COMP 551 – Applied Machine Learning
Lecture 18: Bayesian Inference

Instructor: Ryan Lowe (ryan.lowe@mail.mcgill.ca)

Slides mostly by: Herke van Hoof

Class web page: *www.cs.mcgill.ca/~hvanho2/comp551*

Announcements

- Assignment 2 grades should be available in the next week or so
- For Kaggle project: try using **square** bounding boxes
  - If you use regular bounding boxes, some digits that correspond to the correct label (e.g. ‘1’) will have a smaller bounding box by area

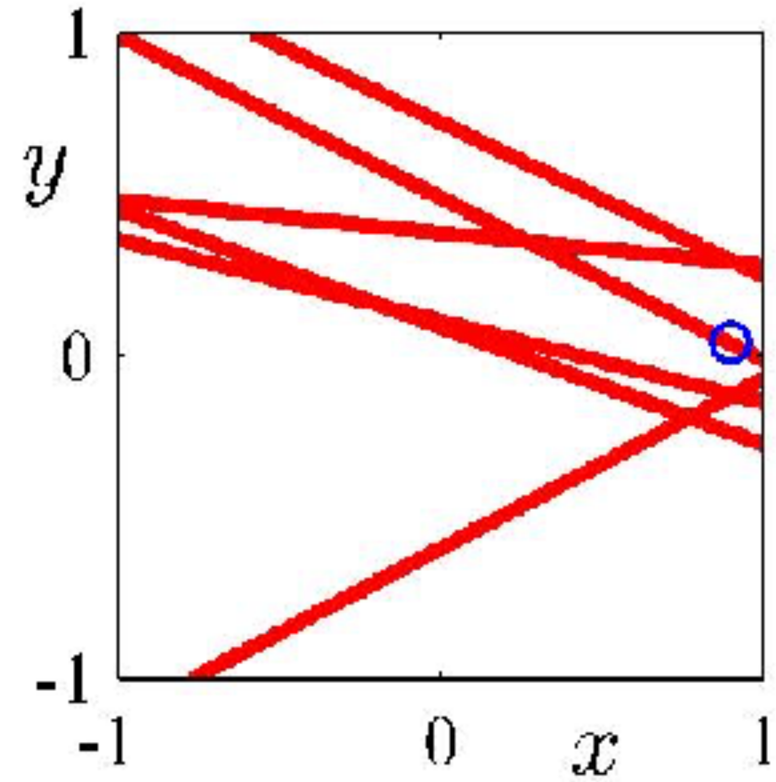
Bayesian probabilities

- An example from regression
- Given few noisy data points, multiple parameter values possible
- Can we **quantify uncertainty** over our parameters using probabilities?
- i.e. given a dataset:

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$$

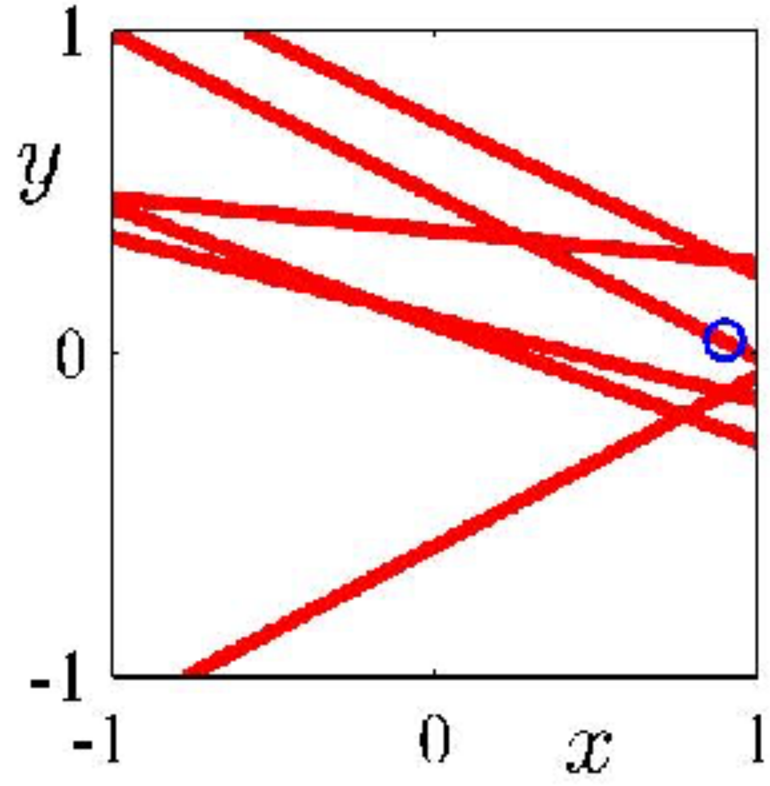
and some model with weights **w**, can we find:

$$p(\mathbf{w}|\mathcal{D}) \text{ ?}$$



Bayesian probabilities

- Yes we can!!
- **Bayesian view:** probability represents *uncertainty about some value or variable*
- We use Bayesian probabilities to represent uncertainty about the *parameters of our model*



Copyright C.M. Bishop, PRML

Bayesian probabilities

- To calculate uncertainty, need to **specify a model**. Two ingredients:
  1. **Prior** over model parameters: $p(\mathbf{w})$
  2. **Likelihood** term: $p(\mathcal{D}|\mathbf{w})$
- We are given a dataset:

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$$

- Want to do **inference** using Bayes' theorem:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}$$

Bayesian terminology

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}$$

- **Likelihood** $p(\mathcal{D}|\mathbf{w})$: our model of the data. Given our weights, how do we assign probabilities to dataset examples?
- **Prior** $p(\mathbf{w})$: before we see any data, what do we think about our parameters?
- **Posterior** $p(\mathbf{w}|\mathcal{D})$: our distribution over weights, given the data we've observed *and our prior*
- **Marginal likelihood** $p(\mathcal{D})$: also called the normalization constant. Does not depend on **w**, so not usually calculated explicitly

Bayesian probabilities

- How do we make predictions if we have a distribution over parameters?

