# Logistic Regression & Laplace Approximation

**Mark van der Wilk**

Department of Computing
Imperial College London

@markvanderwilk
m.vdwilk@imperial.ac.uk

February 20, 2023

# Approximate Inference (Part III)

So far:

- How to use Bayes' rule to learn about unseen quantities (I)
  - Manipulating probability distributions, graphical models
  - Gaussian processes
- How to use uncertainty to make decisions (II)

In part III, we will look at:

- models that require intractable computations
- properties of intractable computations
- approximations to Bayes' rule

# Today

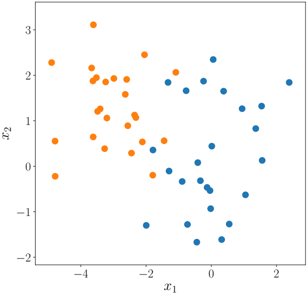
Today we will discuss:

- Non-conjugate model: Logistic Regression
- Posterior approximation: Laplace Approximation
- Predictive approximation: Monte Carlo

# Further Reading

- Pattern Recognition and Machine Learning, Chapter 4 (Bishop, 2006)
- Machine Learning: A Probabilistic Perspective, Chapter 8 (Murphy, 2012)

# Binary Classification




- Supervised learning setting with inputs $\boldsymbol{x}_n \in \mathbb{R}^D$ and binary targets $y_n \in \{0, 1\}$ belonging to classes $\mathcal{C}_1, \mathcal{C}_2$.
- Objective:
  - Given new test input $\mathbf{x}_n^*$, predict the label $y_n^*$.
  - Find a decision boundary/surface that separates the two classes

# Class Posteriors

- Binary classification problem with two classes $\mathcal{C}_1, \mathcal{C}_2$.
- Posterior class probability $p(y = 1|\boldsymbol{x}) = p(\mathcal{C}_1|\boldsymbol{x})$:

$$p(\mathcal{C}_1|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\boldsymbol{x})},$$
$$p(\boldsymbol{x}) = p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1) + p(\boldsymbol{x}|\mathcal{C}_2)p(\mathcal{C}_2)$$

▶▶ Learning from data requires figuring out what $p(\mathbf{x} \mid \mathcal{C}_c)$ is from data.

# Generative modelling

- Inputs can be high-dimensional (e.g. images)
- $p(\mathbf{x} \mid \mathcal{C}_c)$ can be very complicated

Imagine learning how to create photorealistic images before being able to recognise them!

# Density ratios

We only need the **ratio of weighted likelihoods**

$$p(\mathcal{C}_1|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1) + p(\boldsymbol{x}|\mathcal{C}_2)p(\mathcal{C}_2)},$$
$$= \frac{1}{1 + \frac{p(\boldsymbol{x}|\mathcal{C}_2)p(\mathcal{C}_2)}{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1)}},$$

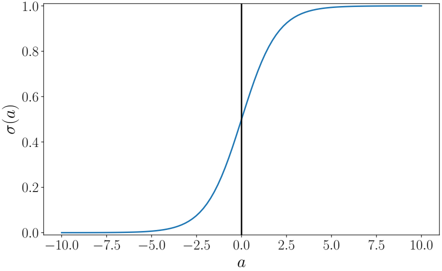
Idea: Instead of learning $p(\mathbf{x}\,|\,\mathcal{C}_c)$, can we just learn $\frac{p(\boldsymbol{x}|\mathcal{C}_2)p(\mathcal{C}_2)}{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1)}$?

$$p(\mathcal{C}_1\,|\,\mathbf{x}, r(\cdot)) = \frac{1}{1 + r(\mathbf{x})} \qquad \text{with } r : \mathbb{R}^D \to \mathbb{R}^+ . \tag{1}$$

