Probabilistic Graphical Models

10-708

More on learning fully observed BNs, exponential families, and generalized linear models

Eric Xing

Lecture 10, Oct 12, 2005
Reading: MJ-Chap. 7,8

Exponential family

- For a numeric random variable X

$$p(x \mid \eta) = h(x)\exp\{\eta^T T(x) - A(\eta)\}$$
$$= \frac{1}{Z(\eta)} h(x)\exp\{\eta^T T(x)\}$$

is an exponential family distribution with natural (canonical) parameter η

- Function $T(x)$ is a *sufficient statistic*.
- Function $A(\eta) = \log Z(\eta)$ is the log normalizer.
- Examples: Bernoulli, multinomial, Gaussian, Poisson, gamma,...
- A distribution $p(x)$ has finite sufficient statistics (independent of number of data cases) iff it is in the exponential family.

Multivariate Gaussian Distribution

- For a continuous vector random variable $x \in \mathbb{R}^k$:

$$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$

Moment parameter

$$= \frac{1}{(2\pi)^{k/2}} \exp\left\{-\tfrac{1}{2}\mathrm{tr}\left(\Sigma^{-1}xx^T\right) + \mu^T\Sigma^{-1}x - \tfrac{1}{2}\mu^T\Sigma^{-1}\mu - \log|\Sigma|\right\}$$

- Exponential family representation

Natural parameter

$$\eta = \left[\Sigma^{-1}\mu; -\tfrac{1}{2}\mathrm{vec}\left(\Sigma^{-1}\right)\right] = \left[\eta_1, \mathrm{vec}(\eta_2)\right], \quad \eta_1 = \Sigma^{-1}\mu \text{ and } \eta_2 = -\tfrac{1}{2}\Sigma^{-1}$$

$$T(x) = \left[x; \mathrm{vec}\left(xx^T\right)\right]$$

$$A(\eta) = \tfrac{1}{2}\mu^T\Sigma^{-1}\mu + \log|\Sigma| = -\tfrac{1}{2}\mathrm{tr}(\eta_2\eta_1\eta_1^T) - \tfrac{1}{2}\log(-2\eta_2)$$

$$h(x) = (2\pi)^{-k/2}$$

  - Note: a k-dimensional Gaussian is a ($d+d^2$)-parameter distribution with a ($d+d^2$)-element vector of sufficient statistics (but because of symmetry and positivity, parameters are constrained and have lower degree of freedom)

Multinomial distribution

- For a binary vector random variable $x \sim \mathrm{multi}(x \mid \pi)$,

$$p(x|\pi) = \pi_1^{x^1}\pi_2^{x^2}\cdots\pi_K^{x^K} = \exp\left\{\sum_k x^k \ln\pi_k\right\}$$

$$= \exp\left\{\sum_{k=1}^{K-1} x^k \ln\pi_k + \left(1 - \sum_{k=1}^{K-1} x^k\right)\ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right)\right\}$$

$$= \exp\left\{\sum_{k=1}^{K-1} x^k \ln\left(\frac{\pi_k}{1-\sum_{k=1}^{K-1}\pi_k}\right) + \ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right)\right\}$$

- Exponential family representation

$$\eta = \left[\ln\left(\pi_k / \pi_K\right); 0\right]$$

$$T(x) = [x]$$

$$A(\eta) = -\ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right) = \ln\left(\sum_{k=1}^{K} e^{\eta_k}\right)$$

$$h(x) = 1$$

Why exponential family?