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \int_{\mathbb{R}} p(y^*, \mathbf{w}|\mathbf{x}^*, \mathcal{D}) d\mathbf{w}$$

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \int_{\mathbb{R}^N} p(\mathbf{w}|\mathcal{D}) p(y^*|\mathbf{x}^*, \mathbf{w}) d\mathbf{w}$$

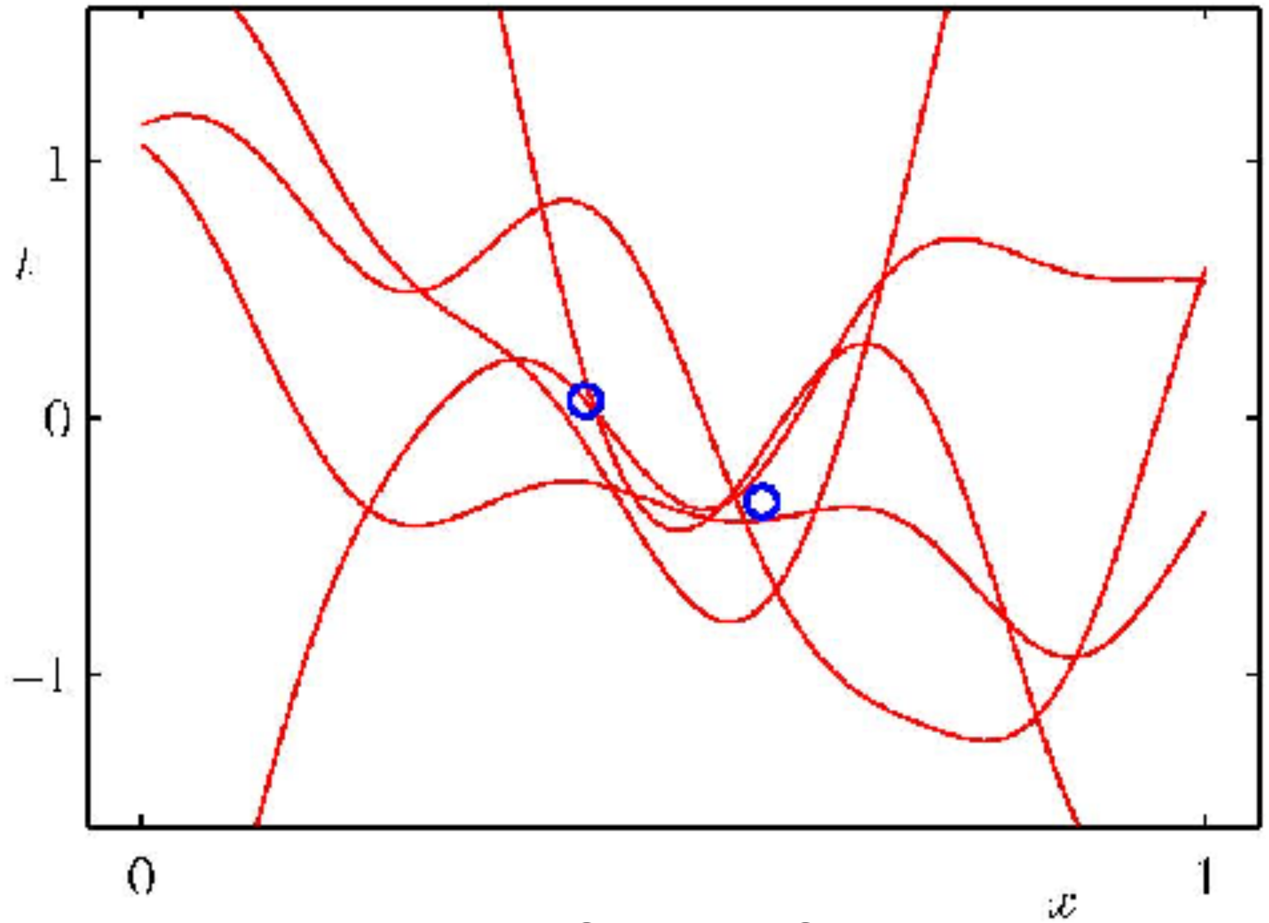
Posterior predictive distribution

- Rather than using a fixed value for parameters, **integrate over all possible parameter values**!
- (Integration is annoying, we will try to avoid this when possible)

Why Bayesian probabilities?

- Maximum likelihood estimates can have **large variance**
- We might desire or need an estimate of uncertainty
- Have **small dataset**, unreliable data, or small batches of data
- Use prior knowledge in a principled fashion

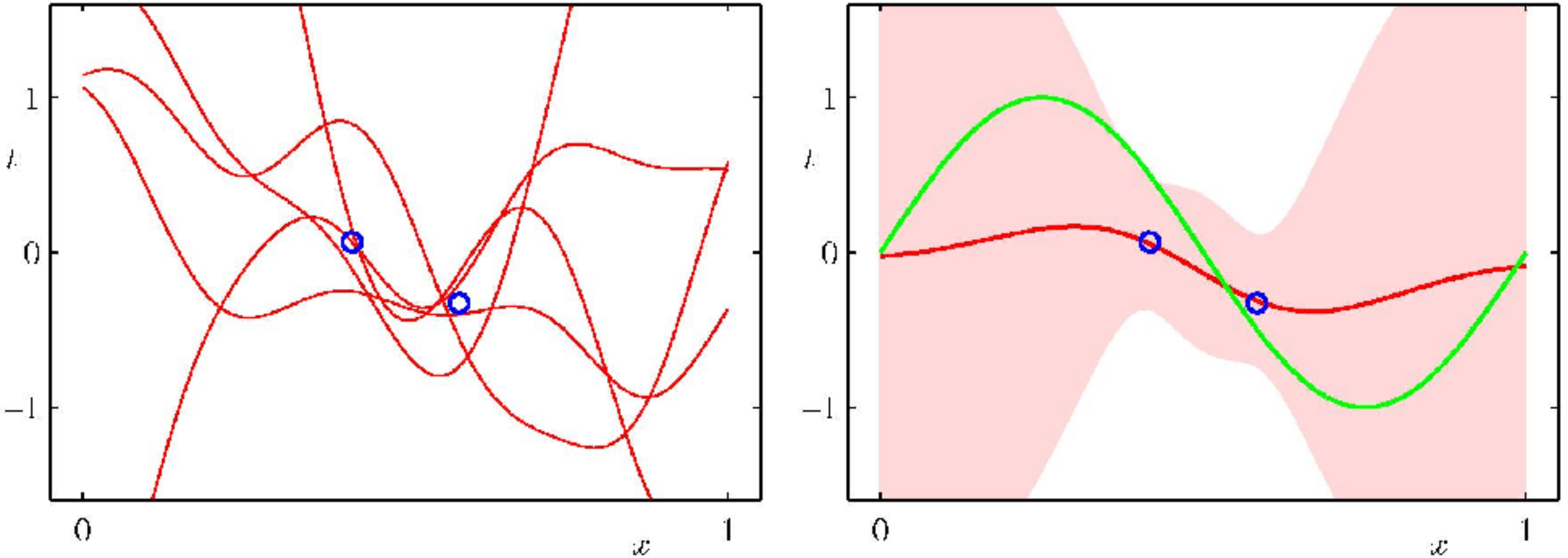
Why do we need uncertainty?