Positive functions are a pain... Let's take logs to use $f : \mathbb{R}^D \to \mathbb{R}$:

$$p(\mathcal{C}_1\,|\,\mathbf{x}, f(\cdot)) = \underbrace{\frac{1}{1 + \exp(-f(\mathbf{x}))}}_{\text{Logistic sigmoid } \sigma(f(\mathbf{x}))} \tag{2}$$

# Logistic Sigmoid




$$f(\mathbf{x}) := \log \frac{p(\mathcal{C}_1|\boldsymbol{x})}{p(\mathcal{C}_2|\boldsymbol{x})} = \log \frac{p(\boldsymbol{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\boldsymbol{x}|\mathcal{C}_2)p(\mathcal{C}_2)}$$

$$\sigma(f(\mathbf{x})) := \frac{1}{1+\exp(-f(\mathbf{x}))} = p(\mathcal{C}_1|\boldsymbol{x})$$

# What type of function should $f(\cdot)$ be?

- Assume Gaussian class conditionals

$$p(\boldsymbol{x}|\mathcal{C}_k) = \mathcal{N}\left(\boldsymbol{x}\,|\,\boldsymbol{\mu}_k, \boldsymbol{\Sigma}\right)$$

  where the covariance matrix $\boldsymbol{\Sigma}$ is shared across all $K$ classes.

- For $K = 2$ we get (Bishop, 2006)

$$p(\mathcal{C}_1|\boldsymbol{x}) = \sigma(\boldsymbol{\theta}^\top \boldsymbol{x} + \theta_0)\,,$$

$$\boldsymbol{\theta} := \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\,, \quad \theta_0 := \frac{1}{2}\left(\boldsymbol{\mu}_2^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1\right) + \log\frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}$$

  ⏩ Argument of the sigmoid is linear in $\boldsymbol{x}$
  ⏩ Decision boundary is a surface along which the posterior class probabilities $p(\mathcal{C}_k|\boldsymbol{x})$ are constant
  ⏩ **Decision boundary is a linear function of $\boldsymbol{x}$**

- If covariances are not shared: Quadratic decision boundaries

# Classifying from data samples

One approach (generative):

1. Define priors over two Gaussian distributions for $p(\mathbf{x} \mid \mathcal{C}_c)$
2. Given data, find posteriors over Gaussians
3. Given our beliefs over $p(\mathbf{x} \mid \mathcal{C}_c)$, apply Bayes' rule to get $p(\mathcal{C}_c \mid \mathbf{x})$

Alternative approach (discriminative):

1. Define prior on linear functions for $f(\cdot)$
2. Given data, find posterior over $f(\cdot)$, which directly translates to $p(\mathcal{C}_c \mid \mathbf{x})$

# Classifying from data samples

One approach:

1. Define priors over two **general** distributions for $p(\mathbf{x} \mid \mathcal{C}_c)$
2. Given data, find posteriors over **distributions**
3. Given our beliefs over $p(\mathbf{x} \mid \mathcal{C}_c)$, apply Bayes' rule to get $p(\mathcal{C}_c \mid \mathbf{x})$

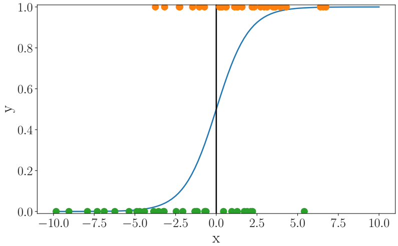
Alternative approach:

1. Define prior on **general, non-linear** functions for $f(\cdot)$
2. Given data, find posterior over $f(\cdot)$, which directly translates to $p(\mathcal{C}_c \mid \mathbf{x})$

# Model Specification – Logistic regression

- Bernoulli likelihood

$$y \in \{0, 1\}$$
$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathrm{Ber}(y|\mu(\boldsymbol{x}))\,,$$
$$\mu(\boldsymbol{x}) = p(y = 1|\boldsymbol{x}) = \sigma(\boldsymbol{\theta}^\top \boldsymbol{x})$$