- Moment generating property

$$\frac{dA}{d\eta} = \frac{d}{d\eta}\log Z(\eta) = \frac{1}{Z(\eta)}\frac{d}{d\eta}Z(\eta)$$
$$= \frac{1}{Z(\eta)}\frac{d}{d\eta}\int h(x)\exp\{\eta^T T(x)\}dx$$
$$= \int T(x)\frac{h(x)\exp\{\eta^T T(x)\}}{Z(\eta)}dx$$
$$= E[T(x)]$$

$$\frac{d^2A}{d\eta^2} = \int T^2(x)\frac{h(x)\exp\{\eta^T T(x)\}}{Z(\eta)}dx - \int T(x)\frac{h(x)\exp\{\eta^T T(x)\}}{Z(\eta)}dx\,\frac{1}{Z(\eta)}\frac{d}{d\eta}Z(\eta)$$
$$= E[T^2(x)] - E^2[T(x)]$$
$$= Var[T(x)]$$

Moment estimation

- We can easily compute moments of any exponential family distribution by taking the derivatives of the log normalizer $A(\eta)$.
- The q^{th} derivative gives the q^{th} centered moment.

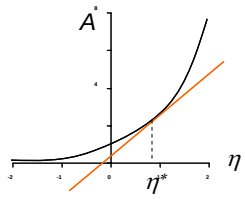
$$\frac{dA(\eta)}{d\eta} = \text{mean}$$
$$\frac{d^2A(\eta)}{d\eta^2} = \text{variance}$$
$$\ldots$$

- When the sufficient statistic is a stacked vector, partial derivatives need to be considered.

Moment vs canonical parameters

- The moment parameter μ can be derived from the natural (canonical) parameter

$$\frac{dA(\eta)}{d\eta} = E[T(x)] \overset{\text{def}}{=} \mu$$



- $A(\eta)$ is convex since

$$\frac{d^2A(\eta)}{d\eta^2} = Var[T(x)] > 0$$

- Hence we can invert the relationship and infer the canonical parameter from the moment parameter (1-to-1):

$$\eta \overset{\text{def}}{=} \psi(\mu)$$

  - A distribution in the exponential family can be parameterized not only by η – the canonical parameterization, but also by μ – the moment parameterization.

MLE for Exponential Family

- For *iid* data, the log-likelihood is

$$\ell(\eta; D) = \log \prod_n h(x_n) \exp\{\eta^T T(x_n) - A(\eta)\}$$

$$= \sum_n \log h(x_n) + \left(\eta^T \sum_n T(x_n)\right) - NA(\eta)$$

- Take derivatives and set to zero:

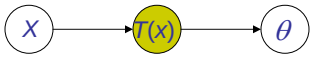
$$\frac{\partial \ell}{\partial \eta} = \sum_n T(x_n) - N\frac{\partial A(\eta)}{\partial \eta} = 0$$

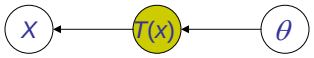
$$\Rightarrow \quad \frac{\partial A(\eta)}{\partial \eta} = \frac{1}{N}\sum_n T(x_n)$$

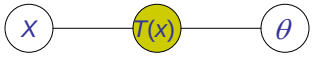
$$\hat{\mu}_{MLE} = \frac{1}{N}\sum_n T(x_n)$$

- This amounts to **moment matching**.
- We can infer the canonical parameters using $\hat{\eta}_{MLE} = \psi(\hat{\mu}_{MLE})$

Sufficiency

- For $p(x|\theta)$, $T(x)$ is *sufficient* for θ if there is no information in X regarding θ yeyond that in $T(x)$.
  - We can throw away X for the purpose pf inference w.r.t. θ.
  - Bayesian view 

$$p(\theta \mid T(x), x) = p(\theta \mid T(x))$$
  - Frequentist view 

$$p(x \mid T(x), \theta) = p(x \mid T(x))$$
  - The Neyman factorization theorem 

    - $T(x)$ is *sufficient* for θ if

$$p(x, T(x), \theta) = \psi_1(T(x), \theta)\psi_2(x, T(x))$$
$$\Rightarrow p(x \mid \theta) = g(T(x), \theta)h(x, T(x))$$

Examples

- Gaussian:

$$\eta = \left[\Sigma^{-1}\mu; -\tfrac{1}{2}\mathrm{vec}\left(\Sigma^{-1}\right)\right]$$
$$T(x) = \left[x; \mathrm{vec}\left(xx^T\right)\right]$$
$$A(\eta) = \tfrac{1}{2}\mu^T\Sigma^{-1}\mu + \tfrac{1}{2}\log|\Sigma|$$
$$h(x) = (2\pi)^{-K/2}$$
$$\Rightarrow \mu_{MLE} = \frac{1}{N}\sum_n T_1(x_n) = \frac{1}{N}\sum_n x_n$$

