Probabilistic Graphical Models

10-708

Statistical learning with basic graphical models

Eric Xing

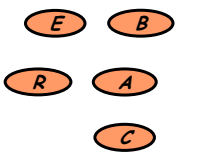
Lecture 9, Oct 10, 2005

Reading: MJ-Chap. 5,6

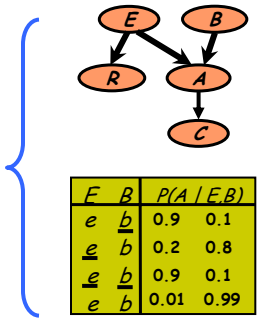
Learning Graphical Models

The goal:

Given set of independent samples (***assignments*** of random variables), find the ***best*** (the most likely?) Bayesian Network (both DAG and CPDs)



(B,E,A,C,R)=(T,F,F,T,F)
(B,E,A,C,R)=(T,F,T,T,F)
……..
(B,E,A,C,R)=(F,T,T,T,F)



Structural learning

Parameter learning

Parameter Learning

- Assume G is known and fixed,
  - from expert design
  - from an intermediate outcome of iterative structure learning
- Goal: estimate from a dataset of N independent, identically distributed (*iid*) training cases $D = \{x_1, \ldots, x_N\}$.
- In general, each training case $x_n = (x_{n,1}, \ldots, x_{n,M})$ is a vector of M values, one per node,
  - the model can be completely observable, i.e., every element in x_n is known (no missing values, no hidden variables),
  - or, partially observable, i.e., $\exists i$, s.t. $x_{n,i}$ is not observed.
- Frequentist vs. Bayesian estimate
- Initially we consider learning parameters for a single node.
- Then we consider how to learn parameters for larger GMs.

Bayesian Parameter Estimation

- Bayesians treat the unknown parameters as a random variable, whose distribution can be inferred using Bayes rule:

$$p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{p(D)} = \frac{p(D \mid \theta)p(\theta)}{\int p(D \mid \theta)p(\theta)d\theta}$$

- This crucial equation can be written in words:

$$\text{posterior} = \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}}$$

- For *iid* data, the likelihood is

$$p(D \mid \theta) = \prod_{n=1}^{N} p(x_n \mid \theta)$$

- The prior p(.) encodes our prior knowledge about the domain
  - therefore Bayesian estimation has been criticized for being "subjective"
  - empirical Bayes – fit prior from the data …

Frequentist Parameter Estimation

Two people with different priors $p(\theta)$ will end up with different estimates $p(\theta|D)$.

- Frequentists dislike this "subjectivity".
- Frequentists think of the parameter as a fixed, unknown constant, not a random variable.
- Hence they have to come up with different "objective" **estimators** (ways of computing from data), instead of using Bayes' rule.
  - These estimators have different properties, such as being "unbiased", "minimum variance", etc.
- A very popular estimator is the maximum likelihood estimator, which is simple and has good statistical properties.

Maximum Likelihood Estimation

- The log-likelihood is monotonically related to the likelihood:

$$\ell(\theta; D) = \log p(D \mid \theta) = \sum_{n=1}^{N} \log p(x_n \mid \theta)$$

- The Idea underlying maximum likelihood estimation (MLE): pick the setting of parameters most likely to have generated the data we saw:

$$\hat{\theta}_{ML} = \arg\max_{\theta} \ell(\theta; D)$$

- Problem of MLE:
  - Often the MLE **overfits** the training data, so it is common to maximize a **regularized** log-likelihood instead:

$$\hat{\theta}' = \arg\max_{\theta} \ell(\theta; D) - c(\theta)$$

  - Sometimes the MLE **underfits** the training data (e.g., certain possible values are not observed due to data sparsity), so it is common to **smooth** the estimated parameter

Being a pragmatic frequentist

- Maximum *a posteriori* (MAP) estimation:

$$\hat{\theta}_{MAP} = \arg\max_\theta p(\theta \mid D) = \arg\max_\theta \ell(\theta; D) + \log p(\theta)$$

- Smoothing with pseudo-counts
  - Recall that for Binomial Distribution, we have

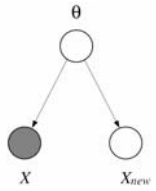
$$\hat{\theta}_{ML}^{head} = \frac{n^{head}}{n^{head} + n^{tail}}$$

  - What if we tossed too few times so that we saw zero head?
    We have $\hat{\theta}_{ML}^{head} = 0$, and we will predict that the probability of seeing a head next is zero!!!
  - The rescue: $\hat{\theta}_{ML}^{head} = \frac{n^{head} + n'}{n^{head} + n^{tail} + n'}$ **But are we still objective?**
    - Where n' is know as the pseudo- (imaginary) count

How estimators should be used?

