Inference and Representation

Rachel Hodos

New York University

Lab 13, December 16, 2015

Outline

1. Review for final exam
   - Representation
   - Learning
     - Parameter learning
     - Structure learning
   - Inference

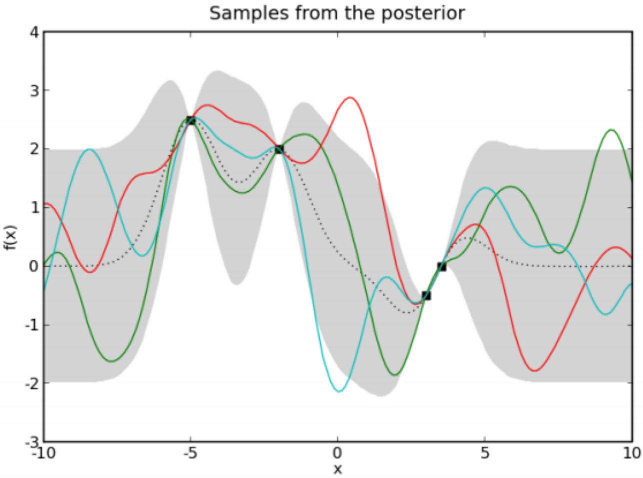
Outline

1. Review for final exam
   - Representation
   - Learning
     - Parameter learning
     - Structure learning
   - Inference

Representation

- Distributions over finite sets of random variables:
  - BN
  - MRF
  - CRF
- Distribution over functions / infinite # of variables:
  - GP

Gaussian Processes



Gaussian Processes

- For a given kernel (and kernel parameters), defines a prior over functions via multivariate Gaussian for any given x's
- In both noisy and noise-free settings, have closed-form expressions for posterior (also a GP)
- Posterior at each point is Gaussian (since marginal of multinormal is univariate normal), so e.g. can plot 95% confidence interval
- Equivalent to Bayesian linear regression on $\phi(\mathbf{x})$, where ϕ is the feature mapping consistent with the chosen kernel
- Usually a small number of parameters to learn, so can estimate via grid search
- Standard setting is regression, but latent GPs extend, e.g. to classification

Outline

1. Review for final exam
   - Representation
   - Learning
     - Parameter learning
     - Structure learning
   - Inference

Overview of Parameter Learning

- Goal: select the 'best' model parameters by minimizing some loss function with respect to the data
- Most of semester focused on MLE: loss $= -\log p(\mathbf{x}; \theta)$
- Can do MAP estimation of parameters using a prior (think of this as regularized MLE)
- We also briefly touched on pseudo-likelihood (see end of lecture 10 for more details)

MLE in fully observed setting

- Discrete BNs: given directly from empirical CPDs (see second half of Lab3 slides for proof)
- Trickier for MRFs due to partition function:
  - However, writing $p(\mathbf{x})$ in log-linear form (see lecture 10 and loglin.pdf), gives a convex objective
  - Hence can use any convex optimization algorithm
  - But computing gradient of Z is equivalent to marginal inference $\implies$ often hard
  - To get around this, can do pseudo-likelihood estimation
- In either case, MLE estimation within exponential family implies **moment-matched** solution

MLE with hidden variables: EM

a special case of

a special case of

Approach	EM	Variational EM	Variational EM with recognition models
Idea	Optimize likelihood via expectation over p(z|x)	Optimize *lower bound* on log(p) via expectation over q(z) ≈ p(z|x)	Learn direct map f: x -> params of q
Guarantees	Guaranteed to converge to local optimum	Can bound error if combined with upper bound to likelihood*	Can bound error if combined with upper bound to likelihood*

*See end of lecture 11

MLE with hidden variables: EM

- Regular EM:
  $\theta_{t+1} = \arg\max_\theta \sum_{m=1}^{M} E_{p(\mathbf{z}_m|\mathbf{x}_m;\theta_t)}[\log p(\mathbf{x}_m, \mathbf{z}_m; \theta)]$
- Variational EM:

$$\theta_{t+1} = \arg\max_\theta \sum_{m=1}^{M} E_{q(\mathbf{z}_m;\phi_t)}[\log p(\mathbf{x}_m, \mathbf{z}_m; \theta)] + H(q(\mathbf{z}; \phi_t))$$

$$\phi_{t+1}^m = \arg\max_\phi E_{q(\mathbf{z}_m;\phi)}[\log p(\mathbf{x}_m, \mathbf{z}_m; \theta_{t+1})] + H(q(\mathbf{z}_m; \phi)) \quad \forall m$$

- Variational EM with recognition model: instead of solving an optimization problem to find each ϕ_m, learn a deterministic mapping $f : \mathbf{x} \to \phi$. Now the variational parameters become the parameters of f.