- Label $y$ depends on input location $\boldsymbol{x}$, i.e., $\mu(\boldsymbol{x})$ needs to be a function of $\boldsymbol{x}$
- Idea: Linear model $\boldsymbol{\theta}^\top \boldsymbol{x}$ (as in linear regression)
- Ensure $0 \leqslant \mu(\boldsymbol{x}) \leqslant 1$
- Squash the linear combination through a function that guarantees this: $\mu(\boldsymbol{x}) = \sigma(\boldsymbol{\theta}^\top \boldsymbol{x})$

$$\Longrightarrow p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathrm{Ber}(y|\sigma(\boldsymbol{\theta}^\top \boldsymbol{x}))$$

# Model fitting

Model is very similar to **linear regression**, but with a different **likelihood**.

- Can we find the posterior?

$$p(\boldsymbol{\theta} \mid X, \mathbf{y}) = \frac{\prod_{n=1}^{N} p(y_n \mid \sigma(\boldsymbol{\theta}^\top \mathbf{x}))p(\boldsymbol{\theta})}{p(\mathbf{y} \mid X)} \tag{3}$$

- Can we find the predictive distribution?

$$p(y^* \mid X, \mathbf{y}, \mathbf{x}^*) = \int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*)p(\boldsymbol{\theta} \mid X, \mathbf{y})\mathrm{d}\boldsymbol{\theta} \tag{4}$$

# Logistic regression posterior

$$p(\boldsymbol{\theta} \mid X, \mathbf{y}) = \frac{\prod_{n=1}^{N} p(y_n \mid \sigma(\boldsymbol{\theta}^\top \mathbf{x})) p(\boldsymbol{\theta})}{p(\mathbf{y} \mid X)} \tag{5}$$

$$= \frac{1}{p(\mathbf{y} \mid X)} \prod_{n=1}^{N} \mathrm{Ber}(y_n | \sigma(\boldsymbol{\theta}^\top \mathbf{x})) \mathcal{N}(\boldsymbol{\theta}; 0, v\mathbf{I}), \tag{6}$$

$$p(\mathbf{y} \mid X) = \int p(\mathbf{y} \mid X, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}. \tag{7}$$

Problem 1:

1. No closed-form solution for the marginal likelihood
2. Can only evaluate the posterior up to a constant

# Logistic regression predictive distribution

$$p(y^* \mid X, \mathbf{y}, \mathbf{x}^*) = \int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*) p(\boldsymbol{\theta} \mid X, \mathbf{y}) \mathrm{d}\boldsymbol{\theta} \tag{8}$$

$$= \frac{1}{p(\mathbf{y} \mid X)} \int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*) \cdot \prod_{n=1}^{N} \mathrm{Ber}(y_n | \sigma(\boldsymbol{\theta}^\top \mathbf{x})) \mathcal{N}(\boldsymbol{\theta}; 0, v\mathbf{I}) \mathrm{d}\boldsymbol{\theta} \tag{9}$$

Problem 2:

- No closed-form solution to integral (similar to marginal likelihood)
- Also need to normalise by the marginal likelihood

# Point Estimate

- Estimate model parameters $\boldsymbol{\theta}$ as a point, not a distribution (MLE or MAP)
- Likelihood (training data $\boldsymbol{X}, \boldsymbol{y}$):

$$p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \mathrm{Ber}(y_n|\sigma(\boldsymbol{\theta}^\top \boldsymbol{x}_n)) = \prod_{n=1}^{N} (\sigma(\boldsymbol{\theta}^\top \boldsymbol{x}_n))^{y_n} (1 - \sigma(\boldsymbol{\theta}^\top \boldsymbol{x}_n))^{1-y_n}$$