- $\hat{\theta}_{MAP}$ is not Bayesian (even though it uses a prior) since it is a point estimate.
- Consider predicting the future. A sensible way is to combine predictions based on all possible values of θ, weighted by their posterior probability, this is what a Bayesian will do:

$$\begin{aligned} p(x_{new} \mid \mathbf{x}) &= \int p(x_{new}, \theta \mid \mathbf{x}) d\theta \\ &= \int p(x_{new} \mid \theta, \mathbf{x}) p(\theta \mid \mathbf{x}) d\theta \\ &= \int p(x_{new} \mid \theta) p(\theta \mid \mathbf{x}) d\theta \end{aligned}$$



- A frequentist will typically use a "plug-in" estimator such as ML/MAP:

$$p(x_{new} \mid \mathbf{x}) = p(x_{new} \mid \hat{\theta}_{ML}), \quad \text{or,} \quad p(x_{new} \mid \mathbf{x}) = p(x_{new} \mid \hat{\theta}_{MAP})$$

  - The Bayesian estimate will collapse to MAP for concentrated posterior

Frequentist vs. Beyesian

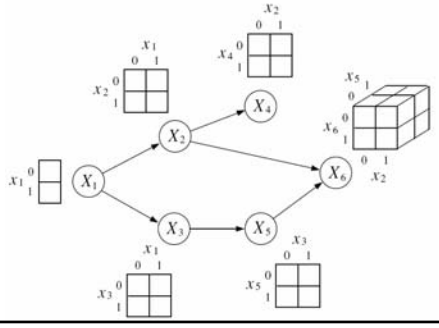
- This is a “theological” war.
- Advantages of Bayesian approach:
  - Mathematically elegant.
  - Works well when amount of data is much less than number of parameters (e.g., one-shot learning).
  - Easy to do incremental (sequential) learning.
  - Can be used for model selection (max likelihood will always pick the most complex model).
- Advantages of frequentist approach:
  - Mathematically/ computationally simpler.
  - "objective", unbiased, invariant to reparameterization
- As $|\mathcal{D}| \to \infty$, the two approaches become the same:

$$p(\theta \mid \mathcal{D}) \to \delta(\theta, \hat{\theta}_{ML})$$

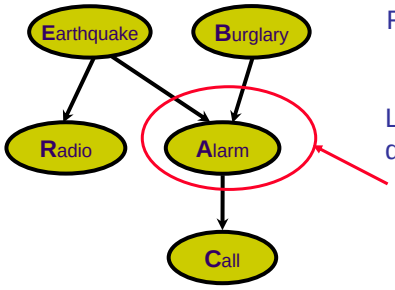
MLE for general BNs

- If we assume the parameters for each CPD are globally independent, and all nodes are fully observed, then the log-likelihood function decomposes into a sum of local terms, one per node:

$$\ell(\theta; \mathcal{D}) = \log p(\mathcal{D} \mid \theta) = \log \prod_n \left(\prod_i p(x_{n,i} \mid \mathbf{x}_{n,\pi_i}, \theta_i) \right) = \sum_i \left(\sum_n \log p(x_{n,i} \mid \mathbf{x}_{n,\pi_i}, \theta_i) \right)$$



How to define parameter prior?



Factorization: $p(\mathbf{X}=\mathbf{x}) = \prod_{i=1}^{M} p(x_i \mid \mathbf{x}_{\pi_i})$

Local Distributions
defined by, e.g., multinomial parameters:

$$p(x_i^k \mid \mathbf{x}_{\pi_i}^j) = \theta_{x_i^k \mid \mathbf{x}_{\pi_i}^j}$$

Assumptions (Geiger & Heckerman 97,99):

- Complete Model Equivalence
- Global Parameter Independence
- Local Parameter Independence
- Likelihood and Prior Modularity

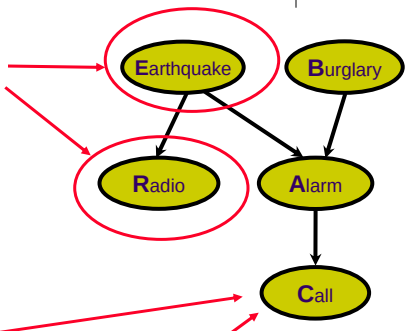
$p(\theta \mid G)$?

Global & Local Parameter Independence

- Global Parameter Independence

  For every DAG model:

