


Bayesian Linear Regression

Mark van der Wilk

Department of Computing
Imperial College London

 @markvanderwilk
m.vdwilk@imperial.ac.uk

November 16, 2021

Mathematics for Machine Learning:

<https://mml-book.com>

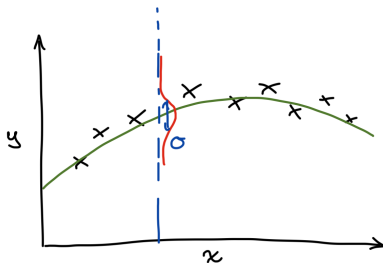
Chapter 9

Probabilistic Models

Probabilistic model: Model of the data is a probability distribution.

$$y_n = f(\mathbf{x}_n; \theta) + \epsilon_n \quad \epsilon_n \sim \mathcal{N}(0, \sigma^2)$$

We can now also estimate the **unpredictability** of our problem:

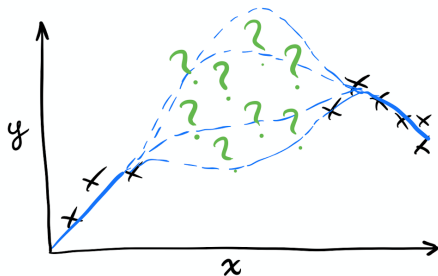


$$(\theta^*, \sigma^{2*}) = \operatorname{argmax}_{\theta, \sigma^2} \log p(\mathbf{y} | \theta, \sigma^2, X) \quad (1)$$

Unpredictability remains even if we **know** underlying function.
Goes by many names... e.g. **aleatoric uncertainty**.

Uncertainty in Parameters/Function

Aren't we also uncertain when we have a lack of data?



This is uncertainty in the parameters that define the function!
Also goes by many names... e.g. **epistemic uncertainty**.

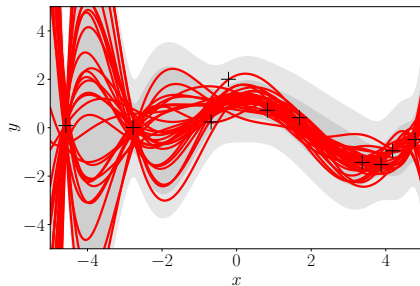
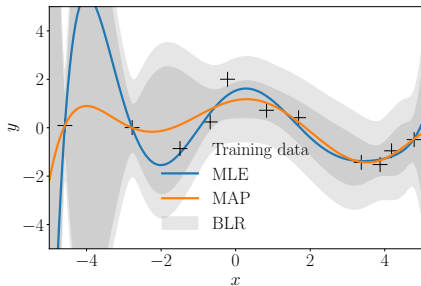
Quantifying Uncertainty with Bayesian Inference

If we knew that for a series of problems, our parameters θ were sampled from $p(\theta)$, then Bayes' rule would give us the probability distribution after observing our data \mathbf{y} :

$$\underbrace{p(\theta|\mathbf{y})}_{\text{posterior}} = \frac{\overbrace{p(\mathbf{y}|\theta)}^{\text{likelihood}} \overbrace{p(\theta)}^{\text{prior}}}{\underbrace{p(\mathbf{y})}_{\text{evidence}}} \quad (2)$$

- ▶ Allows us to quantify uncertainty in parameters θ .
- ▶ Bayesian inference makes a leap of faith: Choose a prior and assume this is the correct one.
- ▶ Choosing priors is important \implies Probabilistic Inference (Spring).

Example

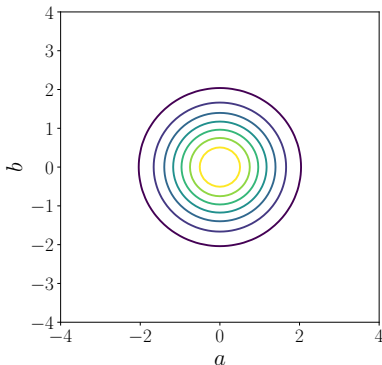


- ▶ Light-gray: uncertainty due to noise (aleatoric uncertainty / unpredictability)
- ▶ Dark-gray: uncertainty due to parameter uncertainty (epistemic uncertainty)
- ▶ Right: Plausible functions under the parameter distribution (every single parameter setting describes one function)

Distribution over Functions

Consider a linear regression setting

$$y = f(x) + \epsilon = a + bx + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$
$$p(a, b) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$



