430.457

Introduction to Intelligent Systems

Prof. Songhwai Oh

ECE, SNU

LEARNING WITH COMPLETE DATA

ML Learning

- Candy bag problem (cherry and lime candies in a bag)
- Parameter: $\theta \in [0, 1]$, the proportion of cherry candies.
- Hypothesis: h_θ.
- Suppose we unwrap N candies, of which c are cherries and $l = N - c$ are limes.
- Likelihood:

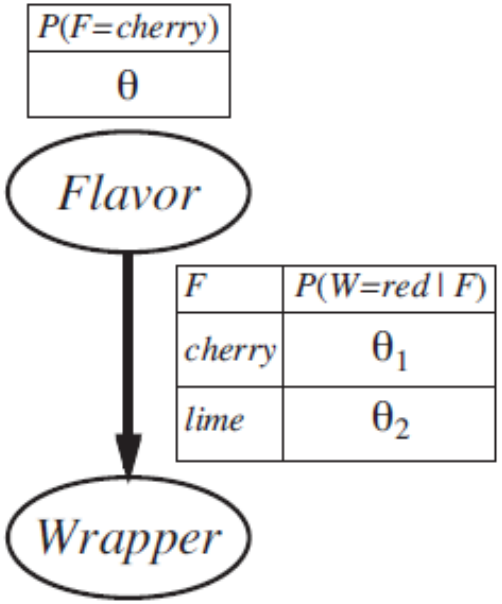
$$P(\mathbf{d}|h_\theta) = \prod_{j=1}^{N} P(d_j|h_\theta) = \theta^c \cdot (1-\theta)^l.$$

- Log likelihood:

$$L(\mathbf{d}|h_\theta) = \log P(\mathbf{d}|h_\theta) = \sum_{j=1}^{N} \log P(d_j|h_\theta) = c\log\theta + l\log(1-\theta).$$

- ML hypothesis $=$ arg max $L(\mathbf{d}|h_\theta)$.

$$\frac{d}{d\theta}L(\mathbf{d}|h_\theta) = \frac{c}{\theta} - \frac{l}{1-\theta} = 0 \qquad \Rightarrow \qquad \theta = \frac{c}{c+l} = \frac{c}{N}.$$



- $Flavor \in \{cherry, lime\}$
- $Wrapper \in \{red, green\}$

$$
\begin{aligned}
&P(Flavor = cherry, Wrapper = green \mid h_{\theta,\theta_1,\theta_2}) \\
&\quad = P(Flavor = cherry \mid h_{\theta,\theta_1,\theta_2}) P(Wrapper = green \mid Flavor = cherry, h_{\theta,\theta_1,\theta_2}) \\
&\quad = \theta \cdot (1 - \theta_1) \, .
\end{aligned}
$$

From N candies, wrapper counts are as follows: r_c of cherries have red wrappers and g_c have green, while r_l of limes have red and g_l have green.

$$P(\mathbf{d} \mid h_{\theta,\theta_1,\theta_2}) = \theta^c (1-\theta)^\ell \cdot \theta_1^{r_c} (1-\theta_1)^{g_c} \cdot \theta_2^{r_\ell} (1-\theta_2)^{g_\ell}$$

$$L = [c \log \theta + \ell \log(1-\theta)] + [r_c \log \theta_1 + g_c \log(1-\theta_1)] + [r_\ell \log \theta_2 + g_\ell \log(1-\theta_2)]$$

$$
\begin{aligned}
\frac{\partial L}{\partial \theta} &= \frac{c}{\theta} - \frac{\ell}{1-\theta} = 0 & \Rightarrow \quad \theta &= \frac{c}{c+\ell} \\
\frac{\partial L}{\partial \theta_1} &= \frac{r_c}{\theta_1} - \frac{g_c}{1-\theta_1} = 0 & \Rightarrow \quad \theta_1 &= \frac{r_c}{r_c+g_c} \\
\frac{\partial L}{\partial \theta_2} &= \frac{r_\ell}{\theta_2} - \frac{g_\ell}{1-\theta_2} = 0 & \Rightarrow \quad \theta_2 &= \frac{r_\ell}{r_\ell+g_\ell}
\end{aligned}
$$

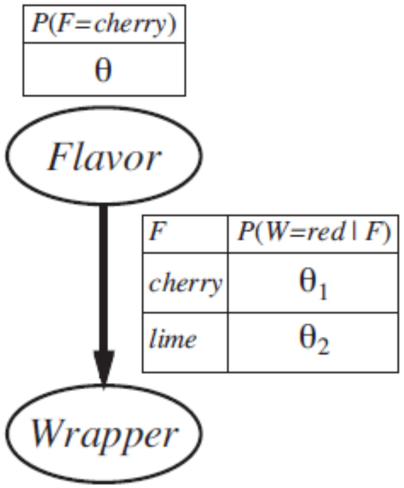
- With complete data, the maximum-likelihood parameter learning problem for a Bayesian network decomposes into separate learning problems, one for each parameter.