$$p(\theta_m \mid G) = \prod_{i=1}^{M} p(\theta_i \mid G)$$



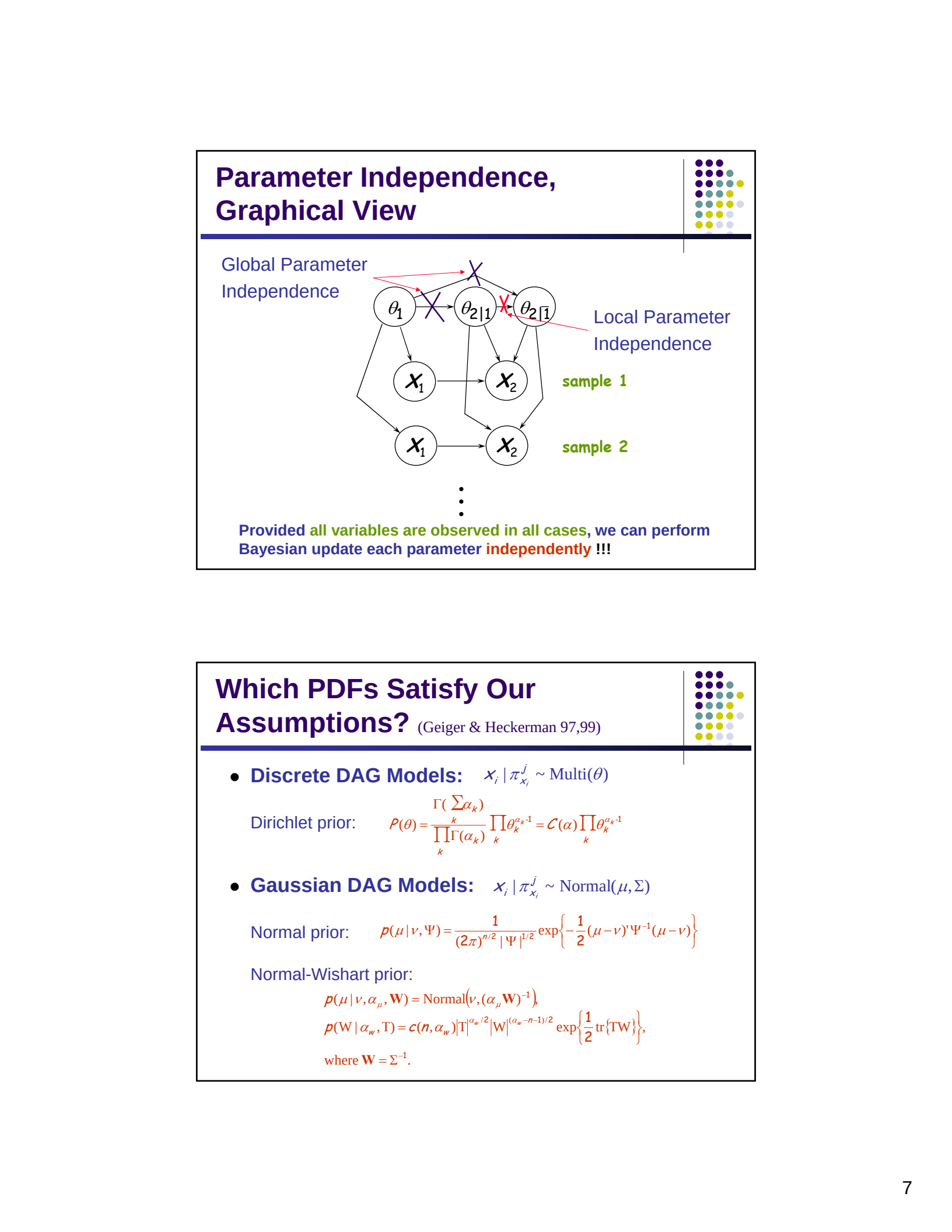
- Local Parameter Independence

  For every node:

$$p(\theta_i \mid G) = \prod_{j=1}^{q_i} p(\theta_{x_i^k \mid \mathbf{x}_{\pi_i}^j} \mid G)$$

$P(\theta_{Call \mid Alarm = YES})$ **independent of** $P(\theta_{Call \mid Alarm = NO})$

Parameter Independence, Graphical View



Global Parameter Independence

Local Parameter Independence

sample 1

sample 2

Provided all variables are observed in all cases, we can perform Bayesian update each parameter independently !!!

Which PDFs Satisfy Our Assumptions? (Geiger & Heckerman 97,99)

- **Discrete DAG Models:** $x_i \mid \pi^j_{x_i} \sim \text{Multi}(\theta)$

  Dirichlet prior: $$P(\theta) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \theta_k^{\alpha_k - 1} = C(\alpha) \prod_k \theta_k^{\alpha_k - 1}$$

- **Gaussian DAG Models:** $x_i \mid \pi^j_{x_i} \sim \text{Normal}(\mu, \Sigma)$

  Normal prior: $$p(\mu \mid \nu, \Psi) = \frac{1}{(2\pi)^{n/2} \mid \Psi \mid^{1/2}} \exp\left\{-\frac{1}{2}(\mu - \nu)' \Psi^{-1} (\mu - \nu)\right\}$$

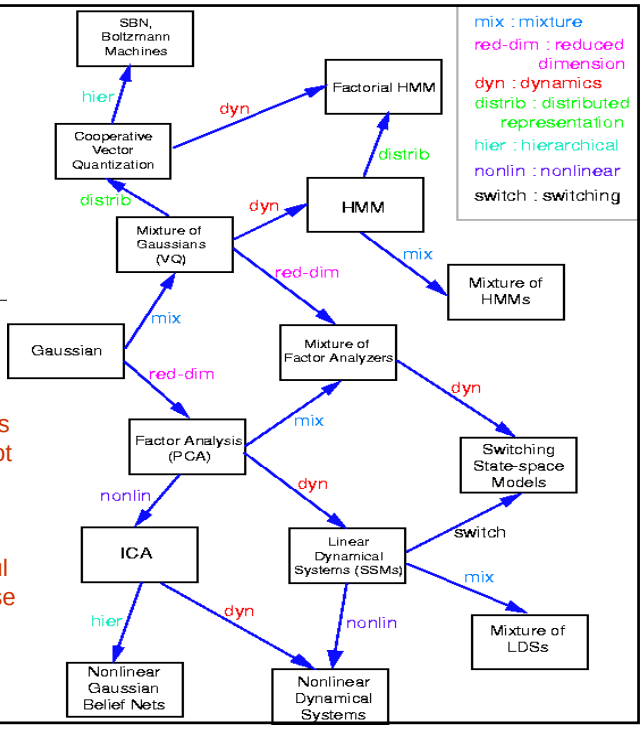
  Normal-Wishart prior:

  $$p(\mu \mid \nu, \alpha_\mu, \mathbf{W}) = \text{Normal}\left(\nu, (\alpha_\mu \mathbf{W})^{-1}\right),$$

  $$p(\mathrm{W} \mid \alpha_w, \mathrm{T}) = c(n, \alpha_w) |\mathrm{T}|^{\alpha_w/2} |\mathrm{W}|^{(\alpha_w - n - 1)/2} \exp\left\{\frac{1}{2} \mathrm{tr}\{\mathrm{TW}\}\right\},$$

  where $\mathbf{W} = \Sigma^{-1}$.

An (incomplete) genealogy of graphical models



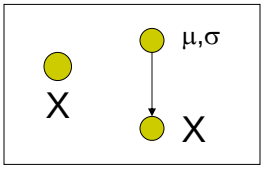
The structures of most GMs (e.g., all listed here), are not learned from data, but designed by human.

But such designs are useful and indeed favored because thereby human knowledge are put into good use …

Simplest GMs: the building blocks

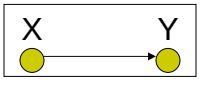
Density estimation

Parametric and nonparametric methods



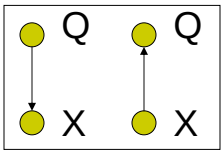
Regression

Linear, conditional mixture, nonparametric



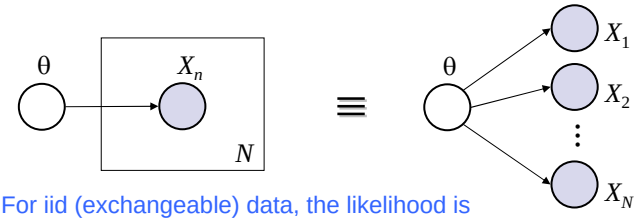
Classification

Generative and discriminative approach



Plates

- A plate is a "macro" that allows subgraphs to be replicated



  - For iid (exchangeable) data, the likelihood is

$$p(D \mid \theta) = \prod_n p(x_n \mid \theta)$$

  - We can represent this as a Bayes net with N nodes.
    - The rules of plates are simple: repeat every structure in a box a number of times given by the integer in the corner of the box (e.g. N), updating the plate index variable (e.g. n) as you go.
    - Duplicate every arrow going into the plate and every arrow leaving the plate by connecting the arrows to each copy of the structure.

Example 1: multinomial model

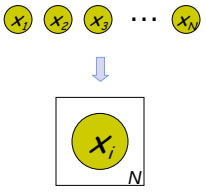
- Data:
  - We observed N ***iid*** die rolls (K-sided): D={5, 1, K, …, 3}
- Representation:

  Unit basis vectors: $x_n = \begin{pmatrix} x_n^1 \\ x_n^2 \\ \vdots \\ x_n^K \end{pmatrix}$, where $x_n^k = \{0,1\}$, and $\sum_{k=1}^{K} x_n^k = 1$

- Model:

$$x_n^k = 1 \text{ w.p. } \theta_k, \text{ and } \sum_{k \in \{1,\ldots,K\}} \theta_k = 1$$

GM:



- How to write the likelihood of a single observation x_n?

$$P(x_i) = P(\{x_n^k = 1, \text{where } k \text{ index the die-side of the } n\text{th roll}\})$$
$$= \theta_k = \theta_1^{x_n^1} \times \theta_2^{x_n^2} \times \cdots \times \theta_K^{x_n^K} = \prod_{k=1}^{K} \theta_k^{x_n^k}$$

- The likelihood of dataset D={x_1, …,x_N}:

$$P(x_1, x_2, \ldots, x_N \mid \theta) = \prod_{n=1}^{N} P(x_n \mid \theta) = \prod_{n=1}^{N} \left(\prod_k \theta_k^{x_n^k} \right) = \prod_k \theta_k^{\sum_{n=1}^{N} x_n^k} = \prod_k \theta_k^{n_k}$$

MLE: constrained optimization with Lagrange multipliers

- Objective function:

$$\ell(\theta;D) = \log P(D \mid \theta) = \log \prod_k \theta_k^{n_k} = \sum_k n_k \log \theta_k$$

- We need to maximize this subject to the constrain $\sum_{k=1}^{K} \theta_k = 1$
- Constrained cost function with a Lagrange multiplier

$$\bar{\ell} = \sum_k n_k \log \theta_k + \lambda\left(1 - \sum_{k=1}^{K} \theta_k\right)$$