Copyright C.M. Bishop, PRML

- Regression with (extremely) small and noisy dataset
- Many functions are compatible with data

Why do we need uncertainty?



Copyright C.M. Bishop, PRML

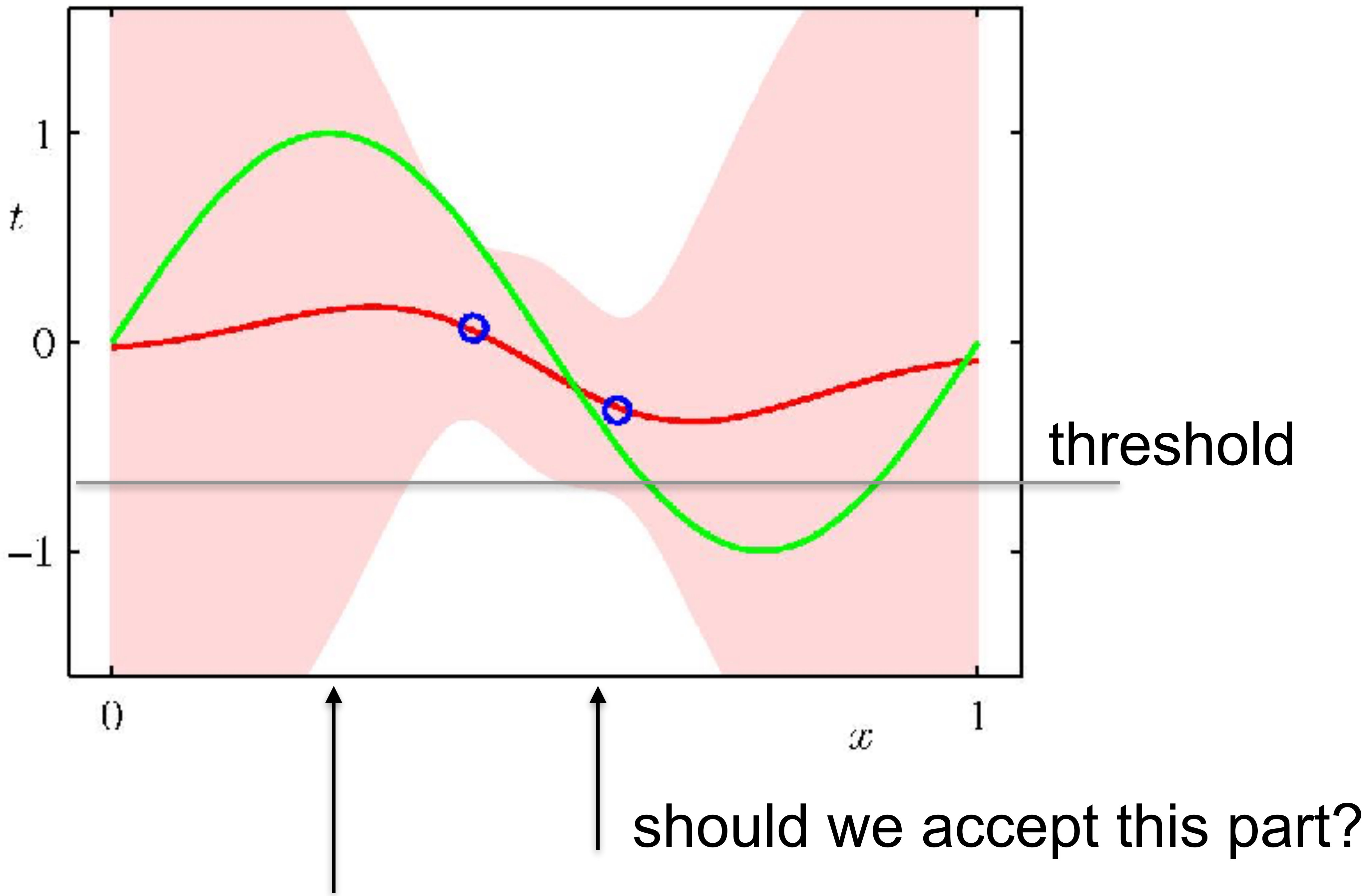
- Quantify the uncertainty using probabilities

  (e.g. Gaussian mean and variance for every input x)

Why do we need uncertainty?

- Knowing uncertainty of output *helpful in decision making*
- Consider inspecting task.
  - ***x***: some measurement
  - ***y***: predicted breaking strength
- **Parts which are too weak (breaking strength < t) are rejected**
  - Falsely rejecting a part incurs a small cost (*c=1*)
  - Falsely accepting a part can cause more damage down the line (expected cost *c=100*)

Decision making under uncertainty



Algorithms for Bayesian inference

- Given a dataset $\mathcal{D}$, how do we make predictions for a new input?

$$\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$$

- **Step 1**: Define a model that represents your data (the **likelihood**): $p(\mathcal{D}|\mathbf{w})$
- **Step 2:** Define a **prior** over model parameters: $p(\mathbf{w})$
- **Step 3:** Calculate **posterior** using Bayes' rule: $p(\mathbf{w}|\mathcal{D}) = \dfrac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}$
- **Step 4:** Make **prediction** by integrating over model parameters:

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \int_{\mathbb{R}^N} p(\mathbf{w}|\mathcal{D})p(y^*|\mathbf{x}^*, \mathbf{w})d\mathbf{w}$$

- **When can we do step 4) in closed form?**

Conjugate priors

- Posterior for some dataset:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{p(\mathcal{D})}$$