Finding the ML hypothesis

1. Write down an expression for the likelihood of the data as a function of the parameter(s).
2. Write down the derivative of the log likelihood with respect to each parameter.
3. Find the parameter values such that the derivatives are zero.

Naïve Bayes Models

$$\mathbf{P}(Cause, Effect_1, \ldots, Effect_n) = \mathbf{P}(Cause) \prod_i \mathbf{P}(Effect_i \mid Cause)$$



Class: Flavor
Attributes: Wrapper

We can learn parameters of a Naïve Bayes model using the ML method as before.

Classify a new example by choosing the most likely class by computing

$$\mathbf{P}(C \mid x_1, \ldots, x_n) = \alpha\, \mathbf{P}(C) \prod_i \mathbf{P}(x_i \mid C)$$

ML Learning: Continuous Model

- Learning the parameters of a Gaussian density function on a single variable.
- Gaussian density function:

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right).$$

- Parameters: the mean μ and the standard deviation σ.
- Given observations $x_1, \ldots, x_N$, the log likelihood is

$$L = \sum_{j=1}^{N} \log \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_j-\mu)^2}{2\sigma^2}} = N(-\log\sqrt{2\pi} - \log\sigma) - \sum_{j=1}^{N} \frac{(x_j-\mu)^2}{2\sigma^2}.$$

ML estimators:

sample mean

$$\begin{aligned} \frac{\partial L}{\partial \mu} &= -\frac{1}{\sigma^2}\sum_{j=1}^{N}(x_j-\mu) = 0 & \Rightarrow \quad \mu &= \frac{\sum_j x_j}{N} \\ \frac{\partial L}{\partial \sigma} &= -\frac{N}{\sigma} + \frac{1}{\sigma^3}\sum_{j=1}^{N}(x_j-\mu)^2 = 0 & \Rightarrow \quad \sigma &= \sqrt{\frac{\sum_j (x_j-\mu)^2}{N}} \end{aligned}$$