- Take derivatives wrt θ_k

$$\frac{\partial \bar{\ell}}{\partial \theta_k} = \frac{n_k}{\theta_k} - \lambda = 0$$

$$n_k = \lambda \theta_k \Rightarrow \sum_k n_k = N = \lambda \sum_k \theta_k = \lambda$$

$$\Rightarrow \quad \hat{\theta}_{k,MLE} = \frac{n_k}{N} \quad \text{or} \quad \hat{\theta}_{MLE} = \frac{1}{N}\sum_n x_n$$

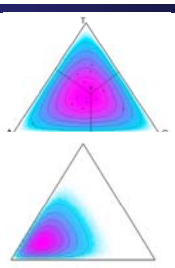
Frequency as sample mean

- Sufficient statistics
  - The counts, $\vec{n} = (n_1, \cdots, n_K), n_k = \sum_n x_n^k$, are **sufficient statistics** of data D

Bayesian estimation:

- Dirichlet distribution:

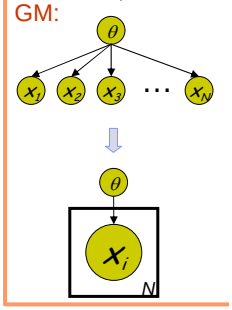
$$P(\theta) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \theta_k^{\alpha_k - 1} = C(\alpha) \prod_k \theta_k^{\alpha_k - 1}$$



- Posterior distribution of θ:

$$P(\theta \mid x_1, \ldots, x_N) = \frac{p(x_1, \ldots, x_N \mid \theta) p(\theta)}{p(x_1, \ldots, x_N)} \propto \prod_k \theta_k^{n_k} \prod_k \theta_k^{\alpha_k - 1} = \prod_k \theta_k^{\alpha_k + n_k - 1}$$

GM:



  - Notice the isomorphism of the posterior to the prior,
  - such a prior is called a **conjugate prior**

- Posterior mean estimation:

$$\theta_k = \int \theta_k p(\theta \mid D) d\theta = C \int \theta_k \prod_k \theta_k^{\alpha_k + n_k - 1} d\theta = \frac{n_k + \alpha_k}{N + |\alpha|}$$

Dirichlet parameters can be understood as pseudo-counts

More on Dirichlet Prior:

- Where is the normalize constant $\mathcal{C}(\alpha)$ come from?

$$\frac{1}{\mathcal{C}(\alpha)} = \int \cdots \int \theta_1^{\alpha_1 - 1} \cdots \theta_K^{\alpha_K - 1} d\theta_1 \cdots d\theta_K = \frac{\prod_k \Gamma(\alpha_k)}{\Gamma\left(\sum_k \alpha_k\right)}$$

  - Integration by parts
  - $\Gamma(\alpha)$ is the gamma function: $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t}\, dt$
  - For inregers, $\Gamma(n+1) = n!$
- Marginal likelihood:

$$p(\{x_1, \ldots, x_N\} \mid \vec{\alpha}) = p(\vec{n} \mid \vec{\alpha}) = \int p(\vec{n} \mid \vec{\theta}) p(\vec{\theta} \mid \vec{\alpha}) d\vec{\theta} = \frac{\mathcal{C}(\vec{\alpha})}{\mathcal{C}(\vec{n} + \vec{\alpha})}$$

- Posterior in closed-form:

$$P(\vec{\theta} \mid \{x_1, \ldots, x_N\}, \vec{\alpha}) = \frac{p(\vec{n} \mid \theta) p(\theta \mid \vec{\alpha})}{p(\vec{n} \mid \vec{\alpha})} = \mathcal{C}(\vec{n} + \vec{\alpha}) \prod_k \theta_k^{\alpha_k + n_k - 1} = \mathrm{Dir}(\vec{n} + \vec{\alpha})$$

- Posterior predictive rate:

$$p(x_{N+1} = i \mid \{x_1, \ldots, x_N\}, \vec{\alpha}) = \int \mathcal{C}(\vec{n} + \vec{\alpha}) \prod_{k \neq i} \theta_k^{\alpha_k + n_k - 1} \times \theta_i^{\alpha_k + n_k} d\vec{\theta} = \frac{\mathcal{C}(\vec{n} + \vec{\alpha})}{\mathcal{C}(\vec{n} + \vec{\alpha} + x_N)} = \frac{n_i + \alpha_i}{|\vec{n}| + |\vec{\alpha}|}$$

Sequential Bayesian updating

- Start with Dirichlet prior $P(\vec{\theta} \mid \vec{\alpha}) = \mathrm{Dir}(\vec{\theta} : \vec{\alpha})$
- Observe N' samples with sufficient statistics $\vec{n}'$. Posterior becomes:

$$P(\vec{\theta} \mid \vec{\alpha}, \vec{n}') = \mathrm{Dir}(\vec{\theta} : \vec{\alpha} + \vec{n}')$$

- Observe another N'' samples with sufficient statistics $\vec{n}''$. Posterior becomes:

$$P(\vec{\theta} \mid \vec{\alpha}, \vec{n}', \vec{n}'') = \mathrm{Dir}(\vec{\theta} : \vec{\alpha} + \vec{n}' + \vec{n}'')$$

- So sequentially absorbing data in any order is equivalent to batch update.