- Posterior for old data can act like a prior for new data:

$$p(\mathbf{w}|\mathcal{D}_1, \mathcal{D}_2) = \frac{p(\mathcal{D}_2|\mathbf{w})\boxed{p(\mathbf{w}|\mathcal{D}_1)}}{p(\mathcal{D}_2)}$$

- **Desirable that posterior and prior have same family!**
  - Otherwise posterior would get more complex with each step
- Such priors are called **<u>conjugate priors</u>** to a likelihood function

Conjugate priors

- Prediction

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \int_{\mathbb{R}^N} \underbrace{p(\mathbf{w}|\mathcal{D})p(y^*|\mathbf{x}^*, \mathbf{w})}_{\text{same family as prior}} d\mathbf{w}$$

- Argument of the integral is unnormalised distribution over **w**
- Integral calculates the normalisation constant
- For many common distributions, constant is known
  - Let's make the prior conjugate to a simple likelihood function, for which the constant is known

Algorithms for Bayesian inference

- Not all likelihood functions have conjugate priors
- However, so-called **exponential family** distributions do
  - Normal
  - Exponential
  - Beta
  - Bernoulli
  - Categorical
  - …

Examples

- We will look into supervised learning problems later
- Start with a simple problem, learning a single parameter with no inputs (i.e. no *x*): **a coin toss**
- Dataset consists of outcomes:

$$D = \{heads, heads, tails, heads, tails, \dots\}$$

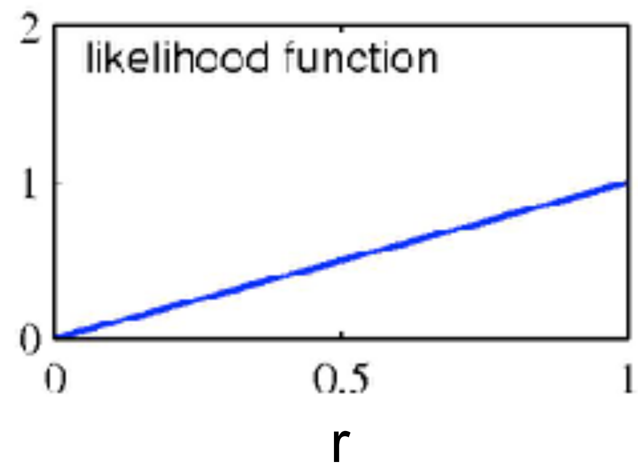
Simple example: coin toss

- Flip (possibly unfair) coin ***N*** times — get ***h*** heads and ***t*** tails
- Probability of ‘heads’ unknown value ***r***
- **How do we calculate the probability of the next flip being ‘heads’ (i.e. value of *r*) in a Bayesian way?**

Simple example: coin toss

- **<u>Step 1</u>: define model (distribution for likelihood)**
- Likelihood for a single flip: $\mathrm{Bern}(y|r) = r^y(1-r)^{1-y}$
  - ***y*** is one ('heads') or zero ('tails')
  - ***r*** is unknown parameter, between 0 and 1

likelihood for y=1



Copyright C.M. Bishop, PRML

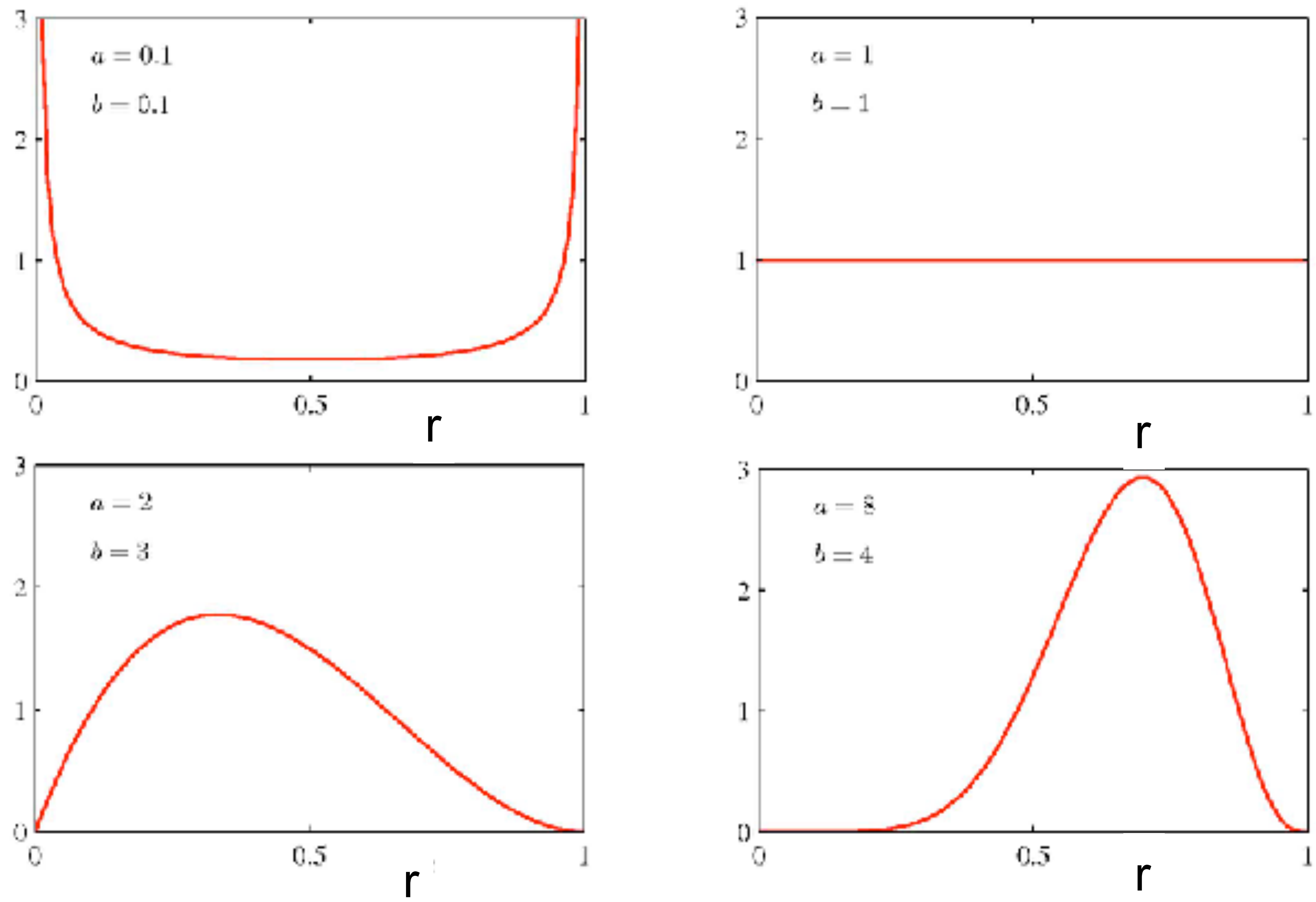
- Likelihood for *N* flips proportional to Binomial:

$$p(h|r,N) = r^h(1-r)^{N-h} \propto \mathrm{Bin}(h|r,N)$$

Simple example: coin toss

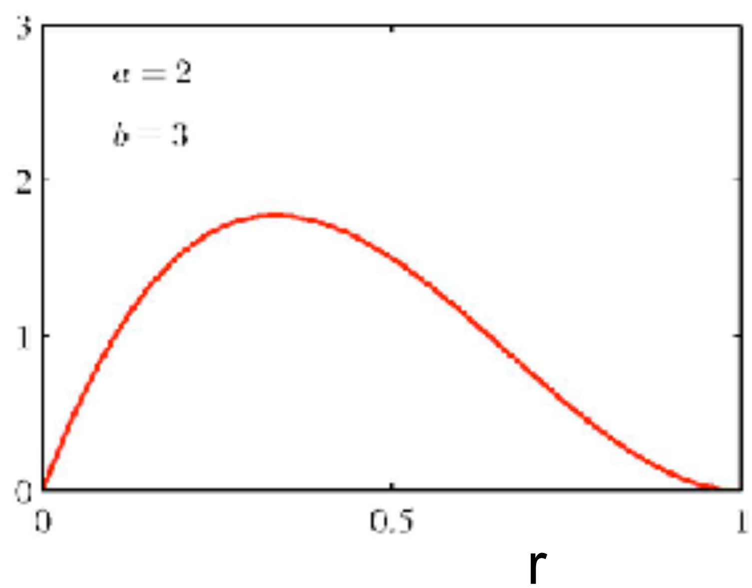
- **<u>Step 2</u>: Define (conjugate) prior *p(r)*:**

$$\mathrm{Beta}(r|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} r^{a-1}(1-r)^{b-1}$$

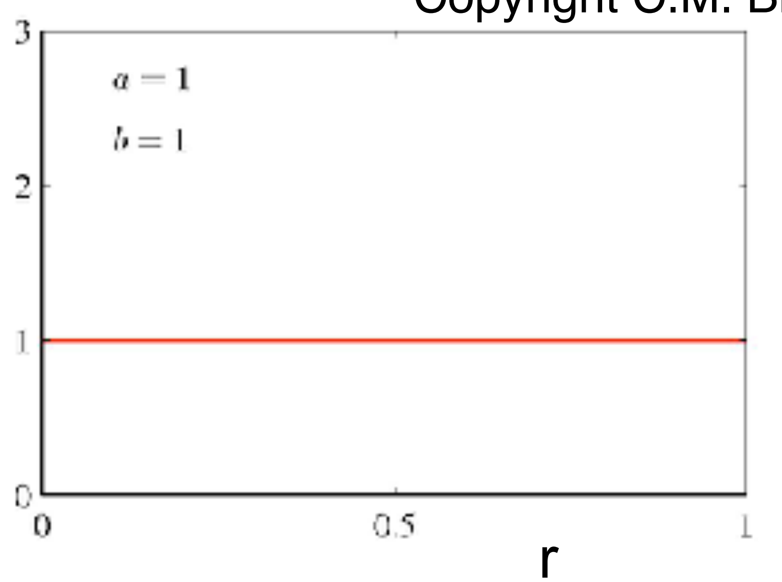


Simple example: coin toss

- <u>Conjugate prior</u>: $$\mathrm{Beta}(r|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} r^{a-1}(1-r)^{b-1}$$
- Prior denotes *a priori* belief over the value r
- r is a value between 0 and 1 (denotes prob. of heads or tails)
- a, b are ‘hyperparameters’



coin probably more likely to give ‘tails’



no idea about the fairness

Simple example: coin toss

- <u>Side note:</u> why is the Beta distribution the conjugate prior for a Binomial likelihood? (*N* = #flips, *h* = #heads)

<u>Step 3:</u> Calculate posterior!

$$
\begin{aligned}
p(r|\mathcal{D}) &= p(r|N,h) \\
&= p(h|r,N)\cdot p(r) \\
&= \mathrm{Bin}(h|r,N)\cdot \mathrm{Beta}(r|a,b) \\
&= r^h(1-r)^{N-h}\cdot\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}r^{a-1}(1-r)^{b-1} \\
&= z^{-1}r^{h+a-1}(1-r)^{N-h+b-1} \\
&= \mathrm{Beta}(r|h+a,N-h+b)
\end{aligned}
$$