$$= \prod_{n=1}^{N} \mu_n^{y_n} (1 - \mu_n)^{1-y_n}$$

$$\mu_n := \sigma(\boldsymbol{\theta}^\top \boldsymbol{x}_n)$$

- Minimise **negative log likelihood (cross-entropy):**

$$NLL = -\sum_{n=1}^{N} y_n \log \mu_n + (1 - y_n) \log(1 - \mu_n)$$

# Model Fitting (2)

- Derivative of sigmoid w.r.t. its argument:

$$\sigma(z_n) = \frac{1}{1+\exp(-z_n)}$$
$$\implies \frac{\mathrm{d}\sigma(z_n)}{\mathrm{d}z_n} = \frac{\exp(-z_n)}{(1+\exp(-z_n))^2} = \sigma(z_n)(1-\sigma(z_n))$$

- Gradient of the negative log-likelihood:

$$\frac{\mathrm{d}NLL}{\mathrm{d}\boldsymbol{\theta}} = -\sum_{n=1}^{N}\left(y_n\frac{1}{\mu_n} - (1-y_n)\frac{1}{1-\mu_n}\right)\frac{\mathrm{d}\mu_n}{\mathrm{d}\boldsymbol{\theta}}$$
$$\frac{\mathrm{d}\mu_n}{\mathrm{d}\boldsymbol{\theta}} = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\theta}}\sigma(\underbrace{\boldsymbol{\theta}^\top\boldsymbol{x}_n}_{z_n}) = \frac{\mathrm{d}\sigma(z_n)}{\mathrm{d}z_n}\frac{\mathrm{d}z_n}{\mathrm{d}\boldsymbol{\theta}} = \sigma(z_n)(1-\sigma(z_n))\boldsymbol{x}_n^\top$$

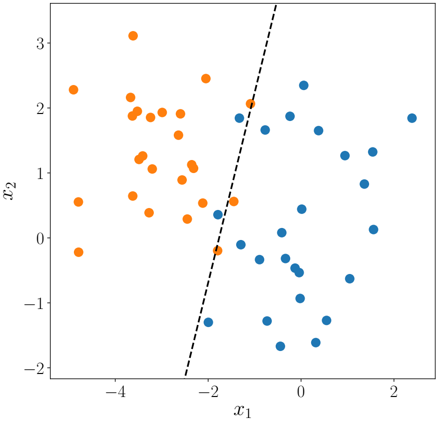
# Model Fitting (3)

$$\frac{\mathrm{d}NLL}{\mathrm{d}\boldsymbol{\theta}} = (\boldsymbol{\mu} - \boldsymbol{y})^\top \boldsymbol{X}$$
$$\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]^\top$$

- No closed-form solution ⏩ Gradient descent methods
- Unique global optimum exists (NLL) is **convex**.

$$p(\boldsymbol{\theta} \,|\, \boldsymbol{X}, \mathbf{y}) \approx \delta(\boldsymbol{\theta} - \boldsymbol{\theta}^*) \tag{10}$$
$$\boldsymbol{\theta}^* = \underset{\theta}{\operatorname{argmax}} \log p(\mathbf{y} \,|\, \boldsymbol{X}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) \tag{11}$$

# Maximum likelihood solution

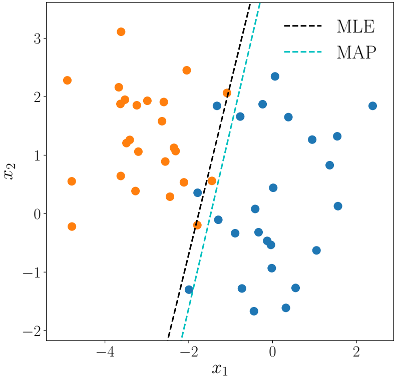



$$p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \mathrm{Ber}(\sigma(\theta_0 + \theta_1 x_1 + \theta_2 x_2))$$