Effect of Prior Strength

- Let $N = |\vec{n}| = \sum_k n_k$ be the number of observed samples
- Let $A = |\vec{\alpha}| = \sum_k \alpha_k$ be the number of "pseudo observations" ---- the strength of the prior
- Let $\vec{\alpha}' = |\vec{\alpha}| / A$ denote the prior means
- Then posterior mean is a convex combination of the prior mean and the MLE:

$$p(x_{N+1} = i \mid \{x_1, \ldots, x_N\}, \vec{\alpha}) = \frac{n_i + \alpha_i}{|\vec{n}| + |\vec{\alpha}|} = \frac{n_i + \alpha_i}{N + A}$$
$$= \frac{A}{N+A}\frac{\alpha_i}{A} + \frac{N}{N+A}\frac{n_i}{N}$$
$$= \lambda \alpha_i' + (1-\lambda)\hat{\theta}_{k,MLE}$$

where $\lambda = \frac{A}{N+A}$.

Effect of Prior Strength

- Suppose we have a uniform prior ($\alpha_k' = 1/K$), and we observe $\vec{n} = (n_h = 2, n_t = 8)$
- Weak prior A = 2. Posterior prediction:

$$p(x = h \mid n_h = 2, n_t = 8, \vec{\alpha} = \vec{\alpha}' \times 2) = \frac{1+2}{2+10} = 0.25$$

- Strong prior A = 20. Posterior prediction:

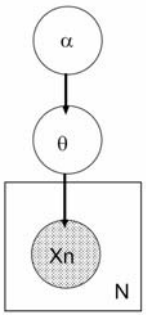
$$p(x = h \mid n_h = 2, n_t = 8, \vec{\alpha} = \vec{\alpha}' \times 20) = \frac{10+2}{20+10} = 0.40$$

- However, if we have enough data, it washes away the prior. e.g., $\vec{n} = (n_h = 200, n_t = 800)$. Then the estimates under weak and strong prior are $\frac{1+200}{2+1000}$ and $\frac{10+200}{20+1000}$, respectively, both of which are close to 0.2

Hierarchical Bayesian Models

- θ are the parameters for the likelihood $p(x|\theta)$
- α are the parameters for the prior $p(\theta|\alpha)$.
- We can have hyper-hyper-parameters, etc.
- We stop when the choice of hyper-parameters makes no difference to the marginal likelihood; typically make hyper-parameters constants.
- Where do we get the prior?
  - Intelligent guesses
  - Empirical Bayes (Type-II maximum likelihood)
    - → computing point estimates of α :

$$\hat{\vec{\alpha}}_{MLE} = \arg\max_{\vec{\alpha}} = p(\vec{n} \mid \vec{\alpha})$$



Example 2: univariate normal

- Data:
  - We observed N ***iid*** real samples:
    D={-0.1, 10, 1, -5.2, …, 3}
- Model: $P(x) = (2\pi\sigma^2)^{-1/2} \exp\{-(x-\mu)^2 / 2\sigma^2\}$
- Log likelihood:

$$\ell(\theta; D) = \log P(D \mid \theta) = -\frac{N}{2}\log(2\pi\sigma^2) - \frac{1}{2}\sum_{n=1}^{N}\frac{(x_n-\mu)^2}{\sigma^2}$$

- MLE: take derivative and set to zero:

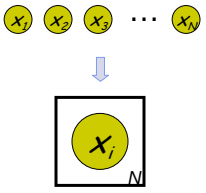
$$\frac{\partial \ell}{\partial \mu} = (1/\sigma^2)\sum_n (x_n - \mu)$$

$$\frac{\partial \ell}{\partial \sigma^2} = -\frac{N}{2\sigma^2} + \frac{1}{2\sigma^4}\sum_n (x_n-\mu)^2$$

$$\Rightarrow \quad \mu_{MLE} = \frac{1}{N}\sum_n (x_n)$$

$$\sigma^2_{MLE} = \frac{1}{N}\sum_n (x_n - \mu_{ML})^2$$

GM:



Bayesian estimation:

- Normal Prior:

$$P(\mu) = (2\pi\tau^2)^{-1/2} \exp\{-(\mu-\mu_0)^2/2\tau^2\}$$

- Joint probability:

$$P(x,\mu) = (2\pi\sigma^2)^{-N/2} \exp\left\{-\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2\right\}$$
$$\times (2\pi\tau^2)^{-1/2} \exp\{-(\mu-\mu_0)^2/2\tau^2\}$$

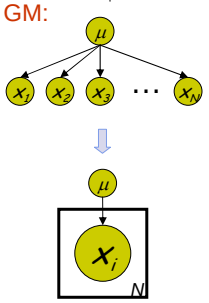
- Posterior:

$$P(\mu \mid x) = (2\pi\tilde{\sigma}^2)^{-1/2} \exp\{-(\mu-\tilde{\mu})^2/2\tilde{\sigma}^2\}$$

