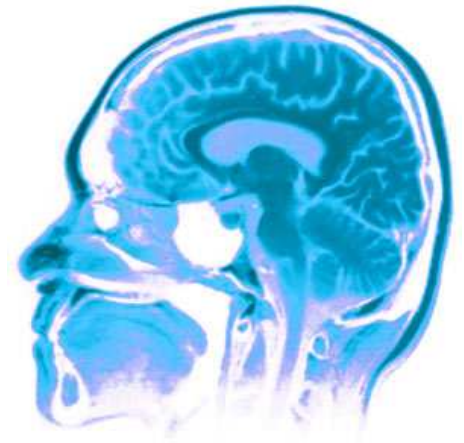




CPSC540



Bayesian learning



Nando de Freitas

January, 2013

University of British Columbia

Outline of the lecture

This lecture introduces **Bayes rule** and Bayesian learning for linear models.

The goal is for you to:

- ☐ Learn how Bayes rule is derived.
- ☐ Learn to apply Bayes rule to simple examples.
- ☐ Learn how to apply Bayesian learning to linear models.
- ☐ Learn the mechanics of conjugate analysis.

Problem 1: Diagnoses



- ☐ The doctor has bad news and good news.
- ☐ The bad news is that you tested positive for a serious disease, and that **the test is 99% accurate** (i.e., the probability of testing positive given that you have the disease is 0.99, as is the probability of testing negative given that you don't have the disease).
- ☐ The good news is that this is a rare disease, striking only 1 in 10,000 people.
- ☐ What are the chances that you actually have the disease?



Bayes rule

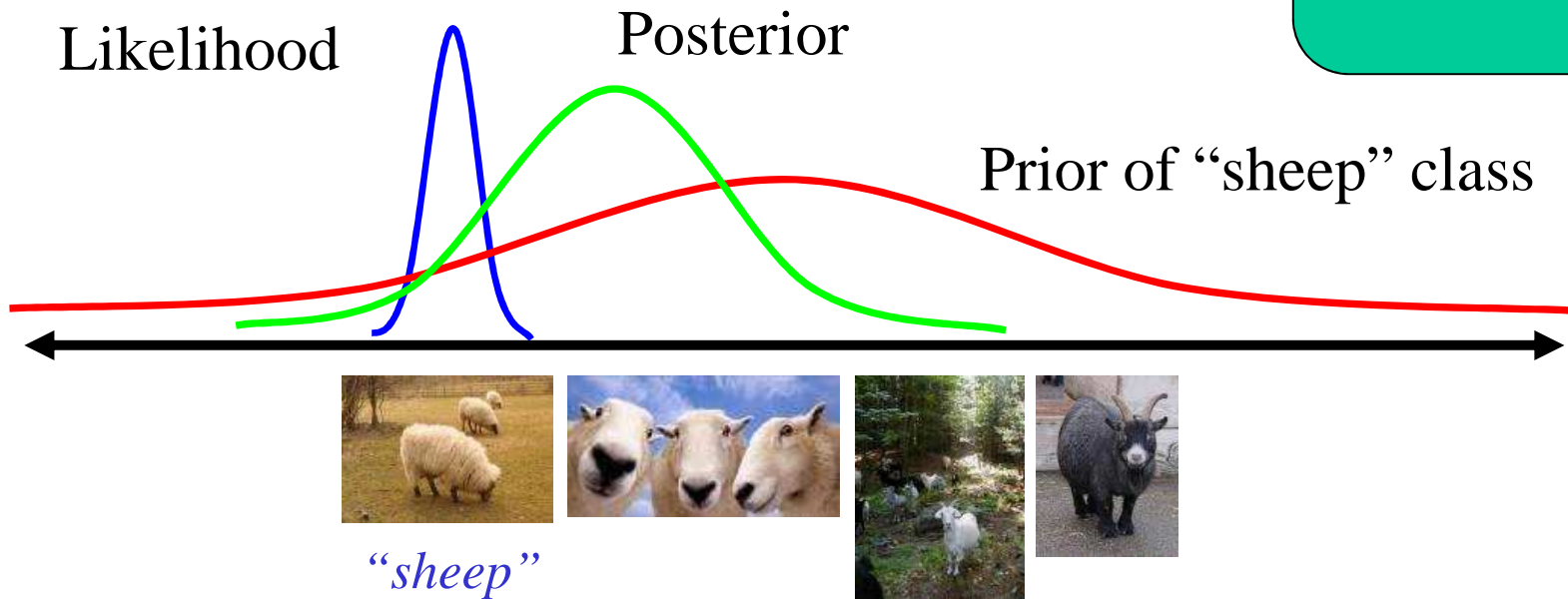
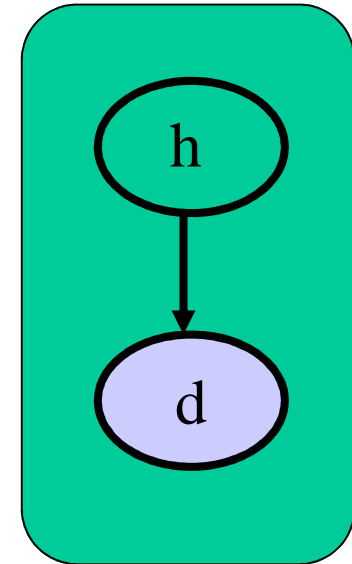
Bayes rule enables us to reverse probabilities:

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



Learning and Bayesian inference

$$p(h | d) = \frac{p(d | h) p(h)}{\sum_{h' \in H} p(d | h') p(h')}$$



Problem 1: Diagnoses

The test is 99% accurate: $P(T=1/D=1) = 0.99$ and $P(T=0/D=0) = 0.99$
Where T denotes test and D denotes disease.

The disease affects 1 in 10000: $P(D=1) = 0.0001$

$$P(D=1 | T=1) = \frac{P(T=1 | D=1) P(D=1)}{P(T=1 | D=0) P(D=0) + P(T=1 | D=1) P(D=1)}$$
$$\approx$$

Speech recognition

$$P(\text{words} \mid \text{sound}) \propto P(\text{sound} \mid \text{words}) P(\text{words})$$

Final beliefs

Likelihood of data

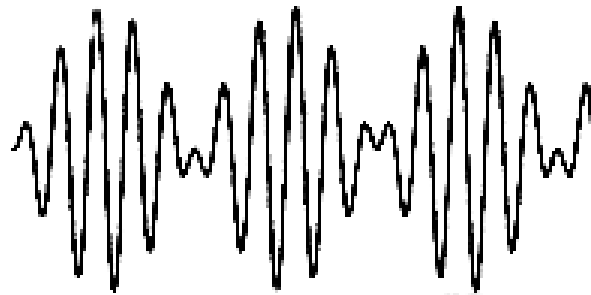
Prior language model

eg mixture of Gaussians

eg unigrams

Hidden Markov Model (HMM)

“Recognize speech”



“Wreck a nice beach”



Bayesian learning for model parameters

Step 1: Given n data, $\mathbf{D} = \mathbf{x}_{1:n} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$, write down the expression for the *likelihood*:

$$p(\mathbf{D} | \boldsymbol{\theta})$$

Step 2: Specify a *prior*: $p(\boldsymbol{\theta})$

Step 3: Compute the *posterior*:

$$p(\boldsymbol{\theta} | \mathbf{D}) = \frac{p(\mathbf{D} | \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathbf{D})}$$

