A Brief Intro to Bayesian Inference

Thomas Bayes (c. 1701 – 1761)

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

CSC411: Machine Learning and Data Mining, Winter 2017

Michael Guerzhoy

Tossing a Coin

- Suppose the coin came up Heads 65 times and Tails 35 times. Is the coin fair?
- Model: $P(heads) = \theta$
- Log-likelihood: $\log P(data|\theta) = 65 \log \theta + 35 \log(1 - \theta)$
  - Maximized at $\theta =$ .65
- But would you conclude that the coin really is not fair?

Prior Distributions

- We can encode out beliefs about what the values of the parameters could be using

$$P(\theta)$$

- Using Bayes' rule, we have

$$P(\theta = \theta_0|\text{data}) = \frac{\mathrm{P}(\theta=\theta_0, data)}{P(data)} = \frac{\overset{\text{likelihood}}{P(data|\theta = \theta_0)}\overset{\text{Prior}}{P(\theta=\theta_0)}}{\underset{\uparrow}{P(data)}}$$

$$= \sum_{\theta_1} P(data|\,\theta = \theta_1)\,P(\theta = \theta_1)$$

Maximum a-posteriori (MAP)

- Maximize the *posterior probability* of the parameter:

$$argmax_{\theta_0} \frac{P(data|\theta = \theta_0)P(\theta=\theta_0)}{P(data)}$$

$$= argmax_{\theta_0} P(data|\theta = \theta_0)P(\theta = \theta_0)$$

$$= argmax_{\theta_0} \log P(data|\theta = \theta_0) + \log P(\theta = \theta_0)$$

- The posterior of probability is the product of the prior and the data likelihood
- Represents the *updated* belief about the parameter, given the observed data

Aside: Bayesian Inference is a Powerful Idea

- You can think about anything like that. You have your prior belief $P(\theta)$, and you observe some new data. Now your belief about θ *must be* proportional to $P(\theta)P(data|\theta)$
  - But only if you are 100% sure that the likelihood function is correct!
  - Recall that the likelihood function is your model of the world – it represents knowledge about how the data is generated for given values of θ
  - Where do you get your original prior beliefs anyway?
- Arguably, makes more sense than Maximum Likelihood

Back to the Coin

- (In Python)

Gaussian Residuals Models

- Log-likelihood:

$$logP(data|\theta) = \sum -\frac{\left(y^{(i)} - \theta^T x^{(i)}\right)^2}{2\sigma^2} - \frac{\mathrm{m}}{2}\log(2\pi\sigma^2)$$

- Suppose we believe that $P(\theta_i) = N\left(0, \left(\frac{1}{2\lambda}\right)\right)$
  - I.e., the coefficients in θ will generally be in $[-1.5/\lambda, 1.5/\lambda]$
- $\log[P(data|\theta)P(\theta)]$ is $\log P(data|\theta) - \lambda|\theta|^2 + const$ (exercise)
- Maximize $\log[P(data|\theta)P(\theta)]$ to get the θ that you believe the most

Why $P(\theta_i) = N\left(0, \left(\frac{1}{2\lambda}\right)\right)$

- More on this later
- If the θ_i’s are allowed to be arbitrarily large, the ratio of the influences of the different features over the decision boundary could be arbitrarily high
  - Difficult to believe that one of the features still matters, but it matters a 10000000 times less than some other feature
    - Easy to believe a feature doesn’t matter at all, though
    - Only reasonable if the inputs are all on the same scale, and the output is on roughly the same scale as the inputs
  - Mostly when we fit coefficients, they don’t get crazy high, so it’s a reasonable prior belief