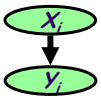
where $\tilde{\mu} = \frac{N/\sigma^2}{N/\sigma^2+1/\tau^2}\bar{x} + \frac{1/\tau^2}{N/\sigma^2+1/\tau^2}\mu_0$, and $\tilde{\sigma}^2 = \left(\frac{N}{\sigma^2}+\frac{1}{\tau^2}\right)^{-1}$

Sample mean

GM:



Two-node BNs



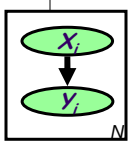
X	Y	$p(Y\|X)$
$\mathbb{R}^n$	$\mathbb{R}^m$	regression
$\mathbb{R}^n$	$\{0,1\}$	binary classification
$\{0,1\}^n$	$\{0,1\}$	binary classification
$\mathbb{R}^n$	$\{1,\ldots,K\}$	multiclass classification
$\{1,\ldots,K\}$	$\mathbb{R}^n$	conditional density modeling

Example 3: conditional Gaussian
--- a classification model

- The data:

$$\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_N, y_N)\}$$

GM:



- Both nodes are observed:
  - X is a class indicator vector

$$p(x_n) = \text{multi}(x_n : \pi) = \prod_k \pi_k^{x_n^k}$$

  - Y is a conditional Gaussian variable with a class-specific mean

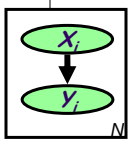
$$p(y_n \mid x_n^k = 1, \mu, \sigma) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(y_n - \mu_k)^2\right\}$$

$$p(y \mid x, \mu, \sigma) = \prod_n \left(\prod_k N(y_n : \mu_k, \sigma)^{x_n^k} \right)$$

MLE of conditional Gaussian

- Data log-likelihood

GM:



$$\ell(\theta; D) = \log \prod_n p(x_n, y_n) = \log \prod_n p(x_n \mid \pi) p(y_n \mid x_n, \mu, \sigma)$$

$$= \sum_n \log \prod_k \pi_k^{x_n^k} + \sum_n \log \prod_k N(y_n; \mu_k, \sigma)^{x_n^k}$$

$$= \sum_n \sum_k x_n^k \log \pi_k - \sum_n \sum_k x_n^k \frac{1}{2\sigma^2}(y_n - \mu_k)^2 + C$$

- MLE

$$\hat{\pi}_{k,MLE} = \arg\max \ell(\theta; D), \quad \Rightarrow \frac{\partial}{\partial \pi_k} \ell(\theta; D) = 0, \forall k, \quad \text{s.t.} \sum_k \pi_k = 1$$

$$\Rightarrow \hat{\pi}_{k,MLE} = \sum_n x_n^k / N = n_k / N$$

the fraction of samples of class m

$$\hat{\mu}_{k,MLE} = \arg\max \ell(\theta; D), \quad \Rightarrow \hat{\mu}_{k,MLE} = \frac{\sum_n x_n^k y_n}{\sum_n x_n^k} = \frac{\sum_n x_n^k y_n}{n_k}$$

the average of samples of class m

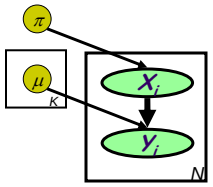
Bsyesian estimation of conditional Gaussian

- Prior:

$$P(\vec{\pi} \mid \vec{\alpha}) = \mathrm{Dir}(\vec{\pi} : \vec{\alpha})$$

$$P(\mu_k \mid \nu) = \mathrm{Normal}(\mu_k : \nu, \tau)$$

GM:



- Posterior mean (Bayesian est.)

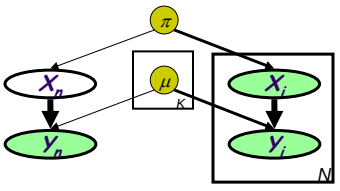
$$\pi_{k,Bayes} = \frac{N}{N+|\alpha|}\hat{\pi}_{k,ML} + \frac{|\alpha|}{N+|\alpha|}\frac{\alpha_k}{|\alpha|} = \frac{n_k+\alpha_k}{N+|\alpha|}$$

$$\mu_{k,Bayes} = \frac{n_k/\sigma^2}{n_k/\sigma^2+1/\tau^2}\hat{\mu}_{k,ML} + \frac{1/\tau^2}{n_k/\sigma^2+1/\tau^2}\nu, \quad \text{and} \quad \sigma^2_{Bayes} = \left(\frac{N}{\sigma^2}+\frac{1}{\tau^2}\right)^{-1}$$

From conditional density modeling to classification

- Given y_n, what is its corresponding x_n?
  - This is a generative classifier
- Frequentist prediction:

GM:



$$p(x_n^k = 1 \mid y_n, \mu, \sigma) = \frac{p(x_n^k = 1, y_n \mid \mu, \sigma, \pi)}{p(y_n \mid \mu, \sigma)} = \frac{\pi_k N(y_n, \mid \mu_k, \sigma)}{\sum_j \pi_j N(y_n, \mid \mu_j, \sigma)}$$

- Bayesian:
  - Do you want to make it a homework?

