Machine Learning and Pattern Recognition - Week 9 - Bayesian Logistic Regression & the Laplace Approximation

Antonio León Villares

November 2022

Contents

Based on the online notes here.

1 Recap: Bayesian Linear Regression

*In **Bayesian Linear Regression**, we have:*

- *a **prior** distribution on **weights**:*

$$P(\underline{w})$$

- *a **likelihood**, of seeing some data $\mathcal{D}$, given a certain **weight** setting:*

$$P(\mathcal{D} \mid \underline{w})$$

- *using this, we can compute a **posterior** distribution, which tells us how **weights** should look, given that we have observed data $\mathcal{D}$:*

$$P(\underline{w} \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \underline{w})P(\underline{w})}{P(\mathcal{D})}$$

*If we use a **normally** distributed **prior** and **likelihood**, then the **posterior** $P(\underline{w} \mid \mathcal{D})$ can be easily sampled from.*

---

*If we want to use our **Bayesian Linear Regression** to actually make predictions, we need to define a **posterior predictive distribution**:*

$$P(y \mid \underline{x}, \mathcal{D})$$

*which predicts a value y, given some observation $\underline{x}$ and all the data seen by the model previously $\mathcal{D}$. To determine the **posterior predictive distribution**, we apply the **sum** and **product** rules, which allow us to condition on **weights**, and thus, allow us to utilise our **posterior** to make predictions:*

$$P(y \mid \underline{x}, \mathcal{D}) = \int P(y, \underline{w} \mid \underline{x}, \mathcal{D}) d\underline{w} = \int P(y \mid \underline{x}, \underline{w}) P(\underline{w} \mid \mathcal{D}) d\underline{w}$$

*where $P(y \mid \underline{x}, \underline{w})$ is the **predictive distribution**, which gives the probability of observing y given our weights $\underline{w}$ and an input $\underline{x}$.*
*In practice, sampling from the **posterior** to get **weights**, or computing the **posterior predictive distribution** are highly **non-trivial** tasks: for instance, we might not even have a closed-form, parametrised distribution. On the other hand, if we have a normal **posterior** and **predictive model**, then we can derive a new normal distribution for the **posterior predictive distribution**.*

2 Bayesian Logistic Regression

2.1 Recap: Logistic Regression

*The **Logistic Regression** model is used to perform **binary classification**:*

$$P(y = 1 \mid \underline{x}, \underline{w}) = \sigma(\underline{w}^T \underline{x} + b) = \frac{1}{1 + \exp(-(\underline{w}^T \underline{x} + b))}$$

*To determine $\underline{w}$, we can use a **Maximum Likelihood Estimation**: given an input matrix X and observations $\underline{y}$:*

$$P(\underline{y} \mid X, \underline{w}) = \prod \sigma(z^{(n)} \underline{w}^T \underline{x}^{(n)})$$

where:

$$z^{(n)} = 2y^{(n)} - 1$$

*and $y^{(n)}$ is a **binary** feature.*
*In practice, we typically **minimise** the **negative log likelihood**, and add a **regularisation parameter**:*

$$\underline{w}^* = arg \max_{\underline{w}} [\log P(\underline{y} \mid X, \underline{w}) - \lambda \underline{w}^T \underline{w}]$$

2.2 Probabilistic Logistic Regression

- **What is the posterior distribution for a Bayesian Logistic Regression model?**
  - as with **Bayesian Linear Regression**, we will have:
    - a **prior** over weights $\underline{w}$:

      $$P(\underline{w})$$

    - a **likelihood** of observing data given weights. This will be our **Logistic Regression**:

      $$P(\mathcal{D} \mid \underline{w}) = \sigma(\underline{w}^T \underline{x} + b)$$

  - using this, we obtain a **posterior**:

    $$P(\underline{w} \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \underline{w}) P(\underline{w})}{P(\mathcal{D})}$$

    where we can compute the **marginal likelihood** $P(\mathcal{D})$ by **marginalisation**:

    $$P(\mathcal{D}) = \int P(\mathcal{D} \mid \underline{w}) P(\underline{w}) d\underline{w}$$