N, *h* describe dataset completely

posterior = prior x likelihood

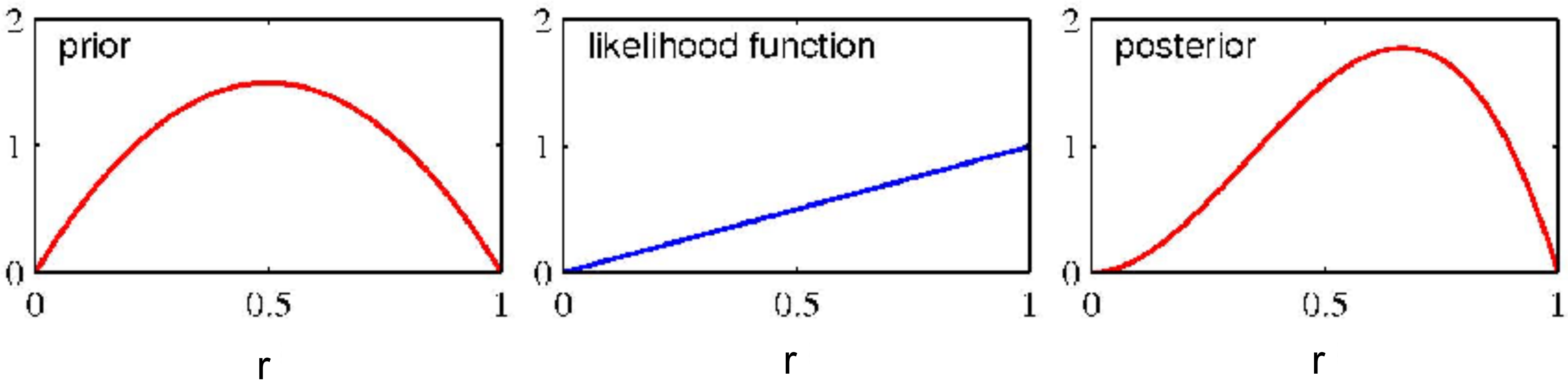
normalization factor

$$z^{-1}=\frac{\Gamma(h+a)\Gamma(N-h+b)}{\Gamma(a+b+N)}$$

Same distribution family (Beta) as prior!!!

Simple example: coin toss

- Posterior: $p(r|\mathcal{D}) = z^{-1} r^{h+a-1}(1-r)^{N-h+b-1}$



- We observe more ‘heads’ -> suspect more strongly coin is biased
- Note that *a, b* get added to the actual outcome: ‘pseudo-observations’

Simple example: coin toss

- Model:
  - <u>Likelihood:</u> $$\mathrm{Bern}(y|r) = r^y(1-r)^{1-y}$$
  - <u>Conjugate prior:</u> $$\mathrm{Beta}(r|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} r^{a-1}(1-r)^{b-1}$$
  - <u>Posterior:</u>

$$\mathrm{Beta}(r|h+a, N-h+b) = \frac{\Gamma(a+b+N)}{\Gamma(a+h)\Gamma(b+N-h)} r^{h+a-1}(1-r)^{N-h+b-1}$$

$$= z^{-1} r^{h+a-1}(1-r)^{N-h+b-1}$$

- **<u>Step 4</u>:** Make prediction!

Simple example: coin toss

- **<u>Step 4</u>**: Make prediction!

likelihood

posterior

$$
\begin{aligned}
p(x=1|\mathcal{D}) &= \int_0^1 p(x=1|r)p(r|\mathcal{D})dr \\
&= \int_0^1 r \cdot \mathrm{Beta}(r|h+a, N-h+b)dr \\
&= \mathbb{E}[\mathrm{Beta}(r|h+a, N-h+b)] \\
&= \frac{h+a}{N+a+b} = \boxed{\frac{\#\mathrm{heads}+a}{\#\mathrm{heads}+\#\mathrm{tails}+a+b}}
\end{aligned}
$$

the mean of the Beta distribution

- Instead of taking one parameter value, average over all of them
- a, b, again interpretable as effective # observations
- **Consider the difference if a=b=1, #heads=1, #tails=0**

Simple example: coin toss

- **<u>Step 4</u>**: Make prediction!

likelihood

posterior

$$p(x=1|\mathcal{D}) = \int_0^1 p(x=1|r)p(r|\mathcal{D})dr$$
$$= \int_0^1 r \cdot \text{Beta}(r|h+a, N-h+b)dr$$
$$= \mathbb{E}[\text{Beta}(r|h+a, N-h+b)]$$
$$= \frac{h+a}{N+a+b} = \boxed{\frac{\#\text{heads}+a}{\#\text{heads}+\#\text{tails}+a+b}}$$

the mean of the Beta distribution

- Instead of taking one parameter value, average over all of them
- a, b, again interpretable as effective # observations
- **Note that as #flips increases, prior starts to matter less**

Takeaways

- **Instead of predicting using one parameter value, average over all of them**
  - True for all Bayesian models
- **Hyperparameters interpretable as effective # observations**
  - True for many Bayesian models (depends on parametrization)
- **As amount of data increases, prior starts to matter less**
  - True for all Bayesian models

Example 2: mean of a 1d Gaussian

- Try to learn the **mean** μ of a Gaussian distribution that generated some real number. e.g. *D = {0.3427}*
- Note: still no *x,* only *y*
- Model:
  - **Step 1:** Likelihood $p(y) = \mathcal{N}(\mu, \sigma^2)$
  - **Step 2:** Conjugate prior $p(\mu) = \mathcal{N}(0, \alpha^{-1})$
- Assume **variances of the distributions are known** (σ, α)
- Prior: we know the mean is close to zero but not its exact value

Example 2: inference for Gaussian

- Calculation is slightly easier to carry out in log space
  - log likelihood: $$\text{const} - \frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}$$
  - log conjugate prior: $$\text{const} - \frac{1}{2}\mu^2\alpha$$
- **Step 3:** calculate posterior distribution (in log space) $\log p(\mu|\mathcal{D})$

Inference for Gaussian

$$\log p(\mu|\mathcal{D}) = \log p(\mu) + \log p(\mathcal{D}|\mu) + \text{const}$$

$$\text{const} - \frac{1}{2}\left(\frac{(y-\mu)^2}{\sigma^2} + \mu^2\alpha\right)$$

