) - \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})^T \boldsymbol{A}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}}),$$

- By assumption

$$p(\boldsymbol{w}|\boldsymbol{y}) = \frac{1}{Z} f(\boldsymbol{w}) \qquad \Rightarrow \qquad Z = \int f(\boldsymbol{w}) \mathrm{d}\boldsymbol{w}$$

- Plugging in the approximation for $\ln f(\boldsymbol{w})$

$$\begin{aligned} Z &= \int f(\boldsymbol{w}) \mathrm{d}\boldsymbol{w} \\ &\approx f(\boldsymbol{w}_{\text{MAP}}) \int \exp\left(-\frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})^T \boldsymbol{A}(\boldsymbol{w} - \boldsymbol{w}_{\text{MAP}})\right) \\ &= f(\boldsymbol{w}_{\text{MAP}}) \frac{(2\pi)^{\frac{D}{2}}}{|\boldsymbol{A}|^{\frac{1}{2}}} \end{aligned}$$

- Using $f(\boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w})$, our approximation for $p(\boldsymbol{y})$ becomes

$$\ln p(\boldsymbol{y}) \approx \ln p(\boldsymbol{y}|\boldsymbol{w}_{\text{MAP}}) + \ln p(\boldsymbol{w}_{\text{MAP}}) + \frac{D}{2} \ln(2\pi) - \frac{1}{2} \ln |\boldsymbol{A}|$$

- Very useful for model selection, parameter tuning etc

The posterior predictive distribution

How to make predictions?

- For classification, we need the predictive distribution for a new input $\boldsymbol{x}^*$. The likelihood for a input data point $\boldsymbol{x}*$ is

$$p(y^* = 1|\boldsymbol{w}, \boldsymbol{x}^*) = \sigma(\phi(\boldsymbol{x}^*)^T \boldsymbol{w})$$

- As always, we want to take the posterior uncertainty into account using the sum rule

$$\begin{aligned}
p(y^* = 1|\boldsymbol{y}, \boldsymbol{x}^*) &= \int p(y^* = 1|\boldsymbol{x}^*, \boldsymbol{w}) p(\boldsymbol{w}|\boldsymbol{y}) \mathrm{d}\boldsymbol{w} \\
&\approx \int p(y^* = 1|\boldsymbol{x}^*, \boldsymbol{w}) q(\boldsymbol{w}) \mathrm{d}\boldsymbol{w} \\
&= \int \sigma(\phi(\boldsymbol{x}^*)^T \boldsymbol{w}) \mathcal{N}(\boldsymbol{w}|\boldsymbol{m}, \boldsymbol{S}) \mathrm{d}\boldsymbol{w} \\
&= \int \sigma(f) \mathcal{N}(f|\mu, \sigma^2) \mathrm{d}f
\end{aligned}$$

  where

$$\mu = \phi(\boldsymbol{x}^*)^T \boldsymbol{m} \qquad\qquad \sigma^2 = \phi(\boldsymbol{x}^*)^T \boldsymbol{S} \phi(\boldsymbol{x}^*)$$

- The good news: we only have to calculate 1D integrals to make predictions

- The bad news: the integral does not have analytical solution

Evaluating predictive distributions for logistic regression

How does uncertainty in f affect the distribution of $\sigma(f)$?

$$p(y^* = 1|\mathbf{y}, \mathbf{x}^*) = \int \sigma(f)\mathcal{N}(f|\mu, \sigma^2)\mathrm{d}f$$

- General strategies for evaluating this integral

  1. Monte Carlo methods (sampling)

  $$p(y^* = 1|\mathbf{y}, \mathbf{x}^*) \approx \frac{1}{S}\sum_{i=1}^{S} \sigma\left(f^{(i)}\right) \qquad \text{for} \qquad f^{(i)} \sim \mathcal{N}(f|\mu, \sigma^2)$$

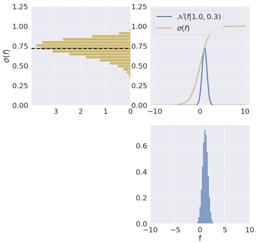
  2. Numerical integration (Gauss-Hermite integration)

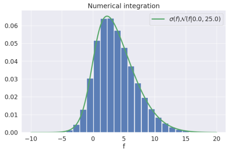
  $$p(y^* = 1|\mathbf{y}, \mathbf{x}^*) \approx \frac{1}{\sqrt{\pi}}\sum_{i=1}^{S} w_i h(\sqrt{2}\sigma x_i + \mu)$$

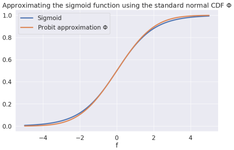
  3. Probit approximation

  $$\sigma(y) \approx \Phi\left(y\sqrt{\frac{\pi}{8}}\right)$$

  where Φ is the CDF of the standard normal







Let's zoom out and summarize

- We introduced *logistic regression* as a *discriminative* approach for binary classification
- We saw to use the *Laplace approximation* to approximate the *posterior* of the weights
- We briefly discussed three strategies to compute *the predictive distribution*
  1. Sampling
  2. Numerical integration
  3. Probit approximation