BAYESIAN NETWORKS (DGM)

Conditional Independence: $X \perp Y|Z : P(X,Y|Z) = P(X|Z)P(Y|Z), P(X|Y,Z) = P(X|Z)$

Product rule: $P(A,B|C) = \frac{P(A,B,C)}{P(C)} = \frac{P(A|B,C)P(B,C)}{P(C)} = P(A|B,C)P(B|C)$

Markov assumption: Any random variable is locally dependent on its parent nodes. $X \perp (X_{nonDesc} \setminus X_{parent}) \mid X_{parent}$ (useful to find all CIs in DGM), so $P(x_1,x_2,...,x_N = \prod_{i=1}^n P(x_i|x_{\pi_i})$.

Graph Separation: From the three canonical 3-node graphs, a path is said to be *blocked* / d-separated (conditional independence) if it contains a node satisfying either:

(a) the arrows meet 'head-to-tail' or 'tail-to-tail' at that node, and the node is in conditioning set C;

(b) the arrows meet head-to-head at that node, and neither the node nor any of its descendants is in C.

Bayes Ball Algorithm: This is a reachability-based procedure: (1) shade the nodes in the conditioning set C; (2) place a ball at each node in the query set A; (3) let the balls traverse the graph according to the d-separation rules (separation means no passing). If none of the balls reach any node in B, then $A \perp B \mid C$; otherwise $A \not\perp B \mid C$. The procedure can be implemented using a breadth-first search.

Markov Blanket: For DGM, the Markov blanket of a node X_i comprises the set of parents, children and co-parents of X_i and $P(x_i|x_1,...,x_{i-1},x_{i+1},...,x_N) = P(x_i|MB(x_i))$. For UGM, the Markov blanket of a node X_i is its direct neighbour.

MARKOV RANDOM FIELDS (UGM)

Rules of Conditional Independence: $X|Y$ means $P(X,Y) = P(X)P(Y)$

1. **Symmetry:** $X \perp Y \mid Z \Rightarrow Y \perp X \mid Z$.
2. **Decomposition:** $X \perp \{Y,W\} \mid Z \Rightarrow X \perp Y \mid Z$ and $X \perp W \mid Z$.
3. **Weak Union:** $X \perp \{Y,W\} \mid Z \Rightarrow X \perp Y \mid \{W,Z\}$ and $X \perp W \mid \{Y,Z\}$.
4. **Contraction:** $X \perp Y \mid \{W,Z\}$ and $X \perp W \mid Z \Rightarrow X \perp \{Y,W\} \mid Z$.
5. **Intersection:** $X \perp Y \mid \{W,Z\}$ and $X \perp W \mid \{Y,Z\} \Rightarrow X \perp \{Y,W\} \mid Z$.

I-MAP: All independences in probability distribution P or graph G.

Conditional independence for UGMs: (1) global Markov property: Given node sets A,B,C, we have $X_A \perp X_B \mid X_C$ if C separates A from B in graph G. (2) local Markov property: A node X_s is conditionally independent of all others given its Markov blanket: $X_s \perp V \setminus \{\text{mb}(X_s), X_s\} \mid \text{mb}(X_s)$. (3) pairwise Markov property: Two nodes X_s and X_t are conditionally independent given the rest nodes if no edge connects them: $X_s \perp X_t \mid V \setminus \{X_s, X_t\}$, where $\mathscr{E}_{st} = \varnothing$.

Hammersley–Clifford Theorem: A positive distribution $p(y) > 0$ satisfies the CI properties of an undirected graph G if p can be written as a product of factors, one per maximal clique: $p(y \mid \theta) = \frac{1}{Z(\theta)} \prod_{c \in C} \psi_c(y_c \mid \theta_c)$

Moralization: A DGM can be converted into a UGM by marrying the unmarried parents of a node.

Conditional Random Fields: A CRF is an MRF where all clique potentials are conditioned on all observed variables x: $p(y \mid x,w) = \frac{1}{Z(x,w)} \prod_c \psi_c(y_c \mid x,w)$.. MRF models $P(x,y)$, while CRF models $P(y|x)$.

VARIABLE ELIMINATION AND BELIEF PROPAGATION

Reconstituted Graph: Elimination means removing a node from the graph and connecting all its remaining neighbors. The original and newly created edges are recorded in the reconstituted graph.