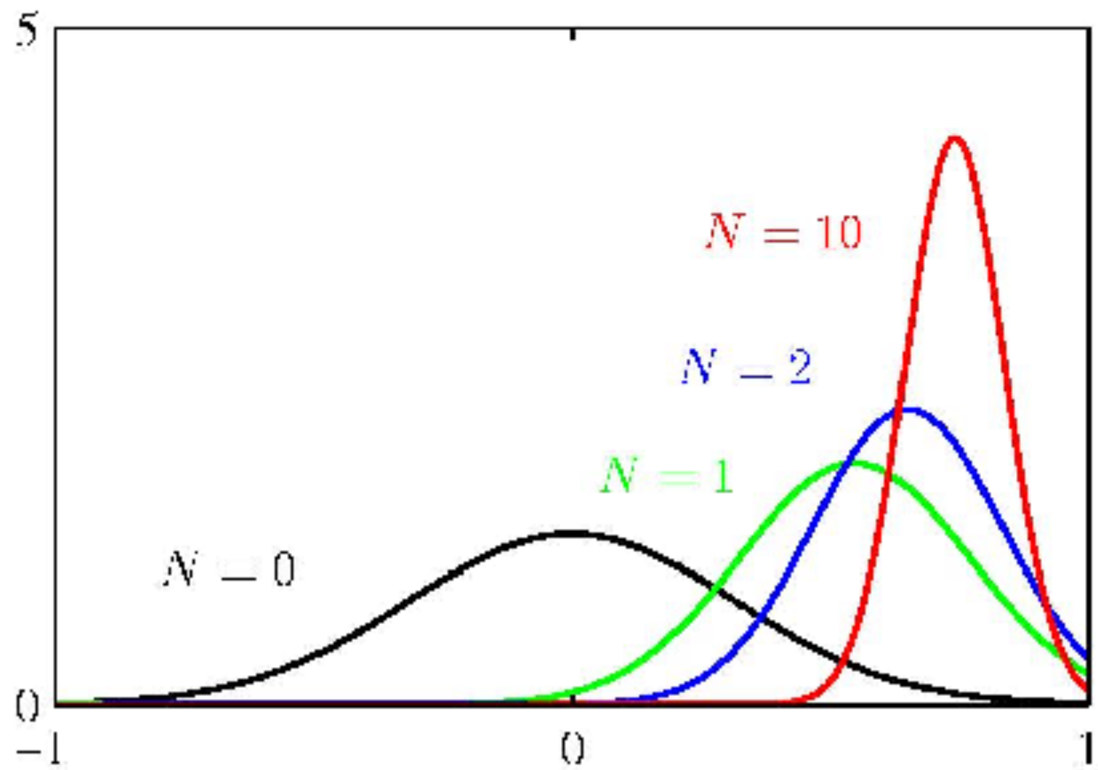
$$\begin{aligned}
\frac{(y-\mu)^2}{\sigma^2} + \mu^2\alpha &= -2\frac{y\mu}{\sigma^2} + \frac{\mu^2}{\sigma^2} + \mu^2\alpha + \text{const} \\
&= -2\frac{y\mu}{\sigma^2} + (\alpha + \sigma^{-2})\mu^2 + \text{const} \\
&= -2\frac{\alpha+\sigma^{-2}}{\alpha+\sigma^{-2}}\frac{1}{\sigma^2}y\mu + (\alpha+\sigma^{-2})\mu^2 + \text{const} \\
&= \frac{\left(\frac{\sigma^{-2}}{\alpha+\sigma^{-2}}y - \mu\right)^2}{(\alpha+\sigma^{-2})^{-1}} + \text{const}
\end{aligned}$$

<u>Step 3:</u> calculate
$\log p(\mu|\mathcal{D})$

mean of posterior distribution of μ: between MLE (y) and prior (0)

covariance of posterior: smaller than either covariance of likelihood or prior

Inference for Gaussian



Prediction for Gaussian

- **Step 4:** make prediction

$$p(y^*|\mathcal{D}) = \int_{-\infty}^{\infty} p(y^*, \mu|\mathcal{D}) d\mu$$

$$= \int_{-\infty}^{\infty} p(y^*|\mu) p(\mu|\mathcal{D}) d\mu$$

$$= \int_{-\infty}^{\infty} \mathcal{N}(y^*|\mu, \sigma^2) \mathcal{N}\left(\mu \,\middle|\, \frac{\sigma^{-2}}{\alpha + \sigma^{-2}} y_{\text{train}}, \frac{1}{\alpha + \sigma^{-2}}\right) d\mu$$

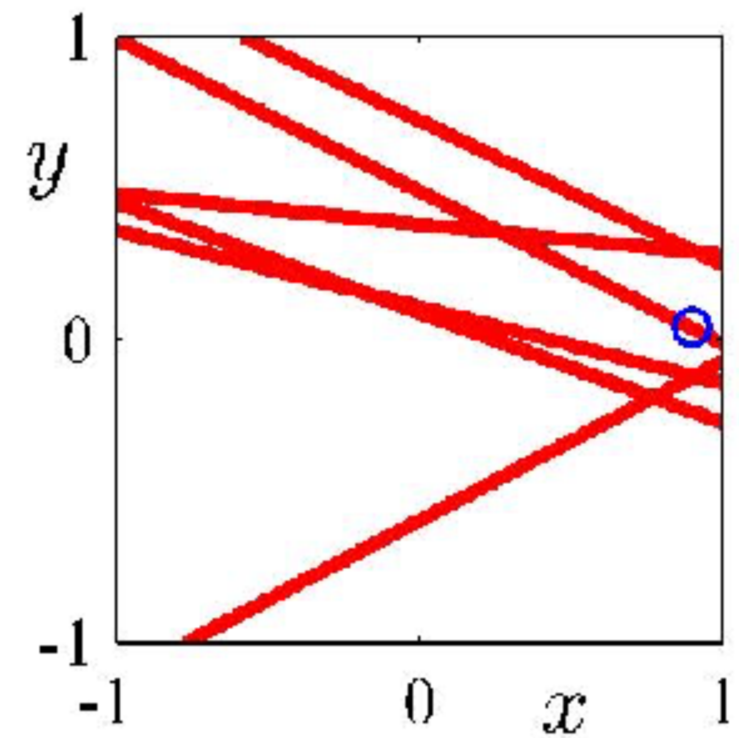
- Convolution of Gaussians, can be solved in closed form

$$p(y^*|\mathcal{D}) = \mathcal{N}\left(y^* \,\middle|\, \frac{\sigma^{-2}}{\alpha + \sigma^{-2}} y_{\text{train}}, \sigma^2 + \frac{1}{\alpha + \sigma^{-2}}\right)$$

noise + parameter uncertainty

Bayesian vs. frequentist

- Can we quantify uncertainty over models using probabilities?
- **Classical / frequentist statistics: no**
  - Probability represents *frequency of repeatable event*
  - There is only one true model
  - Do not consider 'prior knowledge'