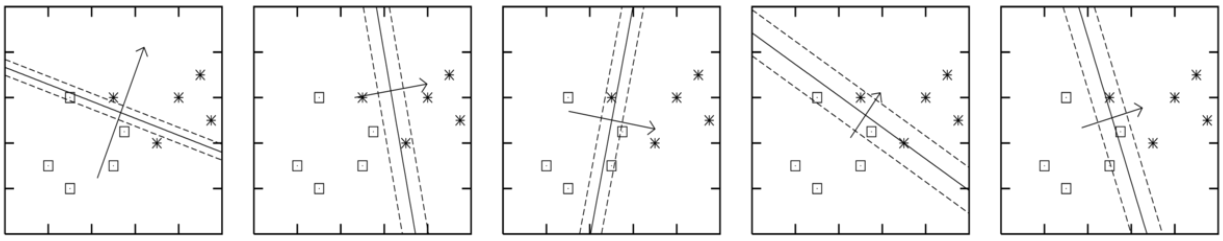

Figure 1: Sample weights taken from a Bayesian Logistic Regression model. Each figure displays the decision boundary for the model ($\sigma(\underline{w}^T\underline{x} + b) = 0.5$). $\underline{w}$ is **perpendicular** to the decision boundary.

  - notice, we have sampled very different weights, all of which give **reasonable** decision boundaries

- **How can we use Bayesian Logistic Regression for classification?**
  - we need to compute the **posterior predictive distribution**
  - analogously to **Bayesian Linear Regression**:

$$P(y \mid \underline{x}, \mathcal{D}) = \int P(y, \underline{w} \mid \underline{x}, \mathcal{D}) d\underline{w} = \int P(y \mid \underline{x}, \underline{w}) P(\underline{w} \mid \mathcal{D}) d\underline{w}$$

  - this is a **weighted integral**: we are averaging all possible models $P(y \mid \underline{x}, \underline{w}) = \sigma((2y-1)[\underline{w}^T\underline{x}+b])$ given how **plausible** the parameters for the model are $P(\underline{w} \mid \mathcal{D})$

- **How do the contours of Bayesian Logistic Regression differ from those of standard Logistic Regression?**
  - with **standard** Logistic Regression, we have a **fixed weight**, so the **contours** will be **parallel** to the **boundary**
  - with **bayesian** Logistic Regression, we consider **all** possible models, so the contours need not be parallel or linear
  - this is because **different predictors** have different confidence levels when far away from the data

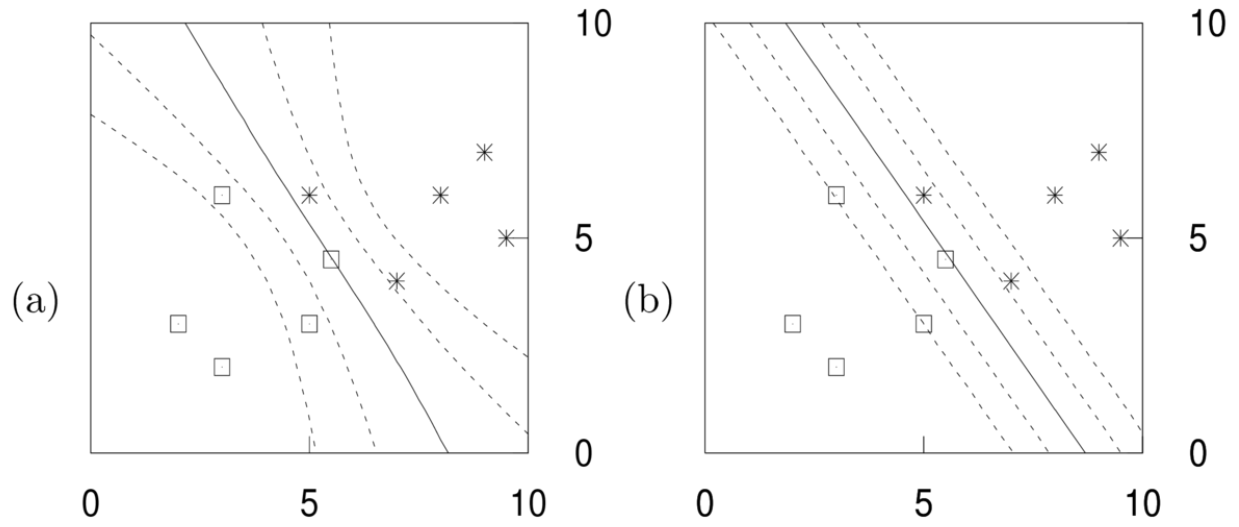

Figure 2: We plot contours for $P(y = 1 \mid \underline{x}, \mathcal{D})$. To the left, contours for **Bayesian Logistic Regression**: getting away from the data warps the contours, since in regions away from data the different predictors wll disagree. To the right, contours for a fixed weight **Logistic Regression**: getting away from the data doesn't affect the uncertainty of the predictions.