# Comments on Maximum Likelihood

- If the classes are linearly separable, the decision boundary is not unique and the predictions will become extreme
- Overfitting is a again a problem when we work with features $\boldsymbol{\phi}(\boldsymbol{x})$ instead of $\boldsymbol{x}$ (or a GP for that matter)
- Maximum a posteriori estimation can address these issues to some degree

# MAP Solution

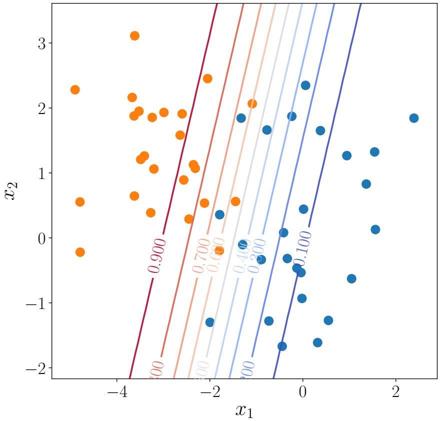



- Log-posterior:

$$\log p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}) = \log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) + \text{ const}$$

- No closed-form solution for $\boldsymbol{\theta}_{\mathrm{MAP}}$
  ▶▶ Numerical maximization of the log-posterior

# Predictive Labels




$$p(y = 1|\boldsymbol{x}, \boldsymbol{\theta}_{\text{MAP}}) = \text{Ber}(\sigma(\boldsymbol{x}^\top \boldsymbol{\theta}_{\text{MAP}})$$

# Approximate Inference

If we can't do the required integrals exactly,
... can we approximate them?

- The true posterior is intractable
- Can we find a manageable distribution that is close?

Gaussian distributions are manageable,
so can we find a Gaussian approximation?

# Laplace Approximation

For a distribution $p(\mathbf{x}) = \frac{1}{Z}\tilde{p}(\mathbf{x})$

- Maximising $\tilde{p}(\mathbf{x})$ gives us the mode $\mathbf{x}^*$
- Can we find an approximation to the variance?
  - ⏩ 2nd order Taylor-series approximation

$$\log p(\mathbf{x}) \approx -\log Z + \log \tilde{p}(\mathbf{x}^*) + \mathbf{J}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^\top \boldsymbol{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*)$$

$$\mathbf{J}: \quad \text{Jacobian}, \qquad \mathbf{H}: \text{Hessian}.$$

$$\log p(\mathbf{x}) \approx -\log Z + \log \tilde{p}(\mathbf{x}^*) + \cancelto{0}{\mathbf{J}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*)} + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^\top \boldsymbol{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*)$$

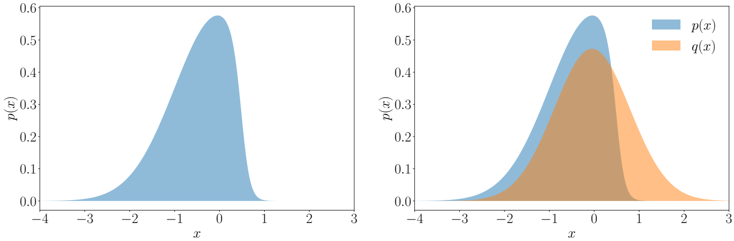
# Laplace Approximation: Marginal Likelihood

We can apply the Laplace approximation to approximate a posterior:

$$
\begin{aligned}
p(\mathbf{x}|\mathbf{y}) &= \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{\int p(\mathbf{y}|\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{x}} \\
&= \frac{1}{Z}\tilde{p}(\mathbf{x})
\end{aligned}
$$

- $Z$ is the marginal likelihood!