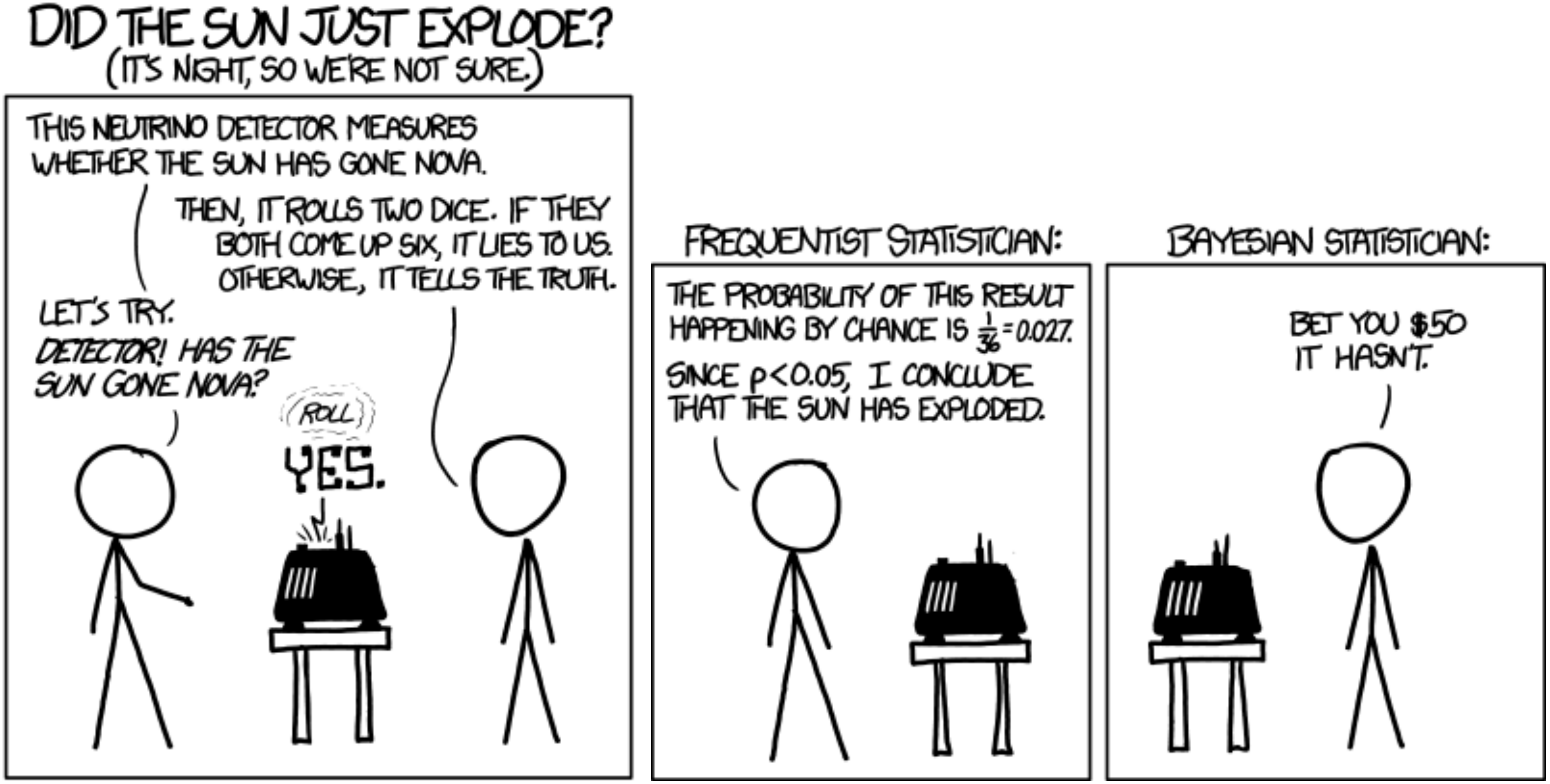


Copyright C.M. Bishop, PRML

Bayesian probabilities

- Note: **that Bayes' theorem is used does not mean a method uses a Bayesian view on probabilities!**
- Bayes' theorem is a consequence of the sum and product rules of probability
- Many frequentist methods refer to Bayes' theorem (naive Bayes, Bayesian networks)
- Bayesian view on probability: **Can represent uncertainty** (in parameters) **using probability**

Bayesian probabilities



Randall Munroe / xkcd.com

Inference vs. Learning

- Different (overlapping!) communities use different terminology, can be confusing
- In *traditional machine learning*:
  - **Learning:** adjusting the parameters of your model to fit the data (by optimization of some cost function)
  - **Inference:** given your model + parameters and some data, make some prediction (e.g. the class of an input image)
- In *Bayesian statistics*, inference is to say something about the process that generated some data **(includes parameter estimation)**
- Take-away: in an ML problem, we can find a good value of params by optimization (*learning*) or calculate a distribution over params (*inference*)

Why Bayesian probabilities?

- **Maximum likelihood estimates can have large variance**
  - Overfitting in e.g. linear regression models
  - MLE of coin flip probabilities with three sequential ‘heads’

Why Bayesian probabilities?

- Maximum likelihood estimates can have large variance
- **We might desire or need an estimate of uncertainty**
  - Use uncertainty in decision making

    Knowing uncertainty important for many loss functions
  - Use uncertainty to decide which data to acquire

    (active learning, experimental design)

Why Bayesian probabilities?

- Maximum likelihood estimates can have large variance
- We might desire or need an estimate of uncertainty
- **Have small dataset, unreliable data, or small batches of data**
  - Account for reliability of different pieces of evidence
  - Possible to update posterior incrementally with new data
  - Variance problem especially bad with small data sets

Why Bayesian probabilities?

- Maximum likelihood estimates can have large variance
- We might desire or need an estimate of uncertainty
- Have small dataset, unreliable data, or small batches of data
- **Use prior knowledge in a principled fashion**

Why Bayesian probabilities?

- Maximum likelihood estimates can have large variance
- We might desire or need an estimate of uncertainty
- Have small dataset, unreliable data, or small batches of data
- Use prior knowledge in a principled fashion
- **In practice, using prior knowledge and uncertainty particularly makes difference with small data sets**

Why not Bayesian probabilities?

- Prior induces bias
- Misspecified priors: if prior is wrong, posterior can be far off
- Prior often chosen for mathematical convenience, not actually knowledge of the problem
- In contrast to frequentist probability, uncertainty is subjective, different between different people / agents

What you should know

- What is the Bayesian view of probability?
- Why can the Bayesian view be beneficial?
- What are the general inference and prediction steps?
- Role of the following distributions:
  - Likelihood, prior, posterior, posterior predictive
- How can posterior and posterior predictive distribution be used?