- **What is MAP estimation?**
  - stands for **maximum a posteriori estimation**
  - MLE seeks to **maximise** the **likelihood** of the **data**, given the **model**:

$$\underset{\underline{w}}{arg\ max}\ P(\mathcal{D} \mid \underline{w})$$

  - MAP seeks to **maximise** the **posterior** (i.e maximise the probability of the parameters, given the data; also known as finding the **mode** of the distribution)

$$\underset{\underline{w}}{arg\ max}\ P(\underline{w} \mid \mathcal{D}) = \underset{\underline{w}}{arg\ max}\ P(\mathcal{D} \mid \underline{w})P(\underline{w})$$

    (since the **marginal likelihood** $P(\mathcal{D})$ is constant it doesn't affect the max)

- **What is the result of fitting Logistic Regression weights using MAP?**
  - we can apply MAP to the **negative log-likelihood** to find the **weights** for our **Bayesian Logistic Regression** model:

$$\begin{aligned}\underline{w}^* &= -\underset{\underline{w}}{arg\ min}\ \log P(\underline{w} \mid \mathcal{D}) \\ &= -\underset{\underline{w}}{arg\ min}\ [\log P(\mathcal{D} \mid \underline{w}) + \log P(\underline{w})]\end{aligned}$$

  - if we have a **Gaussian prior**:

$$\underline{w} \sim \mathcal{N}(0, \sigma_w^2 \mathbb{I})$$

    then:

$$\log P(\underline{w}) = -\frac{1}{2\sigma_w^2}\underline{w}^T\underline{w}$$

– hence, our MAP estimation becomes:

$$\underline{w}^* = -arg\ \underset{\underline{w}}{min}\ \left[\log P(\mathcal{D} \mid \underline{w}) - \frac{1}{2\sigma_w^2}\underline{w}^T\underline{w}\right]$$

– this is precisely the negative log likelihood with L^2 regularisation of the weights!

- **Is MAP a Bayesian procedure?**
  - no, since **Bayesian** methods don't fix an unknwon parameter vector (in our case $\underline{w}$)
  - we can think of MAP as a**crude** approximation for a Bayesian procedure: as we saw above, Bayesian Logistic Regression has very different contours than the contours obtained by a Logistic Regression model fitted with MAP

3 The Laplace Approximation

3.1 The Posterior for Bayesian Logistic Regression

- **How tractable is the computation of the predicitve posterior for Bayesian Logistic Regression?**
  - in general, evaluating:

$$P(y \mid \underline{x}, \mathcal{D}) = \int P(y \mid \underline{x}, \underline{w}) P(\underline{w} \mid \mathcal{D}) d\underline{w}$$

    in **closed-form** is **intractable**
  - even if we use the trick:

$$P(y \mid \underline{x}, \mathcal{D}) = \mathbb{E}_{\underline{w} \sim P(\underline{w} \mid \mathcal{D})}[P(y \mid \underline{x}, \underline{w})]$$

    and approximate using **Monte-Carlo Estimation**:

$$\begin{aligned}\mathbb{E}_{\underline{w} \sim P(\underline{w} \mid \mathcal{D})}[P(y \mid \underline{x}, \underline{w})] &\approx \frac{1}{K}\sum_{k=1}^{K} P(y \mid \underline{x}, \underline{w}^{(k)}), \qquad \underline{w}^{(k)} \sim P(\underline{w} \mid \mathcal{D}) \\ &= \frac{1}{K}\sum_{k=1}^{K} \sigma(\underline{w}^{(k)T}\underline{x})\end{aligned}$$

    we still need to sample $\underline{w}$ from our **posterior**:

$$P(\underline{w} \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \underline{w}) P(\underline{w})}{P(\mathcal{D})}$$

    which is problematic for 2 reasons:

    1. computing the **marginal likelihood**:

$$P(\mathcal{D}) = \int P(\mathcal{D} \mid \underline{w}) P(\underline{w}) d\underline{w}$$

       is again, highly non-trivial
    2. even if we choose to ignore it (since it is a constant), how would we sample from the likelihood $P(\mathcal{D} \mid \underline{w})$ (for example, if its a Logistic Regression, or some other model which isn't as nice as some standard distribution, like normal/binomial/bernoulli/etc...)
  - methods such as **Markov Chain Monte Carlo** can be used to **sample** from **posteriors** of models like **logistic regression** and **neural networks**; however, this is beyond the scope of the course

- **What alternatives are available for computing the posterior of the Bayesian Logistic Regression?**

  1. Use a parametric distribution (i.e normal, bernoulli) to define the **likelihood** and **priors**
  2. Make approximations:
     - reduce the non-parametric distribution into a **simpler** one (i.e Gaussian)
     - approximate by matching the **moments** (i.e mean, variance, etc...) of the distribution
     - use MAP for approximating the distribution

- **When will the posterior of a Bayesian Logistic Regression not be Gaussian?**

  - say we observe a datapoint at $x = -20$, with label $y = 1$; we also know that we have a **normal** prior:

    $$P(\underline{w}) \propto \mathcal{N}(\underline{w}; 0, 1)$$

    and a **logistic** likelihood, with bias 10
  - if we compute the **posterior** for the single observation, the posterior is:

    $$P(\underline{w} \mid \mathcal{D}) \propto \mathcal{N}(w; 0, 1)\sigma(10 - 20w)$$

    which we can plot as a function of w:

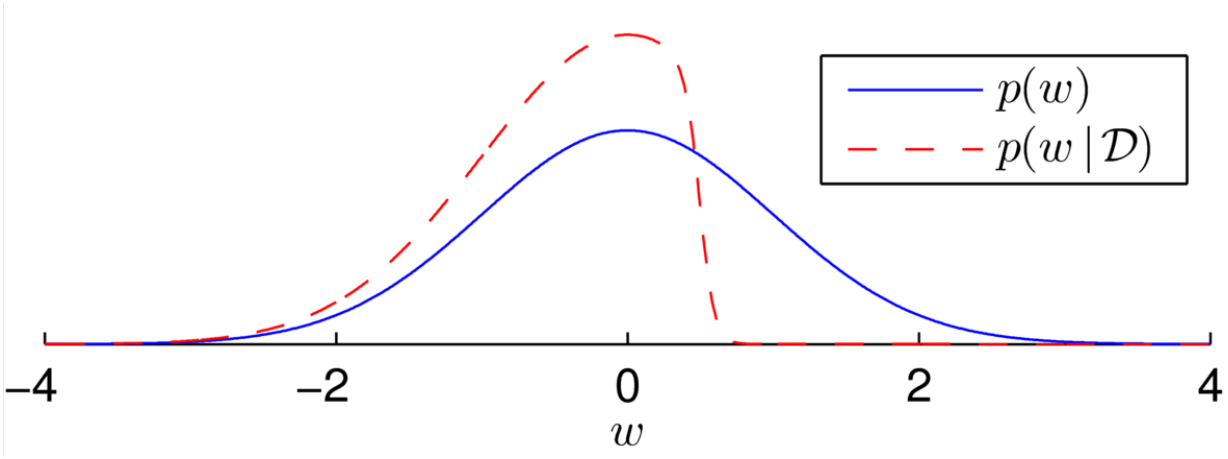

Figure 3: Since the sigmoid is basically 0 when $w > 0.7$, it "slices" a section of the positive region of the prior. After normalising, we obtain the above posterior distribution.

  - the posterior is **not** symmetric, and thus, can't be **Gaussian**
  - its believe is now that the weight must be **negative**: otherwise, $x = -20$ should've had the label $y = 0$

- **When will the posterior of a Bayesian Logistic Regression look Gaussian?**

  - if we have many observations, the posterior will begin to look Gaussian
  - for instance, if we sample 500 labels $z^{(n)}$ from a logsitic regression model with no bias, and $x^{(n)} \sim \mathcal{N}(0, 100)$, and then build a Probabilistic Logistic Regression model with:

    $$P(\underline{w}) \propto \mathcal{N}(\underline{w}; 0, 1)$$

we get posterior:

$$P(\underline{w} \mid \mathcal{D}) \propto \mathcal{N}(\underline{w}; 0, 1) \prod_{n=1}^{500} \sigma(w x^{(n)} z^{(n)}$$

which will *look* Gaussian:

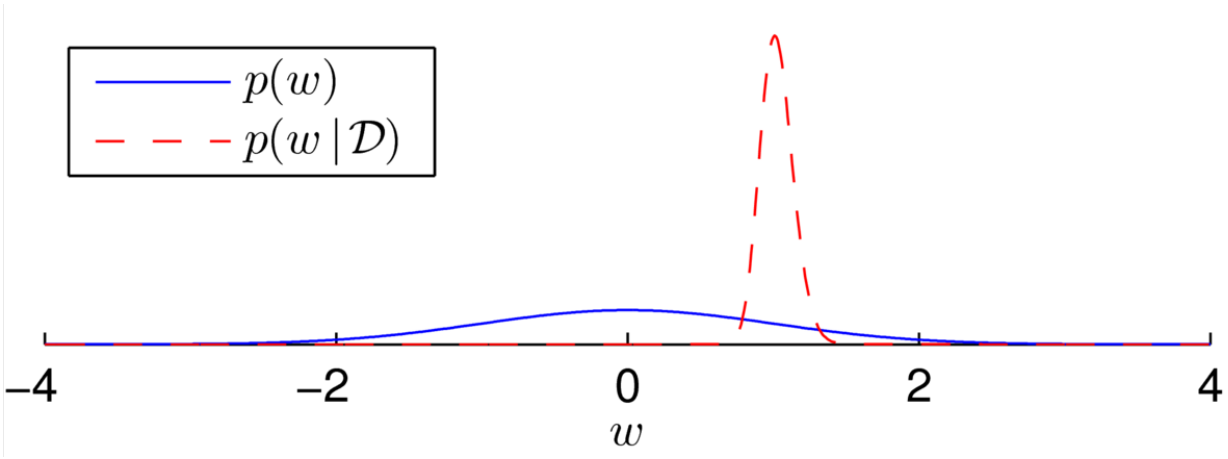

Figure 4: The posterior over many samples looks (but isn't) Gaussian. Notice, it has high confidence that $w = 1$, as expected (since $w = 1$ is what we used to generate the data in the first place)

3.2 Computing the Laplace Approximation

- **What is the Laplace approximation?**
  - a way of **approximating** a non-parametric **distribution** by using a **Gaussian**
  - the resulting **Gaussian** will:
    - have the same **mode**
    - have the same **curvature** at the location of the mode
- **What is the Laplace approximation for the posterior Bayesian Logistic Regression model?**
  - define the **energy** as the **negative log probability** of the weight distribution for a Bayesian Logistic Regression model:

    $$E(\underline{w}) = -\log P(\underline{w}, \mathcal{D})$$

    (this is defined up to the normalisation constant)
  - this tells us that $P(\underline{w} \mid \mathcal{D})$ is a distribution of the form:

    $$\frac{\exp(-E(\underline{w}))}{K}$$

    where K is some normalisation constant
  - notice, finding weights by minimising E is equivalent to performing a MAP approximation for the

(log) posterior:

$$\begin{aligned}\underline{w}^* &= -arg\ \underset{\underline{w}}{min}\ \log P(\underline{w} \mid \mathcal{D}) \\ &= -arg\ \underset{\underline{w}}{min}\ \log\left(\frac{P(\underline{w}, \mathcal{D})}{P(\mathcal{D})}\right) \\ &= -arg\ \underset{\underline{w}}{min}\ \log\left(P(\underline{w}, \mathcal{D})\right) \\ &= -arg\ \underset{\underline{w}}{min}\ E(\underline{w})\end{aligned}$$

where we have used monotonicity of the logarithm

- at the **minimum** of E, we have that $\nabla_{\underline{w}} E(\underline{w}^*) = \underline{0}$; moreover, the **curvature** of E at $\underline{w}^*$ is determined by the **Hessian**:

$$H_{ij}(\underline{w}^*) = \frac{\partial^2 E}{\partial w_i \partial w_j}(\underline{w}^*)$$

which tells us how quickly E changes at the minimum in a given direction

- now, the **energy** for **multivariate Gaussian** with mean μ and covariance Σ (up to the normalising constant) is:

$$E_{\mathcal{N}}(\underline{w}) = \frac{1}{2}(\underline{w} - \underline{\mu})^T \Sigma^{-1}(\underline{w} - \underline{\mu})$$