- Multinomial:

$$\eta = \left[\ln\left(\pi_k / \pi_K\right); 0\right]$$
$$T(x) = [x]$$
$$A(\eta) = -\ln\left(1 - \sum_{k=1}^{K-1}\pi_k\right) = \ln\left(\sum_{k=1}^{K} e^{\eta_k}\right)$$
$$h(x) = 1$$
$$\Rightarrow \mu_{MLE} = \frac{1}{N}\sum_n x_n$$

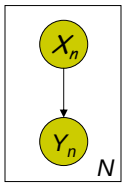
- Poisson:

$$\eta = \log\lambda$$
$$T(x) = x$$
$$A(\eta) = \lambda = e^{\eta}$$
$$h(x) = \frac{1}{x!}$$
$$\Rightarrow \mu_{MLE} = \frac{1}{N}\sum_n x_n$$

Generalized Linear Models (GLIMs)

- The graphical model
  - Linear regression
  - Discriminative linear classification
  - Commonality:

    model $\mathrm{E}(y)=\mu=f(\theta^T x)$
    - What is $p()$, the cond. dist. Of Y?
    - What is $f()$, the response function?



- GLIM
  - The observed input x is assumed to enter into the model via a linear combination of its elements $\xi = \theta^T x$
  - The conditional mean μ is represented as a function $f(\xi)$ of ξ, where f is known as the response function
  - The observed output y is assumed to be characterized by an exponential family distribution with conditional mean μ.

GLIM, cont.

$$\theta \rightarrow \xi \xleftarrow{} x \qquad \xi \xrightarrow{f} \mu \xrightarrow{\psi} \eta$$

$$p(y \mid \eta) = h(x)\exp\left\{\eta^T y - A(\eta)\right\}$$

$$\Rightarrow p(y \mid \eta) = h(x)\exp\left\{\tfrac{1}{\phi}\left(\eta^T y - A(\eta)\right)\right\}$$

- The choice of exp family is constrained by the nature of the data y
  - Example: y is a continuous vector → multivariate Gaussian
    y is a class label → Bernoulli or multinomial
- The choice of the response function
  - Following some mild constrains, e.g., [0,1]. Positivity …
  - Canonical response function: $f = \psi^{-1}(\cdot)$
    - In this case $\theta^T x$ directly corresponds to canonical parameter η.

MLE for GLIMs with natural response

- Log-likelihood

$$\ell = \sum_n \log h(y_n) + \sum_n \left(\theta^T x_n y_n - A(\eta_n)\right)$$

- Derivative of Log-likelihood

$$\begin{aligned}\frac{d\ell}{d\theta} &= \sum_n \left(x_n y_n - \frac{dA(\eta_n)}{d\eta_n}\frac{d\eta_n}{d\theta}\right) \\ &= \sum_n (y_n - \mu_n)x_n \\ &= X^T(y-\mu)\end{aligned}$$

This is a fixed point function because μ is a function of θ

- Online learning for canonical GLIMs
  - Stochastic gradient ascent = least mean squares (LMS) algorithm:

$$\theta^{t+1} = \theta^t + \rho\left(y_n - \mu_n^t\right)x_n$$

where $\mu_n^t = \left(\theta^t\right)^T x_n$ and ρ is a step size

Batch learning for canonical GLIMs

- The Hessian matrix

$$\begin{aligned}H &= \frac{d^2\ell}{d\theta d\theta^T} = \frac{d}{d\theta^T}\sum_n (y_n - \mu_n)x_n = \sum_n x_n \frac{d\mu_n}{d\theta^T} \\ &= -\sum_n x_n \frac{d\mu_n}{d\eta_n}\frac{d\eta_n}{d\theta^T} \\ &= -\sum_n x_n \frac{d\mu_n}{d\eta_n} x_n^T \quad \text{since } \eta_n = \theta^T x_n \\ &= -X^T W X\end{aligned}$$

where $X = \left[x_n^T\right]$ is the design matrix and

$$W = \text{diag}\left(\frac{d\mu_1}{d\eta_1}, \ldots, \frac{d\mu_N}{d\eta_N}\right)$$

which can be computed by calculating the 2nd derivative of $A(\eta_n)$

Iteratively Reweighted Least Squares (IRLS)

