A Brief Intro to Bayesian Inference

\[ P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)} \]

Thomas Bayes (c. 1701 – 1761)
Tossing a Coin

• Suppose the coin came up Heads 65 times and Tails 35 times. Is the coin fair?

• Model: \( P(\text{heads}) = \theta \)

• Log-likelihood:  
  \[
  \log P(\text{data}|\theta) = 55 \log \theta + 45 \log(1 - \theta)
  \]
  • 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

\[
P(\theta = \theta_0|\text{data}) = \frac{P(\theta = \theta_0, \text{data})}{P(\text{data})} = \frac{P(\text{data}|\theta = \theta_0)P(\theta = \theta_0)}{P(\text{data})}
\]

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

Maximum a-posteriori (MAP)

• Maximize the *posterior probability* of the parameter:

\[
\arg \max_{\theta_0} \frac{P(\text{data}|\theta = \theta_0)P(\theta = \theta_0)}{P(\text{data})}
\]

\[
= \arg \max_{\theta_0} P(\text{data}|\theta = \theta_0)P(\theta = \theta_0)
\]

\[
= \arg \max_{\theta_0} \log P(\text{data}|\theta = \theta_0) + \log P(\theta = \theta_0)
\]

• 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 $\theta$ 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 $\theta$
  • Where do you get your original prior beliefs anyway?
• Arguably, makes more sense than Maximum Likelihood
Back to the Coin

• (See bayes.py)
Gaussian Residuals Models

• Log-likelihood:

\[
\log P(data|\theta) = \sum -\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2} + 2m/\log(2\pi\sigma^2)
\]

• Suppose we believe that \( P(\theta_i) = N \left(0, \left(\frac{1}{2\lambda}\right)\right) \)
  
  • I.e., the coefficients in \( \theta \) will generally be in \([-1.5\lambda, 1.5\lambda]\)

• \( \log[P(data|\theta)P(\theta)] \) is \( \log P(data|\theta) + \lambda|\theta|^2 + \text{const} \) (exercise)

• Maximize \( \log[P(data|\theta)P(\theta)] \) to get the \( \theta \) 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 $\theta_i$’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
  • Mostly when we fit coefficients, they don’t get crazy high, so it’s a reasonable prior belief