- since Σ is positive definite, $E_{\setminus}(\underline{w}) \geq 0$, and clearly:

$$\underline{w}^* = \underline{\mu}$$

minimises the energy

- we now determine the curvature. In one dimension:

$$E_{\mathcal{N}}(w) = \frac{(2-\mu)^2}{2\sigma^2}$$

so:

$$\frac{d^2 E}{d^w} = \frac{d}{dw}\left(\frac{2(w-\mu)}{2\sigma^2}\right) = \frac{1}{\sigma^2}$$

so by analogy, the **Hessian** H will be:

$$H = \Sigma^{-1}$$

- hence, the **Laplace Approximation**:

$$P(\underline{w} \mid \mathcal{D}) \approx \mathcal{N}(\underline{w}; \underline{w}^*, H^{-1})$$

will have the same **mode** and **curvature** as the posterior $P(\underline{w} \mid \mathcal{D})$

- **What is the Laplace approximation for the normalisation constant of the posterior Bayesian Logistic Regression model?**
  - we can use the **Laplace Approximation** to approximate:

$$P(\mathcal{D})$$

  - we have:

$$\begin{aligned}P(\underline{w} \mid \mathcal{D}) &= \frac{P(\underline{w}, \mathcal{D})}{P(\mathcal{D})} \\ &\approx \mathcal{N}(\underline{w}; \underline{w}^*, H^{-1}) \\ &= \frac{|H|^{1/2}}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}(\underline{w} - \underline{w}^*)^T H (\underline{w} - \underline{w}^*)\right)\end{aligned}$$

– at the mode $\underline{w}^*$ we get that:

$$\frac{P(\underline{w}^*, \mathcal{D})}{P(\mathcal{D})} \approx \frac{|H|^{1/2}}{(2\pi)^{D/2}}$$

so we approximate $P(\mathcal{D})$ by:

$$P(\mathcal{D}) \approx \frac{P(\underline{w}^*, \mathcal{D})(2\pi)^{D/2}}{|H|^{1/2}} = P(\underline{w}^*, \mathcal{D})|2\pi H^{-1}|^{1/2}$$

3.3 Predicting with Bayesian Logistic Regression

- **How can we use the Laplace Approximation to classify using Bayesian Logistic Regression?**
  - we want to be able to compute:

$$P(y \mid \underline{x}, \mathcal{D}) = \int P(y \mid \underline{x}, \underline{w}) = P(\underline{w} \mid \mathcal{D}) d\underline{w}$$

(this is technically all conditioned on $\mathcal{M}$, our model choices)

  - using the **Laplace Approximation**, this is:

$$P(y = 1 \mid \underline{x}, \mathcal{D}) \approx \int \sigma(\underline{w}^T \underline{x}) \mathcal{N}(\underline{w}; \underline{w}^*, H^{-1}) d\underline{w}$$

which can be written as an **expectation**:

$$P(y = 1 \mid \underline{x}, \mathcal{D}) \approx \mathbb{E}_{\underline{w} \sim \mathcal{N}(\underline{w}; \underline{w}^*, H^{-1})}[\sigma(\underline{w}^T \underline{x})]$$

  - now, notice since $\underline{w}$ is normally distributed, $a = \underline{w}^T \underline{x}$ is a **scalar** which should also be normally distributed
  - in particular:

$$\mathbb{E}[\underline{w}^T \underline{x}] = (\mathbb{E}[\underline{w}])^T \underline{x} = \underline{w}^{*T} \underline{x}$$

$$\begin{aligned} Var(a) &= \mathbb{E}[(\underline{w}^T \underline{x})^2] - (\underline{w}^{*T} \underline{x})^2 \\ &= \mathbb{E}[\underline{x}^T \underline{w} \underline{w}^T \underline{x}] - x^T \underline{w}^* \underline{w}^{*T} \underline{x} \\ &= \underline{x}^T \left(\mathbb{E}[\underline{w} \underline{w}^T] - \underline{w}^* \underline{w}^{*T} \right) \underline{x} \\ &= \underline{x}^T Cov(\underline{w}) \underline{x} \\ &= \underline{x}^T H^{-1} \underline{x} \end{aligned}$$

so we have that:

$$p(a) = \mathcal{N}(a; \underline{w}^{*T} \underline{x}, \underline{x}^T H^{-1} \underline{x})$$