Variable Elimination

1: **Goal:** Compute $p(x_F \mid x_E)$ via Variable Elimination on a directed graph.
2: **Initialize**(G,F): choose elimination order I with F last; for each node X_i, place its CPT $p(x_i \mid x_{\pi_i})$ into the active list.
3: **Evidence**(E): for each evidence variable $i \in E$, add $\delta(x_i,\bar{x}_i)$ to the active list.
4: **Update**(G) *(marginalization)*: for each variable i in ordering I:
5: collect all active factors involving x_i and multiply potentials to get $\phi_i(x_i)$
6: Example: $(\phi_5(x_2,x_3) = p(x_5 \mid x_3)m_6(x_2,x_5))$
7: marginalize: $m_i(x_{\setminus i}) = \sum_{x_i} \phi_i(x_i)$ (Example: $m_5(x_2,x_3) = \sum_{x_5} p(x_5 \mid x_3)m_6(x_2,x_5)$)
8: replace old factors with m_i in the active list
9: **Normalize**(F): obtain the desired conditional distribution $p(x_F \mid x_E) = \frac{\phi_F(x_F)}{\sum_{x_F} \phi_F(x_F)}$.

Treewidth: one less than the smallest achievable cardinality of the largest clique over all possible elimination orderings. The maximum clique wil influence the computation complexity.

Sum-Product Algorithm

1: **Initialize:** For each node $i \in V$, set $\psi_i^E(x_i) = \begin{cases} \psi_i(x_i)\,\delta(x_i,\bar{x}_i), & i \in E, \\ \psi_i(x_i), & i \notin E. \end{cases}$
2: Choose a root r, perform inward pass (collect): For each edge $(j \to i)$ toward r send $m_{j\to i}(x_i) = \sum_{x_j} \psi_j^E(x_j)\psi_{ij}(x_i,x_j)\prod_{k\in N(j)\setminus i} m_{k\to j}(x_j)$.
3: Outward pass (distribute): For each edge $(i \to j)$ away from r send $m_{i\to j}(x_j) = \sum_{x_i} \psi_i^E(x_i)\psi_{ij}(x_i,x_j)\prod_{k\in N(i)\setminus j} m_{k\to i}(x_i)$.
4: **Marginal:** For any node i, $p(x_i|E) \propto \psi_i^E(x_i)\prod_{j\in N(i)} m_{j\to i}(x_i)$.

FACTOR GRAPH AND JUNCTION TREE ALGORITHM

Joint probability of factor graph: $p(x) = \prod_s f_S(x_S)$. The method for constructing factor graphs is based on the original joint distribution. Taking DGMs as an example, if it's $P(X_i)$, there's an independently connected factor of X_i; if it's a conditional probability, this factor is between multiple variables.

Messages from leaf node: (1) message from leaf variable $\to$ factor: $v_{is}(x_i) = 1$; (2) message from leaf factor $\to$ variable: $\mu_{si}(x_i) = f_s(x_i)$

Sum-Product of graph factor

1: **Evidence incorporation:** $\psi_i^E(x_i) = \psi_i(x_i)\delta(x_i,\bar{x}_i)$ if $i \in E$, else $\psi_i^E(x_i) = \psi_i(x_i)$.
2: **Collect (leaf $\to$ root):** $\mu_{si}(x_i) = \sum_{x_{N(s)\setminus i}} f_S(x_{N(s)}) \prod_{j\in N(s)\setminus i} v_{js}(x_j)$,
$v_{is}(x_i) = \prod_{t\in N(i)\setminus s} \mu_{ti}(x_i)$.
3: **Distribute (root $\to$ leaves):** $v_{is}(x_i) = \prod_{t\in N(i)\setminus s} \mu_{ti}(x_i)$,
$\mu_{si}(x_i) = \sum_{x_{N(s)\setminus i}} f_S(x_{N(s)}) \prod_{j\in N(s)\setminus i} v_{js}(x_j)$.
4: **Marginal:** $p(x_i) \propto \psi_i^E(x_i)\prod_{s\in N(i)} \mu_{si}(x_i)$.

MAP-Elimination Algorithm

1: **Input:** graph $G(\mathscr{V},\mathscr{E})$, evidence set E
2: **Initialize:** choose elimination order I; for each $X_i \in \mathscr{V}$ place $p(x_i \mid x_{\pi_i})$ on active list
3: **Evidence:** for each $i \in E$ place $\delta(x_i,\bar{x}_i)$ on active list
4: **Update (maximization):** for each $i \in I$:
- collect all active factors involving x_i and remove them, then multiply them: $\phi_i^{\max}(x_{T_i}) = \prod \phi$
- maximize out x_i: $m_i^{\max}(x_{S_i}) = \max_{x_i} \phi_i^{\max}(x_{T_i})$ and place $m_i^{\max}(x_{S_i})$ back on the active list