Bayesian linear regression

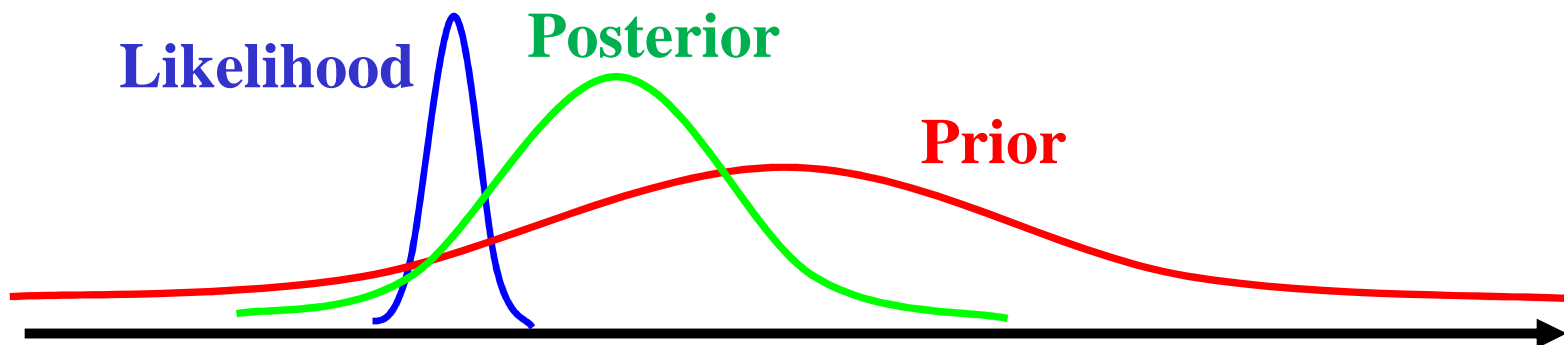
The likelihood is a Gaussian, $\mathcal{N}(\mathbf{y}|\mathbf{X}\boldsymbol{\theta}, \sigma^2\mathbf{I}_n)$. The conjugate prior is also a Gaussian, which we will denote by $p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_0, \mathbf{V}_0)$.

Using Bayes rule for Gaussians, the posterior is given by

$$p(\boldsymbol{\theta}|\mathbf{X}, \mathbf{y}, \sigma^2) \propto \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_0, \mathbf{V}_0)\mathcal{N}(\mathbf{y}|\mathbf{X}\boldsymbol{\theta}, \sigma^2\mathbf{I}_n) = \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\theta}_n, \mathbf{V}_n)$$

$$\boldsymbol{\theta}_n = \mathbf{V}_n \mathbf{V}_0^{-1} \boldsymbol{\theta}_0 + \frac{1}{\sigma^2} \mathbf{V}_n \mathbf{X}^T \mathbf{y}$$

$$\mathbf{V}_n^{-1} = \mathbf{V}_0^{-1} + \frac{1}{\sigma^2} \mathbf{X}^T \mathbf{X}$$



Bayesian linear regression

Assume σ^2 is known.

$$P(\theta | x, y, \sigma^2) \propto P(y | x, \theta, \sigma^2) P(\theta)$$

$$\propto e^{-\frac{1}{2}(y - x\theta)^T (\sigma^2 I)^{-1} (y - x\theta)} e^{-\frac{1}{2}(\theta - \theta_0)^T V_0^{-1} (\theta - \theta_0)}$$

$$= e^{-\frac{1}{2} \left\{ y^T (\sigma^2 I)^{-1} y - 2 y^T (\sigma^2 I)^{-1} x \theta + \theta^T x^T (\sigma^2 I)^{-1} x \theta + \theta^T V_0^{-1} \theta + \theta_0^T V_0^{-1} \theta_0 - 2 \theta_0^T V_0^{-1} \theta \right\}}$$

$$= e^{-\frac{1}{2} \left\{ \text{const} + \underbrace{\theta^T (x^T (\sigma^2 I)^{-1} x + V_0^{-1}) \theta}_{\text{Call this } V_n^{-1}} - 2 (y^T (\sigma^2 I)^{-1} x + \theta_0^T V_0^{-1}) \theta \right\}}$$

$$= e^{-\frac{1}{2} \left\{ \text{const} + \theta^T V_n^{-1} \theta - 2 \left(\frac{y^T x}{\sigma^2} + \theta_0^T V_0^{-1} \right) \theta \right\}}$$

$$= e^{-\frac{1}{2} \left\{ \text{const} + \theta^T V_n^{-1} \theta - 2 \theta_n^T V_n^{-1} \theta + 2 \theta_n^T V_n^{-1} \theta - 2 \left(\frac{y^T x}{\sigma^2} + \theta_0^T V_0^{-1} \right) \theta \right\}}$$

$$= e^{-\frac{1}{2} \left\{ \text{const}_2 + (\theta - \theta_n)^T V_n^{-1} (\theta - \theta_n) + 2 \left[\theta_n^T V_n^{-1} - \frac{y^T x}{\sigma^2} - \theta_0^T V_0^{-1} \right] \theta \right\}}$$