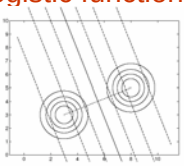
From conditional density modeling to classification, cont.

- Understanding the predictive distribution

$$p(x_n^k = 1 \mid y_n, \mu, \sigma) = \frac{p(x_n^k = 1, y_n \mid \mu, \sigma, \pi)}{p(y_n \mid \mu, \sigma)} = \frac{\pi_k N(y_n, \mid \mu_k, \sigma)}{\sum_j \pi_j N(y_n, \mid \mu_j, \sigma)} \quad *$$

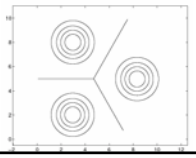
- For two class (i.e., K=2), * turns out to be the logistic function

$$p(x_n^k = 1 \mid y_n) = \frac{1}{1 + e^{-\theta^T y_n}}$$



- For multiple class (i.e., K>2), * correspond to a softmax function

$$p(x_n^k = 1 \mid y_n) = \frac{e^{-\theta_k^T y_n}}{\sum_j e^{-\theta_j^T y_n}}$$

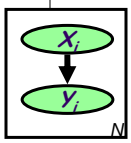


Example 4: linear regression

- The data:

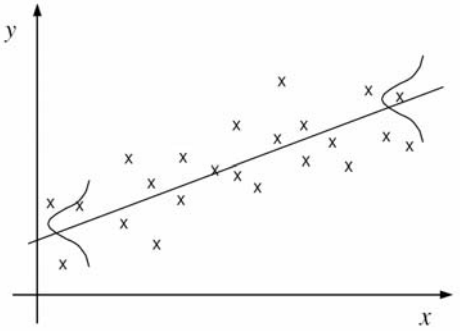
$$\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_N, y_N)\}$$

GM:



- Both nodes are observed:
  - X is an input vector
  - y is a response vector

- A regression problem
  Models $p(y|x)$ directly,
  rather than $p(x,y)$



Multivariate Linear Regression

- Consider vector-valued input $X \in \mathbb{R}^k$ leading to vector-valued output $Y \in \mathbb{R}^d$ via regression matrix $A \in \mathbb{R}^{k \times d}$:

$$p(y \mid x) = \frac{1}{(2\pi)^{-d/2}|\Sigma|^{-1/2}} \exp\left\{-\frac{1}{2}(y - Ax)^T \Sigma^{-1}(y - Ax)\right\}$$

- Log-(conditional-) likelihood

$$\ell = -\frac{1}{2}\sum_n |\Sigma| - \frac{1}{2}\sum_n (y_n - Ax_n)^T \Sigma^{-1}(y_n - Ax_n) + c$$

- To take derivatives wrt a matrix, we use the following identity

$$\frac{\partial\left((\mathrm{M}a + b)^T \mathrm{C}(\mathrm{M}a + b)\right)}{\partial \mathrm{M}} = (\mathrm{C} + \mathrm{C}^T)(\mathrm{M}a + b)a^T$$

where $\mathrm{M} = A, a = -x_n, b = y_n$ and $\mathrm{C} = \Sigma^{-1}$

Multivariate Linear Regression

- Log-(conditional-) likelihood

$$\ell = -\frac{1}{2}\sum_n |\Sigma| - \frac{1}{2}\sum_n (y_n - Ax_n)^T \Sigma^{-1}(y_n - Ax_n) + c$$

- Using $\frac{\partial\left((\mathrm{M}a + b)^T \mathrm{C}(\mathrm{M}a + b)\right)}{\partial \mathrm{M}} = (\mathrm{C} + \mathrm{C}^T)(\mathrm{M}a + b)a^T$

we have $\frac{\partial \ell}{\partial A} = -\frac{1}{2}\sum_n 2\Sigma^{-1}(y_n - Ax_n)x_n^T$

$$= \Sigma^{-1}\left(\sum_n y_n x_n^T - A\sum_n x_n x_n^T\right) \stackrel{\text{def}}{=} \Sigma^{-1}(S_{yx'} - AS_{xx'}) = 0$$

where $S_{yx'}$ and $S_{xx'}$ are the sufficient statistics.

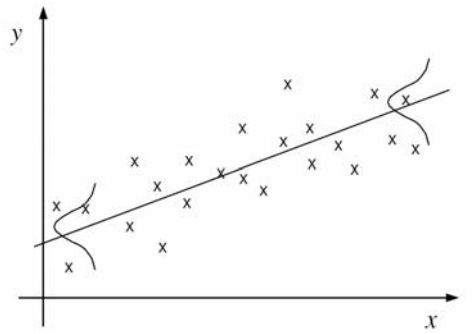
Hence $A = S_{yx'}S_{xx'}^{-1}$

1-D linear regression

$A = S_{YX} S_{XX}^{-1}$

- In the special case of scalar outputs, let $A = \theta^T$, and the design matrix $X = [x_1, \ldots, x_N]$ as a row vector and $Y = [y_1, \ldots, y_N]'$ as a column vector. Then we get the normal equations

$$\theta = (X^T X)^{-1} X^T Y$$



Bayesian linear regression