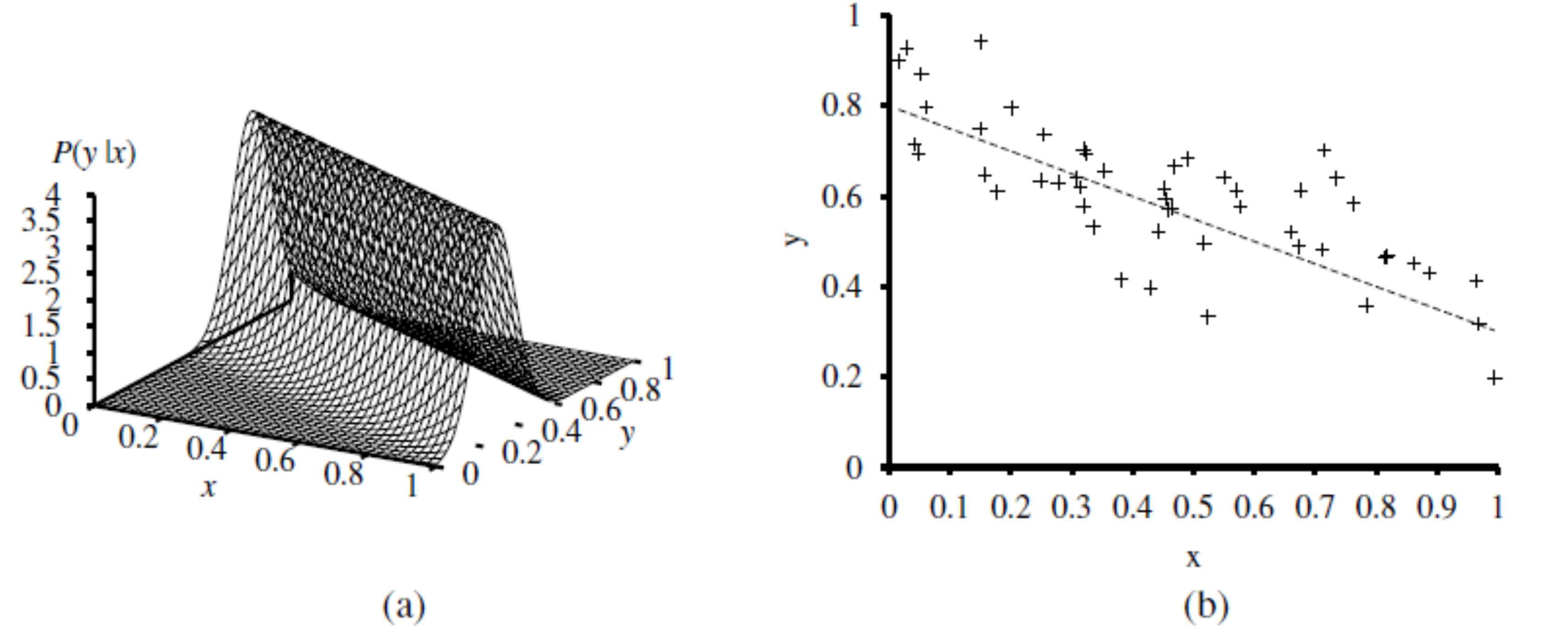


Figure 20.4 (a) A linear Gaussian model described as $y = \theta_1 x + \theta_2$ plus Gaussian noise with fixed variance. (b) A set of 50 data points generated from this model.

$$P(y \mid x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(y-(\theta_1 x+\theta_2))^2}{2\sigma^2}}$$

parameters: σ, θ_1, θ_2

Same as the linear regression (except we now consider σ)

Bayesian Parameter Learning

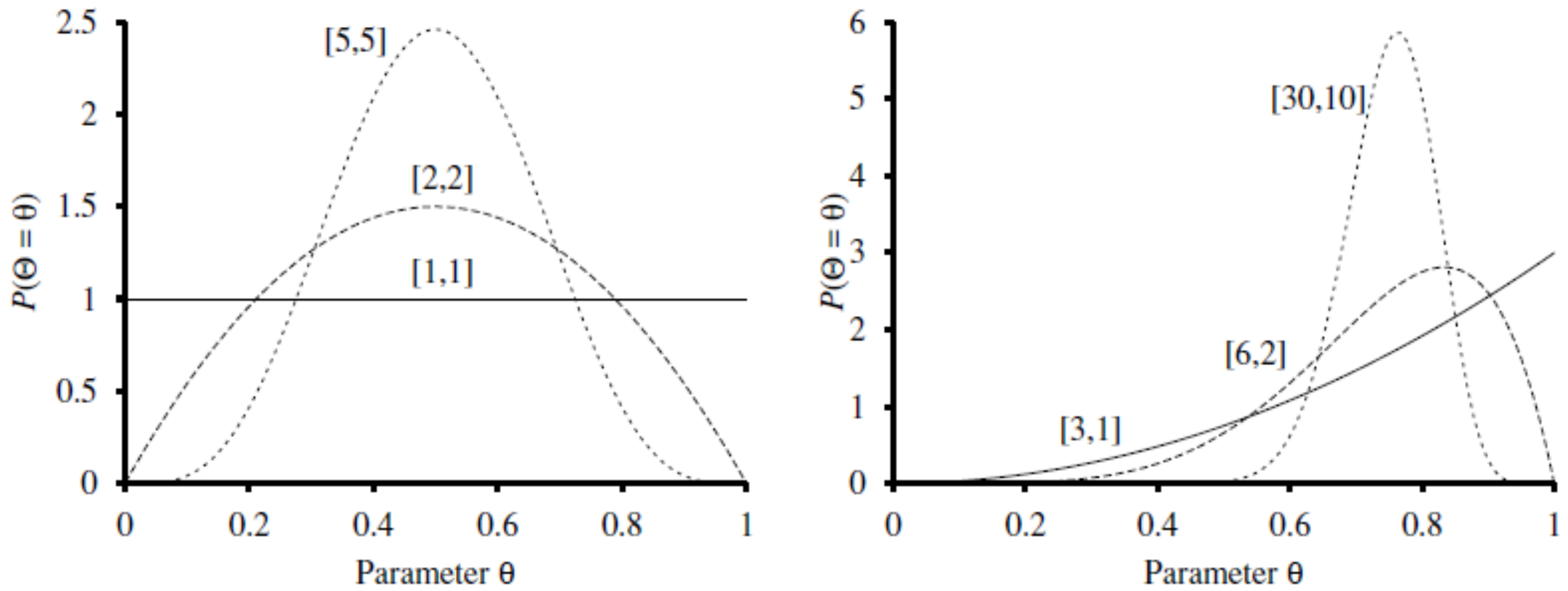
- ML learning has a problem when the data set is small.
- For example, after seeing one cherry candy, the ML hypothesis is that the bag is 100% cherry (i.e., $\theta = 1.0$). Unless one's hypothesis prior is that bags must be either all cherry or all lime, this is not a reasonable conclusion.
- The problem can avoided by assigning a prior probability distribution over the possible hypotheses.
- In our candy example, we assign a hypothesis prior $P(\Theta)$ on θ.