# Laplace Approximation: Example




- Unnormalized distribution:

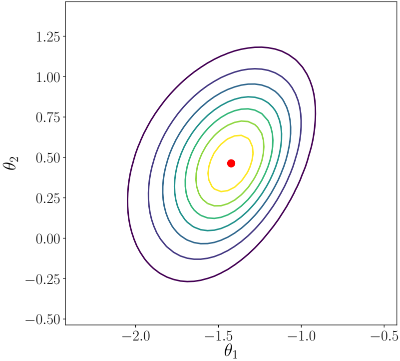
$$\tilde{p}(x) = \exp(-\tfrac{1}{2}x^2)\sigma(ax + b)$$

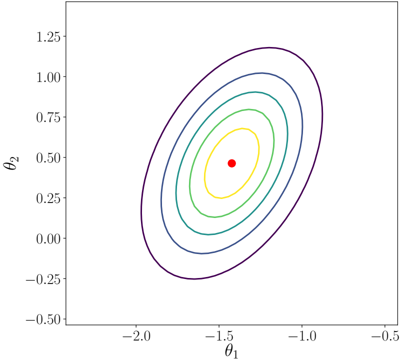
$$q(x) = \mathcal{N}\left(x \,\middle|\, x^*, (1 + a^2\mu_*(1-\mu_*))^{-1}\right), \quad \mu_* := \sigma(ax_* + b)$$

# Laplace Approximation: Properties

- Only need to know the unnormalized distribution $\tilde{p}$
- Finding the mode: numerical methods (optimization problem)
- Captures only local properties of the distribution
- Multimodal distributions: Approximation will be different depending on which mode we are in (not unique)
- For large datasets, we would expect the posterior to converge to a Gaussian (Bernstein-von Mises theorem)
  ▸▸ Laplace approximation should work well in this case

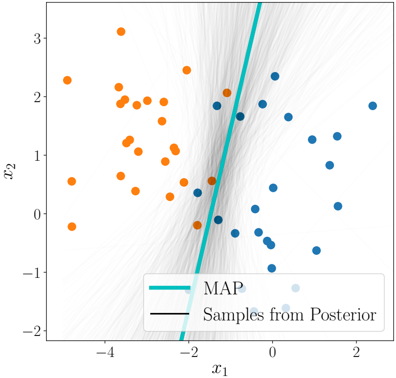
# Logistic Regression Posterior Approximation




- Left: true parameter posterior
- Right: Laplace approximation

# Posterior Decision Boundary




- Parameter samples $\boldsymbol{\theta}_i$ drawn from Laplace approximation $q(\boldsymbol{\theta})$ of posterior $p(\boldsymbol{\theta}|\boldsymbol{X})$
- Decision boundary drawn for each $\boldsymbol{\theta}_i$

# Predictions

Assume a Gaussian distribution $q(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ on the parameters (e.g., Laplace approximation of the posterior). Then:

$$p(y^* \mid X, \mathbf{y}, \mathbf{x}^*) = \int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*) p(\boldsymbol{\theta} \mid X, \mathbf{y}) \mathrm{d}\boldsymbol{\theta} \tag{14}$$

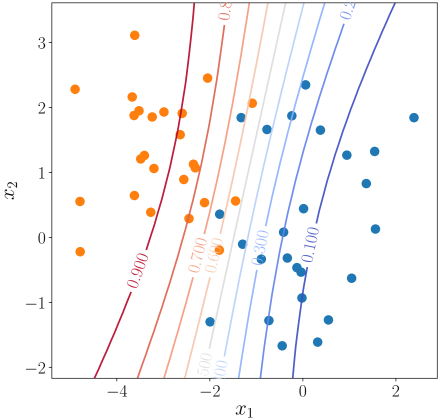
$$\approx \int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*) q(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \tag{15}$$

▸▸ **Integral intractable** ▸▸ Use **Monte Carlo** approximation

$$\int p(y^* \mid \boldsymbol{\theta}, \mathbf{x}^*) q(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \approx \frac{1}{S} \sum_{s=1}^{S} p(y^* \mid \boldsymbol{\theta}^{(s)}, \mathbf{x}^*) \tag{16}$$