  - hence, our expectation changes to:

$$\begin{aligned} P(y = 1 \mid \underline{x}, \mathcal{D}) &\approx \mathbb{E}_{a \sim \mathcal{N}(a; \underline{w}^{*T} \underline{x}, \underline{x}^T H^{-1} \underline{x})}[\sigma(a)] \\ &= \int \sigma(a) \mathcal{N}(a; \underline{w}^{*T} \underline{x}, \underline{x}^T H^{-1} \underline{x}) da \end{aligned}$$

which is now a **one-dimensional** integral

  - one dimensional integrals are easy to compute numerically, and can be done very easily

- **What is the probit approximation?**

- we can approximate the whole **predictive posterior** by using the **probit approixiamation**

$$P(y=1 \mid \underline{x}, \mathcal{D}) \approx \sigma(\kappa \underline{w}^{*T} \underline{x}), \qquad \kappa = \frac{1}{\sqrt{1 + \frac{\pi}{8} \underline{x}^T H^{-1} \underline{x}}}$$

- this has the benefit of:
  - being **quick** to evaluate
  - being **interpretable**
  - being a **closed-form** expression
- this uses the MAP weights, and scales the activation down if there is uncertainty (so predictions will be less confident away from data)

3.4 Evaluating the Laplace Approximation

- **When is the Laplace approximation reasonable?**
  - assuming $E(\underline{w})$ is well behaved, we can expand its **Taylor Series** about the mode $\underline{w}^*$
  - in one-dimension:

$$E(w) \approx E(w^*) + E'(w^*)(w - w^*) + \frac{1}{2} E''(w^*)(w - w^*)^2 + \mathcal{O}(\delta^3) \approx E(w^*) + \frac{1}{2} H (w - w^*)^2$$

    where $E'(w^*) = 0$, as w^* minimises E
  - in multiple dimensions:

$$E(\underline{w}) \approx E(\underline{w}^*) + \frac{1}{2} (\underline{w} - \underline{w}^*)^T H (\underline{w} - \underline{w}^*)$$

  - this indicates that close to $\underline{w}^*$ the energy behaves **quadratically**: precisely like the energy for a **Gaussian** distribution
  - hence, if the Taylor series is accurate (i.e the posterior is tightly peaked), the Taylor expansion of the energy will be accurate, and our approximation as a Gaussian will be good
  - this is the **Bayesian Central Limit Theorem**
- **In which situations is the Laplace approximation unreasonable?**
  - when the approximated distribution isn't very Gaussian like, even if the **Laplace approximation** matches the **mode** and **curvature**, the approximation can be very off
  - for example, in certain directions of parameter space, data might not be too informative, which can produce an **asymmetric posterior** (as we saw above)

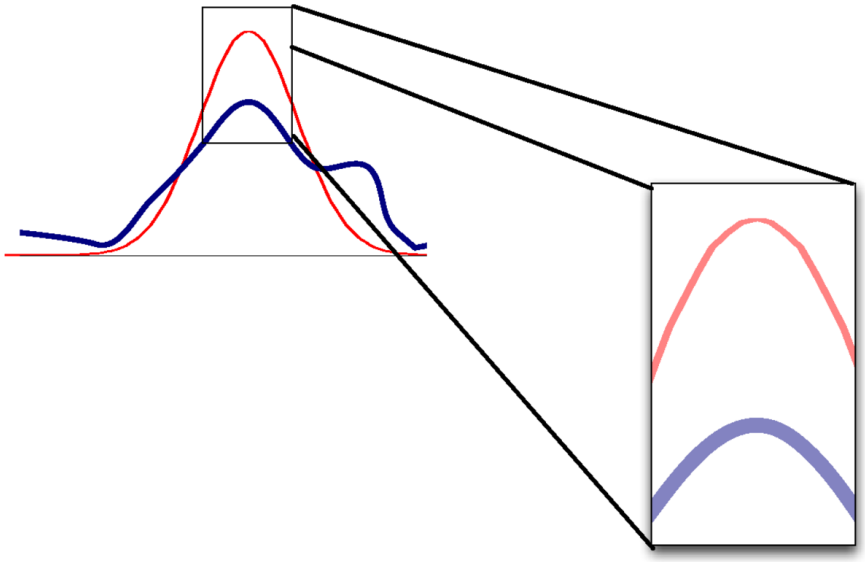

Figure 5: When the posterior distribution is non-Gaussian, then the values of the densities will likely not match.