Bayesian linear regression

$$\Theta_n^T V_n^{-1} - \frac{y^T x}{G^2} - \Theta_0^T V_0^{-1} = 0 \quad \text{when } \Theta_n = V_n \left[V_0^{-1} \Theta_0 + \frac{x^T y}{G^2} \right]$$

and when this happens, we have:

$$P(\theta | x, y, G^2) \propto e^{-\frac{1}{2} (\theta - \Theta_n)^T V_n^{-1} (\theta - \Theta_n)}$$

By the definition of a multivariate Gaussian, we have:

$$\int e^{-\frac{1}{2} (\theta - \Theta_n)^T V_n^{-1} (\theta - \Theta_n)} d\theta = |2\pi V_n|^{1/2}$$

$$\therefore P(\theta | x, y, G^2) = |2\pi V_n|^{-1/2} e^{-\frac{1}{2} (\theta - \Theta_n)^T V_n^{-1} (\theta - \Theta_n)}$$



Bayesian linear regression

Consider the special case where $\boldsymbol{\theta}_0 = \mathbf{0}$ and $\mathbf{V}_0 = \tau_0^2 \mathbf{I}_d$, which is a spherical Gaussian prior. Then the posterior mean reduces to

$$\begin{aligned}\boldsymbol{\theta}_n &= \frac{1}{\sigma^2} \mathbf{V}_N \mathbf{X}^T \mathbf{y} = \frac{1}{\sigma^2} \left(\frac{1}{\tau_0^2} \mathbf{I}_d + \frac{1}{\sigma^2} \mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T \mathbf{y} \\ &= (\lambda \mathbf{I}_d + \mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}\end{aligned}$$

where we have defined $\lambda := \frac{\sigma^2}{\tau_0^2}$. We have therefore recovered **ridge regression** again!

Bayesian versus ML plugin prediction

$$\text{Posterior mean: } \boldsymbol{\theta}_n = (\lambda \mathbf{I}_d + \mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

$$\text{Posterior variance: } \mathbf{V}_n = \sigma^2 (\lambda \mathbf{I}_d + \mathbf{X}^T \mathbf{X})^{-1}$$