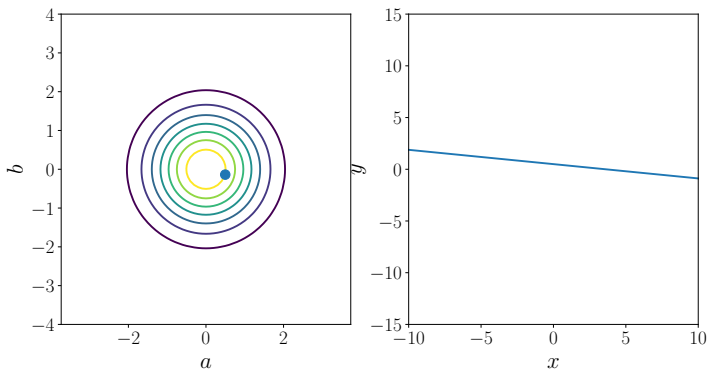
Sampling from the Prior over Functions

Consider a linear regression setting

$$y = f(x) + \epsilon = a + bx + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

$$p(a, b) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$f_i(x) = a_i + b_i x, \quad [a_i, b_i] \sim p(a, b)$$



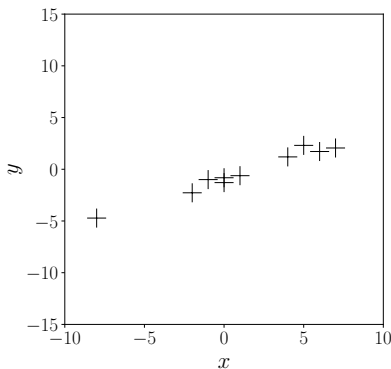
Sampling from the Posterior over Functions

Consider a linear regression setting

$$y = f(x) + \epsilon = a + bx + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

$$p(a, b) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$\mathbf{X} = [x_1, \dots, x_N], \quad \mathbf{y} = [y_1, \dots, y_N] \quad \text{Training inputs/targets}$$



Sampling from the Posterior over Functions

Consider a linear regression setting

$$y = f(x) + \epsilon = a + bx + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

$$p(a, b) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$p(a, b | \mathbf{X}, \mathbf{y}) = \mathcal{N}(\mathbf{m}_N, \mathbf{S}_N) \quad \text{Posterior}$$

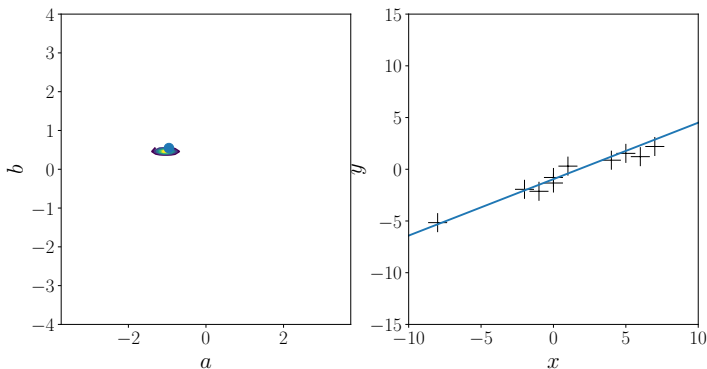
Sampling from the Posterior over Functions

Consider a linear regression setting

$$y = f(x) + \epsilon = a + bx + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_n^2)$$

$$[a_i, b_i] \sim p(a, b | \mathbf{X}, \mathbf{y})$$

$$f_i = a_i + b_i x$$



Model: Bayesian Linear Regression

We never put a distribution on any \mathbf{x}_n , so we drop from conditioning.

$$p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}; \mathbf{0}, \mathbf{I}_M) \quad (3)$$

$$p(y_n | \boldsymbol{\theta}) = \mathcal{N}(y_n; \boldsymbol{\phi}(\mathbf{x}_n)^\top \boldsymbol{\theta}, \sigma^2) \quad (4)$$

$$p(\mathbf{y} | \boldsymbol{\theta}) = \mathcal{N}(\mathbf{y}; \Phi(\mathbf{X})\boldsymbol{\theta}, \sigma^2 \mathbf{I}_N) \quad (5)$$

Two goals:

- ▶ Find posterior over parameters $p(\boldsymbol{\theta} | \mathbf{y})$
- ▶ Find predictive posterior $p(\mathbf{y}^* | \mathbf{y})$

Posterior over Parameters

Board:

- ▶ Equating coefficients (tests your matrix algebra skills!)
- ▶ Joint Gaussian
- ▶ Woodbury identity

Method 1: Crunching densities

$$\log p(\mathbf{y}|\boldsymbol{\theta}) = \log p(\mathbf{y}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) \quad (6)$$

$$= c - \frac{1}{2\sigma^2}(\mathbf{y} - \Phi(\mathbf{X})\boldsymbol{\theta})^\top(\mathbf{y} - \Phi(\mathbf{X})\boldsymbol{\theta}) - \frac{1}{2}\boldsymbol{\theta}^\top\boldsymbol{\theta} \quad (7)$$

This is a vector quadratic in $\boldsymbol{\theta}$! $\text{board} \implies$ Gaussian.

- ▶ Equate coefficients. Can rearrange... or find $\mathbb{E}+\mathbb{V}$ by other means
- ▶ Find maximum to find mean board
- ▶ Find Hessian to find covariance board

$$p(\boldsymbol{\theta}|\mathbf{y}) = \mathcal{N}\left(\boldsymbol{\theta}; \left[\frac{1}{\sigma^2}\Phi(\mathbf{X})^\top\Phi(\mathbf{X}) + \mathbf{I}_M\right]^{-1} \frac{1}{\sigma^2}\Phi(\mathbf{X})^\top\mathbf{y} \quad (8)$$

$$\left[\frac{1}{\sigma^2}\Phi(\mathbf{X})^\top\Phi(\mathbf{X}) + \mathbf{I}_M\right]^{-1}) \quad (9)$$

Method 2: Joint Gaussian

Find

$$p(\boldsymbol{\theta}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\theta} \\ \mathbf{y} \end{bmatrix}; \begin{bmatrix} \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{\theta}] \\ \mathbb{E}_{\mathbf{y}}[\mathbf{y}] \end{bmatrix}, \begin{bmatrix} \mathbb{V}[\boldsymbol{\theta}] & \mathbb{C}[\boldsymbol{\theta}, \mathbf{y}] \\ \mathbb{C}[\mathbf{y}, \boldsymbol{\theta}] & \mathbb{V}[\mathbf{y}] \end{bmatrix}\right) \quad (10)$$

board

$$p(\boldsymbol{\theta}|\mathbf{y}) = \mathcal{N}\left(\boldsymbol{\theta}; \Phi(X)^{\top} [\Phi(X)\Phi(X)^{\top} + \sigma^2 \mathbf{I}_N]^{-1} \mathbf{y}, \mathbf{I}_M - \Phi(X)^{\top} [\Phi(X)\Phi(X)^{\top} + \sigma^2 \mathbf{I}_N]^{-1} \Phi(X)\right) \quad (11)$$

Computational Considerations

Typical algorithms (i.e. not optimal ones) take:

- ▶ $O(NM^2)$ to multiply matrices of shape $M \times N$ with $N \times M$
(you must be able to derive this)
- ▶ $O(N^3)$ to find a matrix inverse
- ▶ The two results are certainly different in computational complexity!
- ▶ From joint is worse when $N \gg M$
- ▶ Are they different in value?

Woodbury identity

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1} \quad (12)$$

- ▶ Can go back and forth between the two forms.
- ▶ Allows you to implement the most efficient, based on setting of M and N .
- ▶ See exercise to practice.

Predictive posterior

- ▶ Crunching densities (see pdf)
- ▶ Equating coefficients (tests your matrix algebra skills!) (see pdf)
- ▶ Joint Gaussian (see exercise)
- ▶ May also need to apply the Woodbury identity

Method 1: Crunching densities

First, how to express our target in terms of densities we know.

$$p(\mathbf{y}^*|\mathbf{y}) \stackrel{\text{AT}}{=} \int \frac{p(\mathbf{y}^*, \boldsymbol{\theta}, \mathbf{y})}{p(\mathbf{y})} d\boldsymbol{\theta} \quad (13)$$

$$\stackrel{\text{MA}}{=} \int p(\mathbf{y}^*|\boldsymbol{\theta}) \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})} d\boldsymbol{\theta} \quad (14)$$

Next do the integrals / equating coefficients.

\implies I recommend other method.

Method 2: Expectation identities

Conclusion

- ▶ Bayesian Linear Regression quantifies uncertainty due to lack of data (epistemic uncertainty)
- ▶ Gaussians are easy to deal with when conditioning