- Recall Newton-Raphson methods with cost function $\mathcal{J}$

$$\theta^{t+1} = \theta^t - H^{-1}\nabla_\theta \mathcal{J}$$

- We now have

$$\nabla_\theta \mathcal{J} = X^T(y-\mu)$$
$$H = -X^T W X$$

- Now:

$$\theta^{t+1} = \theta^t + H^{-1}\nabla_\theta \ell$$
$$= \left(X^T W^t X\right)^{-1}\left[X^T W^t X\theta^t + X^T(y-\mu^t)\right]$$
$$= \left(X^T W^t X\right)^{-1} X^T W^t z^t$$

where the adjusted response is $z^t = X\theta^t + \left(W^t\right)^{-1}(y-\mu^t)$

- This can be understood as solving the following " Iteratively reweighted least squares " problem

$$\theta^{t+1} = \arg\min_\theta (z - X\theta)^T W (z - X\theta)$$

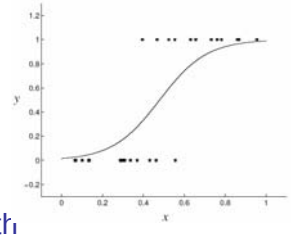
Example 1: logistic regression (sigmoid classifier)

- The condition distribution: a Bernoulli

$$p(y \mid x) = \mu(x)^y (1-\mu(x))^{1-y}$$

where μ is a logistic function

$$\mu(x) = \frac{1}{1+e^{-\eta(x)}}$$



- $p(y|x)$ is an exponential family function, with
  - mean: $E[y \mid x] = \mu = \frac{1}{1+e^{-\eta(x)}}$
  - and canonical response function $\eta = \xi = \theta^T x$

- IRLS

$$\frac{d\mu}{d\eta} = \mu(1-\mu)$$

$$W = \begin{pmatrix} \mu_1(1-\mu_1) & & \\ & \ddots & \\ & & \mu_N(1-\mu_N) \end{pmatrix}$$

Logistic regression: practical issues

- It is very common to use *regularized* maximum likelihood.

$$p(y=\pm 1|x,\theta)=\frac{1}{1+e^{-y\theta^T x}}=\sigma(y\theta^T x)$$

$$p(\theta)\sim \text{Normal}(0,\lambda^{-1}I)$$

$$l(\theta)=\sum_n \log\left(\sigma(y_n\theta^T x_n)\right)-\frac{\lambda}{2}\theta^T\theta$$

  - IRLS takes $O(Nd^3)$ per iteration, where N = number of training cases and d = dimension of input x.
  - Quasi-Newton methods, that approximate the Hessian, work faster.
  - Conjugate gradient takes $O(Nd)$ per iteration, and usually works best in practice.
  - Stochastic gradient descent can also be used if N is large c.f. perceptron rule:

$$\nabla_\theta \ell=\left(1-\sigma(y_n\theta^T x_n)\right)y_n x_n-\lambda\theta$$

Example 2: linear regression

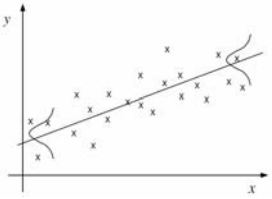
- The condition distribution: a Gaussian

$$p(y|x,\theta,\Sigma)=\frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}}\exp\left\{-\frac{1}{2}(y-\mu(x))^T\Sigma^{-1}(y-\mu(x))\right\}$$