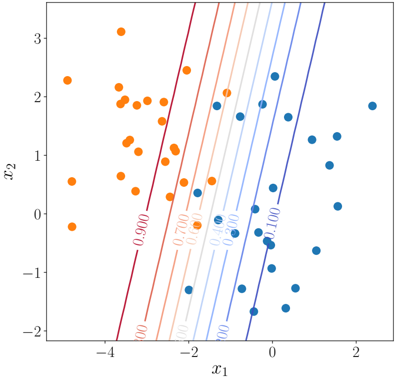
$$\boldsymbol{\theta}^{(s)} \sim q(\boldsymbol{\theta}) \tag{17}$$

# Predictions (2)

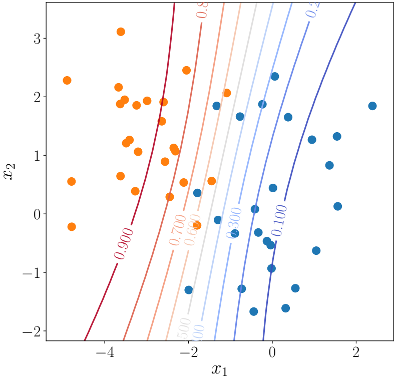



1. Samples from Laplace approximation of the posterior
2. Monte-Carlo estimate of label prediction

# Comparison with MAP Predictions




(a) MAP




(b) Bayesian Logistic Regression

- Predictive labels

# Specifying Monte Carlo Approximations

A full specification of a MC procedure (e.g. in an exam) requires:

- Statement of what is to be computed, e.g. $\int f(\mathbf{x})p(\mathbf{x})\mathrm{d}\mathbf{x}$.
- What we compute in our approximation, e.g. $\sum_{s=1}^{S} f(\mathbf{x}^{[s]})$
- What distribution we sample from, e.g. $\mathbf{x}^{[s]} \sim p(\mathbf{x})$.
- A sentence explaining how we sample from the distribution.

# Sampling Procedures

You can assume that we can generate samples from **categorical distributions**, **uniform distributions**, and **standard Normal distributions**.

To generate samples, you can:

- Reparameterise a distribution. $x = t(\boldsymbol{\epsilon})$ (see MML [2])
  E.g. Gaussian $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{K})$

$$\mathbf{x} = \mathrm{chol}(\boldsymbol{K})\boldsymbol{\epsilon} + \boldsymbol{\mu} \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(0, I_M) \tag{18}$$

- Use rejection sampling (later)
- MCMC (later)

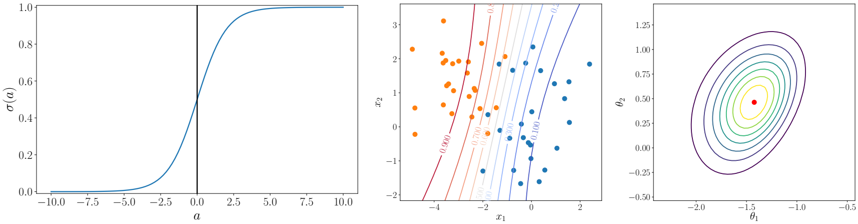
# Accuracy of MC Estimate

Remember from MML:

- As $S \to \infty$, the MC estimate converges to the right value.
- Variance determines accuracy for finite $S$ (Chebyshev's inequality).
- Want low variance!
- Can control this with $S$.
- Other techniques in future lectures.

Todo: Make nice notebook illustrating MC estiamte

# Summary




- Binary classification problems
- Linear model with non-Gaussian likelihood
- Implicit modeling assumption: Gaussian $p(\mathbf{x} \mid \mathcal{C}_c)$
- Parameter estimation (MLE, MAP) no longer in closed form
- Bayesian logistic regression with Laplace approximation of the posterior

# References I


[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer-Verlag, 2006.

[2] M. P. Deisenroth, A. A. Faisal, and C. S. Ong. *Mathematics for machine learning*. Cambridge University Press, 2020.

[3] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.