Beta Distribution

- Beta distribution with parameters a and b:

$$\text{beta}[a, b](\theta) = \alpha \ \theta^{a-1}(1-\theta)^{b-1}.$$

- α is a normalization constant and the mean of the distribution is $a/(a+b)$. A uniform distribution is a special case.
- a and b are called hyperparameters.



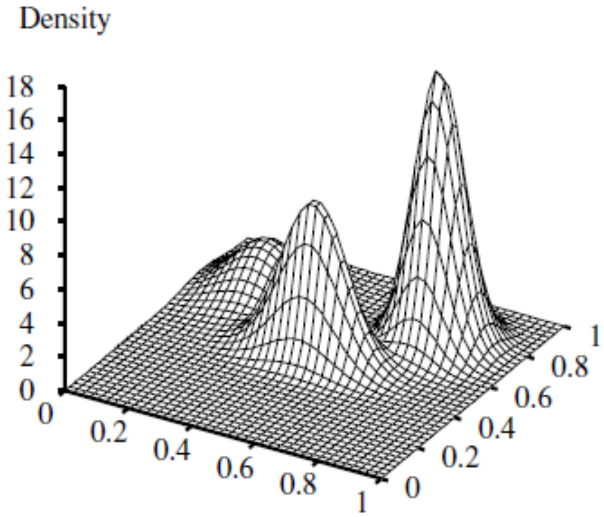
Conjugate Prior

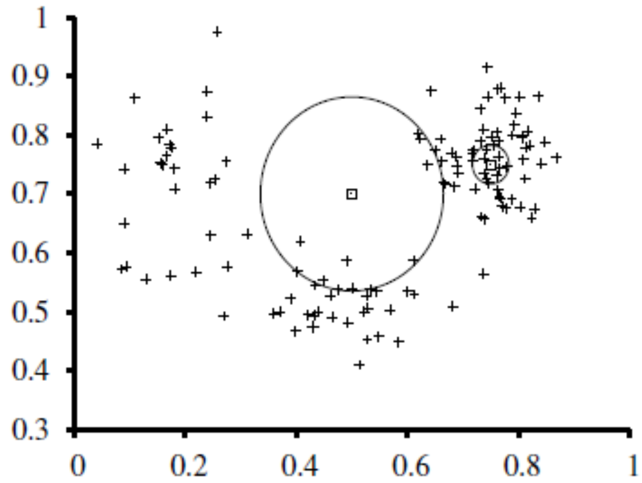
- The beta distribution is the **conjugate prior** for a Bernoulli random variable (e.g., Θ in our candy example).
- That is, if Θ has a prior beta$[a, b]$, then the posterior distribution of Θ given observations is also a beta distribution.

$$\begin{aligned} P(\theta \mid D_1 = cherry) &= \alpha\, P(D_1 = cherry \mid \theta) P(\theta) \\ &= \alpha'\, \theta \cdot \mathrm{beta}[a, b](\theta) = \alpha'\, \theta \cdot \theta^{a-1}(1-\theta)^{b-1} \\ &= \alpha'\, \theta^{a}(1-\theta)^{b-1} = \mathrm{beta}[a+1, b](\theta)\ . \end{aligned}$$