Rescall $\Rightarrow h(x)\exp\left\{-\frac{1}{2}\Sigma^{-1}\left(\eta^T(x)y-A(\eta)\right)\right\}$

where μ is a linear function

$$\mu(x)=\theta^T x=\eta(x)$$



- $p(y|x)$ is an exponential family function, with
  - mean: $E[y\mid x]=\mu=\theta^T x$
  - and canonical response function $\eta_1=\xi=\theta^T x$
- IRLS $\frac{d\mu}{d\eta}=1$, $W=I$

$$\Rightarrow\quad \theta^{t+1}=\left(X^TW^tX\right)^{-1}X^TW^tz^t$$
$$=\left(X^TX\right)^{-1}X^T\left(X\theta^t+(y-\mu^t)\right)$$
$$=\theta^t+\left(X^TX\right)^{-1}X^T(y-\mu^t)$$

Steepest descent

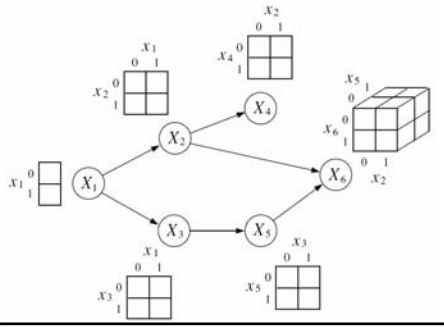
$$\overset{t\to\infty}{\Rightarrow}\quad \theta=(X^TX)^{-1}X^Ty$$

Normal equation

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:

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

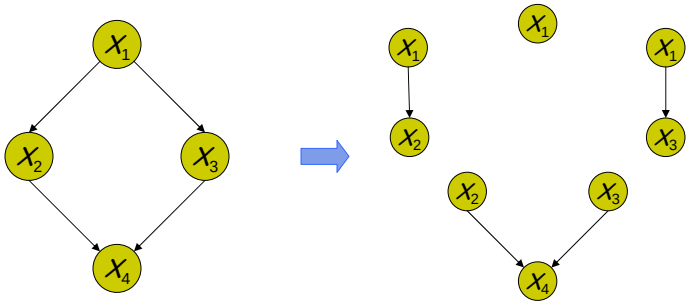


Example: A directed model

- Consider the distribution defined by the directed acyclic GM:

$$p(x \mid \theta) = p(x_1 \mid \theta_1) p(x_2 \mid x_1, \theta_1) p(x_3 \mid x_1, \theta_3) p(x_4 \mid x_2, x_3, \theta_1)$$

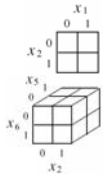
- This is exactly like learning four separate small BNs, each of which consists of a node and its parents.



MLE for BNs with tabular CPDs

- Assume each CPD is represented as a table (multinomial) where

$$\theta_{ijk} \stackrel{\text{def}}{=} p(X_i = j \mid X_{\pi_i} = k)$$



  - Note that in case of multiple parents, $\mathbf{X}_{\pi_i}$ will have a composite state, and the CPD will be a high-dimensional table
  - The sufficient statistics are counts of family configurations

$$n_{ijk} \stackrel{\text{def}}{=} \sum_n x_{n,i}^j x_{n,\pi_i}^k$$

- The log-likelihood is

$$\ell(\theta; D) = \log \prod_{i,j,k} \theta_{ijk}^{n_{ijk}} = \sum_{i,j,k} n_{ijk} \log \theta_{ijk}$$

- Using a Lagrange multiplier to enforce so $\sum_j \theta_{ijk} = 1$ we get

$$\theta_{ijk}^{ML} = \frac{n_{ijk}}{\sum_{i,j',k} n_{ij'k}}$$

MLE and Kulback-Leibler divergence

- KL divergence

$$D(q(x) \,\|\, p(x)) = \sum_x q(x) \log \frac{q(x)}{p(x)}$$

- Empirical distribution

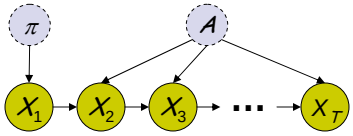
$$\tilde{p}(x) \stackrel{\text{def}}{=} \frac{1}{N} \sum_{n=1}^{N} \delta(x, x_n)$$