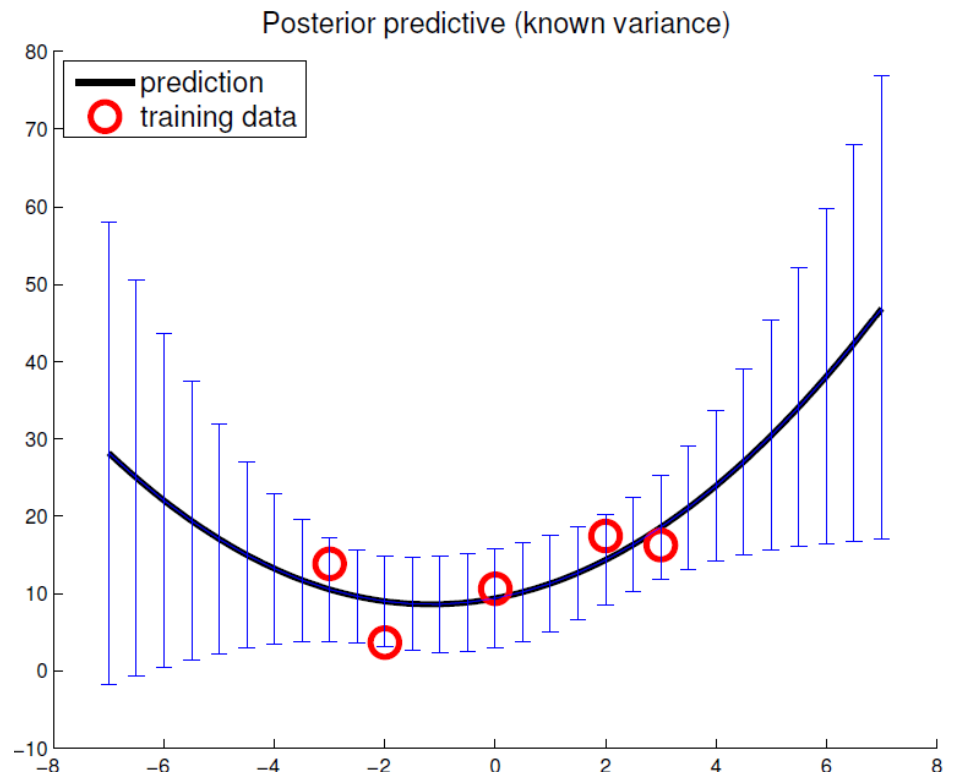
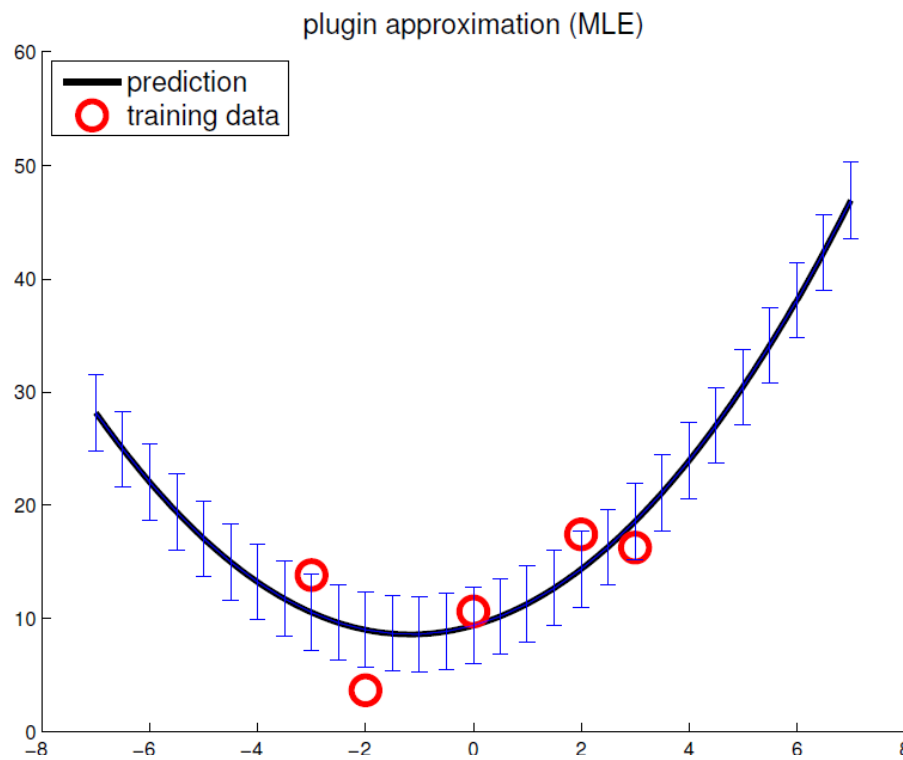
To predict, Bayesians marginalize over the posterior. Let \mathbf{x}_* be a new input. The prediction, given the training data $\mathbf{D}=(\mathbf{X}, \mathbf{y})$, is:

$$\begin{aligned} P(\mathbf{y} | \mathbf{x}_*, \mathbf{D}, \sigma^2) &= \int \mathcal{N}(\mathbf{y} | \mathbf{x}_*^T \boldsymbol{\theta}, \sigma^2) \mathcal{N}(\boldsymbol{\theta} | \boldsymbol{\theta}_n, \mathbf{V}_n) d\boldsymbol{\theta} \\ &= \mathcal{N}(\mathbf{y} | \mathbf{x}_*^T \boldsymbol{\theta}_n, \sigma^2 + \mathbf{x}_*^T \mathbf{V}_n \mathbf{x}_*) \end{aligned}$$

On the other hand, the ML plugin predictor is:

$$P(\mathbf{y} | \mathbf{x}_*, \mathbf{D}, \sigma^2) = \mathcal{N}(\mathbf{y} | \mathbf{x}_*^T \boldsymbol{\theta}_{ML}, \sigma^2)$$

Bayesian versus ML plug-in prediction



Next lecture

In the next lecture, we extend Bayesian learning to nonlinear problems via a technique known as Gaussian processes.