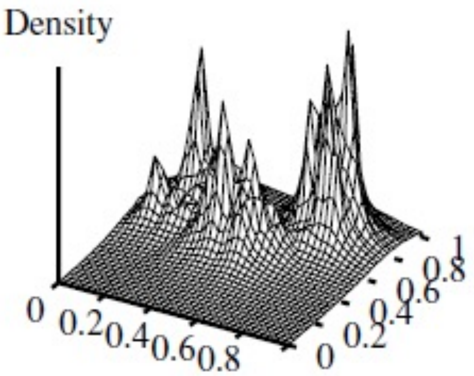
- Benefit: The posterior can be easily computed and used.
- Other examples of conjugate priors:
  - Dirichlet distribution for a multinomial random variable.
  - Gaussian distribution for the mean of a Gaussian random variable.
  - Inverse gamma distribution for the variance of a Gaussian random variable.

Density Estimation with Nonparametric Models

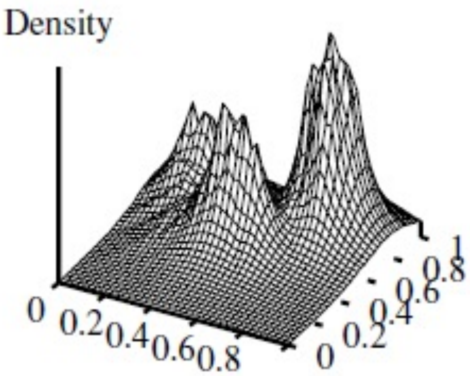




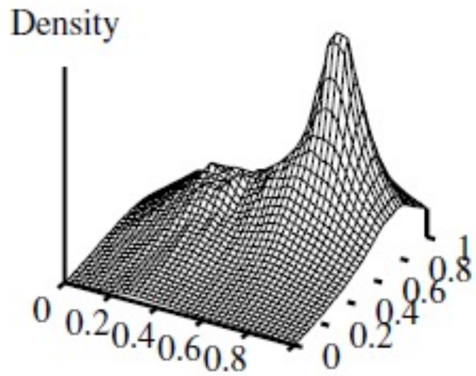
k-nearest neighbor based density estimation



k=3

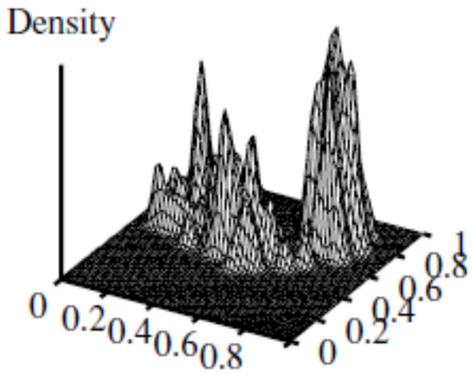


k=10

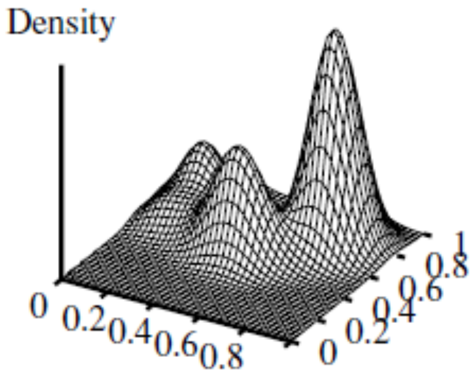


k=40

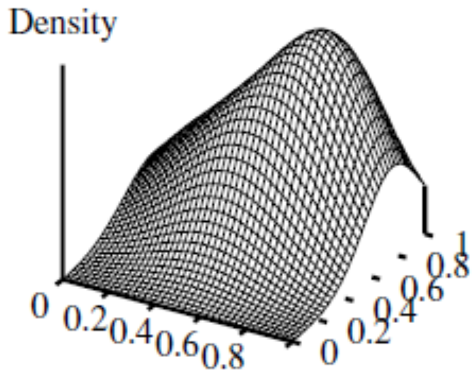
Kernel density estimation



w=0.02



w=0.07



w=0.20

$$P(\mathbf{x}) = \frac{1}{N}\sum_{j=1}^{N} \mathcal{K}(\mathbf{x}, \mathbf{x}_j)$$

Gaussian kernel

$$\mathcal{K}(\mathbf{x}, \mathbf{x}_j) = \frac{1}{(w^2\sqrt{2\pi})^d} e^{-\frac{D(\mathbf{x}, \mathbf{x}_j)^2}{2w^2}}$$