  - Where $\delta(x,x_n)$ is a Kronecker delta function

- $\text{Max}_\theta(\text{MLE}) \equiv \text{Min}_\theta(\text{KL})$

$$\begin{aligned} D(\tilde{p}(x) \,\|\, p(x \mid \theta)) &= \sum_x \tilde{p}(x) \log \frac{\tilde{p}(x)}{p(x \mid \theta)} \\ &= \sum_x \tilde{p}(x) \log \tilde{p}(x) - \sum_x \tilde{p}(x) \log p(x \mid \theta) \\ &= \sum_x \tilde{p}(x) \log \tilde{p}(x) - \frac{1}{N} \sum_n \log p(x_n \mid \theta) \\ &= C + \frac{1}{N} \ell(\theta; D) \end{aligned}$$

Parameter sharing



- Consider a time-invariant (stationary) 1st-order Markov model
  - Initial state probability vector: $\pi_k \stackrel{\text{def}}{=} p(X_1^k = 1)$
  - State transition probability matrix: $A_{ij} \stackrel{\text{def}}{=} p(X_t^j = 1 \mid X_{t-1}^i = 1)$
- The joint: $p(X_{1:T} \mid \theta) = p(x_1 \mid \pi) \prod_{t=2}^{T} \prod_{t=2} p(X_t \mid X_{t-1})$
- The log-likelihood: $\ell(\theta; D) = \sum_n \log p(x_{n,1} \mid \pi) + \sum_n \sum_{t=2}^{T} \log p(x_{n,t} \mid x_{n,t-1}, A)$
- Again, we optimize each parameter separately
  - π is a multinomial frequency vector, and we've seen it before
  - What about A?

Learning a Markov chain transition matrix

- A is a stochastic matrix: $\sum_j A_{ij} = 1$
- Each row of A is multinomial distribution.
- So **MLE** of A_{ij} is the fraction of transitions from i to j

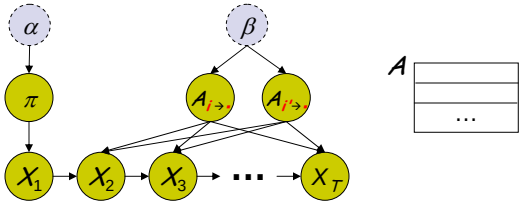
$$A_{ij}^{ML} = \frac{\#(i \to j)}{\#(i \to \bullet)} = \frac{\sum_n \sum_{t=2}^{T} x_{n,t-1}^i x_{n,t}^j}{\sum_n \sum_{t=2}^{T} x_{n,t-1}^i}$$

- Application:
  - if the states X_t represent words, this is called a *bigram language model*
- Sparse data problem:
  - If $i \to j$ did not occur in data, we will have A_{ij}=0, then any futher sequence with word pair $i \to j$ will have zero probability.
  - A standard hack: *backoff smoothing* or *deleted interpolation*

$$\tilde{A}_{i \to \bullet} = \lambda \eta_t + (1 - \lambda) A_{i \to \bullet}^{ML}$$

Bayesian language model

- Global and local parameter independence



- The posterior of $A_{i\rightarrow\bullet}$ and $A_{i'\rightarrow\bullet}$ is factorized despite v-structure on X_t, because X_{t-1} acts like a multiplexer
- Assign a Dirichlet prior β_i to each row of the transition matrix:

$$A_{ij}^{Bayes} \stackrel{\text{def}}{=} p(j \mid i, D, \beta_i) = \frac{\#(i \rightarrow j) + \beta_{i,k}}{\#(i \rightarrow \bullet) + |\beta_i|} = \lambda_i \beta'_{i,k} + (1 - \lambda_i) A_{ij}^{ML}, \text{ where } \lambda_i = \frac{|\beta_i|}{|\beta_i| + \#(i \rightarrow \bullet)}$$

  - We could consider more realistic priors, e.g., mixtures of Dirichlets to account for types of words (adjectives, verbs, etc.)