Structure learning

- Chow-Liu: algorithm to learn tree-structured MRF
  - Closed-form MLE for edges + minimum spanning tree
- Sparsity structure of Gaussian MRF can be estimated via 0's in inverse of data covariance matrix
- BN structure learning can be formulated as an ILP (optional reading)

Outline

1. Review for final exam
   - Representation
   - Learning
     - Parameter learning
     - Structure learning
   - Inference

Overview of marginal inference

- Goal: for some subset $\mathbf{Z}$ of unobserved vars, and possibly a subset $\mathbf{X}$ of observed vars, compute marginals:

$$p(\mathbf{Z}|\mathbf{X} = \mathbf{x})$$

- Sum-product variable elimination: exact
- Sum-product BP: exact for trees, otherwise no guarantees
- Monte Carlo methods: approximate, but exact with infinite sampling
- Variational inference (minimize $D(q||p)$ over some set Q): approximate

Overview of MAP inference

- Goal: find the mode of the prior/posterior given some fixed θ:

$$MAP(\theta) = \arg\max_{\mathbf{x}} p(\mathbf{x}; \theta)$$

- Max-product variable elimination: exact
- Max-product BP: exact for trees, otherwise use MPLP from Lecture 14
- ILP (see next slides)

Review of yesterday's lecture

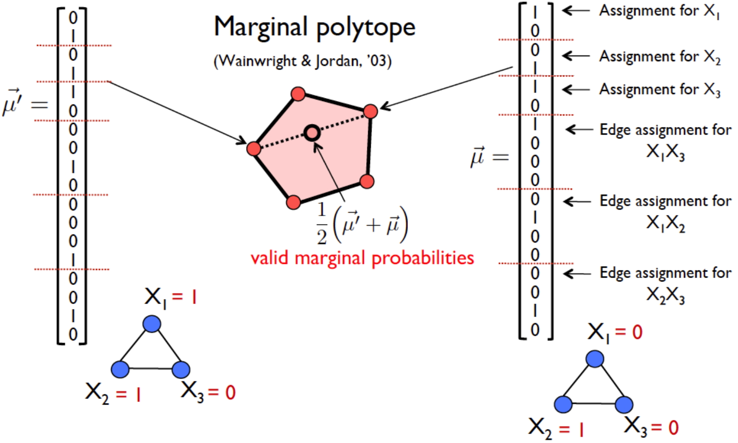
- Def'n of ILP: linear objective with linear constraints and integrality constraints
- Off the shelf ILP solvers
- Formulation of MAP inference as ILP:

$$\mathrm{MAP}(\theta) = \max_{\mu} \sum_{i \in V} \sum_{x_i} \theta_i(x_i)\mu_i(x_i) + \sum_{ij \in E} \sum_{x_i, x_j} \theta_{ij}(x_i, x_j)\mu_{ij}(x_i, x_j)$$

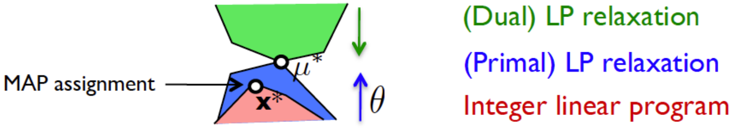
subject to:

$$\begin{aligned}
\mu_i(x_i) &\in \{0,1\} \quad \forall i \in V, x_i \\
\sum_{x_i} \mu_i(x_i) &= 1 \quad \forall i \in V \\
\mu_i(x_i) &= \sum_{x_j} \mu_{ij}(x_i, x_j) \quad \forall ij \in E, x_i \\
\mu_j(x_j) &= \sum_{x_i} \mu_{ij}(x_i, x_j) \quad \forall ij \in E, x_j
\end{aligned}$$

Interpretation of marginal polytope



Linear programming duality



$$\mathrm{MAP}(\theta) \leq \mathrm{LP}(\theta) = \mathrm{DUAL\text{-}LP}(\theta) \leq L(\delta)$$