- this means that the approximation of $P(\mathcal{D})$ won't be good (for example, in the diagram above, since $\mathcal{N}(w^*; w^*, H^{-1}) \geq P(w^*, \mathcal{D})$ our **posterior** is an **overestimate**, so we are **underestimating** $P(w^*, \mathcal{D})$, and thus, we will underestimate $P(\mathcal{D})$)
- another example of where this fares badly is for **multimodal** distributions (which are clearly non-Gaussian)

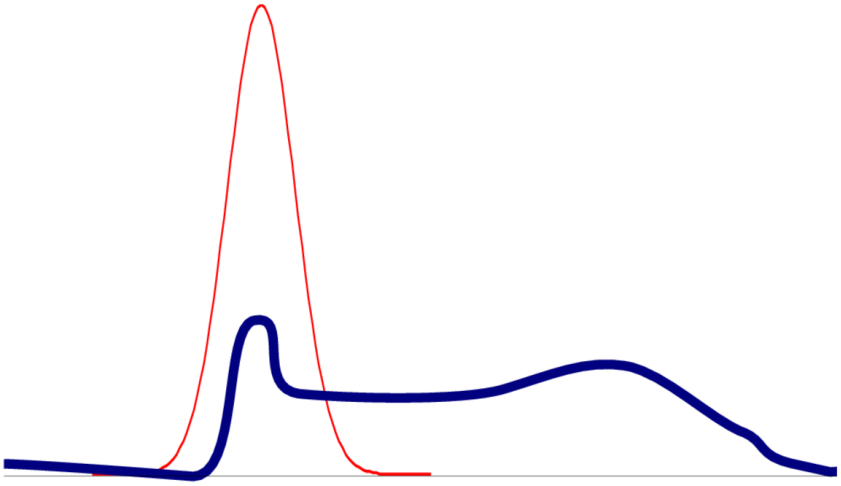

Figure 6: The Laplace Approximation clearly fails at gauging multimodal data, since it just expects one mode. Even if it correctly captures a mode, it might not be the "best" mode.

- this is problematic, since many NN posteriors will be multimodal
- if the **posterior** is **flat** in some direction, there will be near 0 curvature, in which case the Hessian won't be positive-definite, and thus, won't give us a meaningful approximation

4 Question

4.1 Notes Questions

1. **Say that, as above, we consider the posterior for a single data point with $y = 1$ and $x = -20$ with distribution:**

$$P(\underline{w}) \propto \mathcal{N}(w; 0, 0.1)$$

$$P(\underline{w} \mid \mathcal{D}) \propto \mathcal{N}(w; 0, 0.1)\sigma(10 - 20w)$$

How does the posterior $P(\underline{w} \mid \mathcal{D})$ look?

- notice, the variance is now much smaller, so most of the probability mass will be clustered around $w = 0$
- the sigmoid is basically 1 for $w < 0.3$, so:

$$P(\underline{w} \mid \mathcal{D}) \approx P(\underline{w})$$

- the sigmoid will only have the effect of "cutting" the distribution when $w >= 0.5$, but for these points, the prior has nearly 0 probability mass
- hence, prior and posterior should be nearly indistinguishable

2. **The *Poisson* distribution is defined by a parameter λ. Say we have observed r counts from data. The prior and likelihood for a Poisson distribution is:**

$$P(\lambda) \propto \frac{1}{\lambda}$$

$$P(r \mid \lambda) = \exp(-\lambda)\frac{\lambda^r}{r!}$$

Use a Laplace approximation to the powerior over λ, given an observed count, to infer the distribution of λ.

- we define the energy as (up to a constant):

$$E(\lambda) = -\log(P(\lambda)P(r \mid \lambda)) = \lambda - (r-1)\log\lambda$$

- the minimum λ^* is:

$$E'(\lambda) = 1 - \frac{r-1}{\lambda} \implies \lambda^* = r - 1$$

- the curvature H is:

$$E''(\lambda^*) = \frac{r-1}{\lambda^{*2}} = \frac{1}{r-1}$$

- hence, and assuming that $r > 1$ (otherwise the curvature will be undefined):

$$P(\lambda \mid r) \approx \mathcal{N}(\lambda; r-1, r-1)$$