5: **Maximum:** final scalar on active list is $\max_x p^E(x)$

Max-Product for trees

1: **Input:** tree $\mathscr{T}(\mathscr{V},\mathscr{E})$, evidence set E
2: **Evidence:** for each $i \in \mathscr{V}$ set $\psi_i^E(x_i) = \psi(x_i)\mathbf{1}[i \notin E] + \psi(x_i)\delta(x_i,\bar{x}_i)\mathbf{1}[i \in E]$
3: $f = \text{ChooseRoot}(\mathscr{V})$
4: **Inward pass (collect):** for $s \in N(f)$ do Collect(f,s), where Collect(i,j) : for $k \in N(j)\setminus\{i\}$ do Collect(j,k); $\mu_{j\to i}(x_i) = \max_{x_j} \left(\psi_j^E(x_j)\psi(x_i,x_j)\prod_{k\in N(j)\setminus\{i\}} \mu_{k\to j}(x_j)\right)$;
$\delta_{ji}(x_i) = \arg\max_{x_j} \left(\psi_j^E(x_j)\psi(x_i,x_j)\prod_{k\in N(j)\setminus\{i\}} \mu_{k\to j}(x_j)\right)$
5: **Root MAP at X_f:** $x_f^* = \arg\max_{x_f} \left(\psi_f^E(x_f)\prod_{e\in N(f)} \mu_{e\to f}(x_f)\right)$
6: **Outward pass (distribute):** for $s \in N(f)$ do Distribute(f,s), where Distribute(i,j) : set $x_j^* = \delta_{ji}(x_i^*)$; for $k \in N(j)\setminus\{i\}$ do Distribute(j,k)
7: **Output:** MAP configuration $\{x_i^*\}_{i\in\mathscr{V}}$

Cluster Graph: An undirected graph with: (1)nodes are clusters $C_i \subseteq \{X_1,\ldots,X_n\}$. (2)Edge between C_i and C_j has sepset $S_{ij} = C_i \cap C_j$. Cluster potential: $\phi_j(C_j) = \prod_{\psi:\alpha(\psi)=j} \psi$, $\prod_\psi \psi = \prod_j \phi_j$ (each ψ used once).

Running Intersection Property (Junction Tree Property): For each pair of clusters C_i, C_j and variable $X \in C_i \cap C_j$ satisify – There exists a unique path between C_i and C_j such that all clusters and sepsets on the path contain X. (Junction tree is cluster graph which satisfies running intersection property.)

Junction Tree : Sum-Product Algorithm: Use the sum-product algorithm to compute messages from C_i to C_j: $\delta_{i\to j} = \sum_{C_i\setminus S_{ij}} \phi_i \prod_{k\in N(i)\setminus j} \delta_{k\to i}$ (Marginalize C_i not in S_{ij}). The unnormalized marginal probability of clique C_i is: $\tilde{p}(C_i) = \phi_i \prod_{k\in N(i)} \delta_{k\to i}$.

Constructing Junction Tree: (1) Triangulation: Get the reconstituted graph (For DGM, first for Moralization); (2) Get all clusters and all possible sepsets: Use eliminate cliques as clusters, a possible sepset is $S_{ij} = C_i \cap C_j$; (3) Assign cluster potentials: cluster potentials are formed by condition probabilities (DGM), or potentials (UGM); (4) Get junction tree: find the maximum spanning tree with cardinality of sepsets as weight of edges.

KRUSKAL for maximum spanning tree: Sort all edges in descending order of their weights (size of sepsets). Iterate through these edges one by one. For each edge, if adding it to the tree being built does not create a cycle, then adopt it.

PARAMETER LEARNING WITH COMPLETE DATA

Unknown parameter learning for DGMS: A set of N independent and identically distributed complete observations of each random variable $X = \{x_{1,1},\ldots,x_{1,N},\ldots,x_{M,1},\ldots,x_{M,N}\}$. Two commonly used approaches to learn the unknown parameters θ:

1. Maximum likelihood estimate (MLE):
$\hat{\theta} = \arg\max_\theta\, p(x_1,\ldots,x_M \mid \theta) = \arg\max_\theta \prod_{i=1}^N p(x_{1,i},\ldots,x_{M,i} \mid \theta)$.
2. Maximum a posteriori (MAP):
$\hat{\theta} = \arg\max_\theta\, p(\theta \mid x_1,\ldots,x_M) == \arg\max_\theta \left[\prod_{i=1}^N p(x_{1,i},\ldots,x_{M,i} \mid \theta)\right] p(\theta)$.

When the $P(x|\theta)$ is known, we can use derivative = 0 to compute optimal parameters.

Unknown parameter learning for UGMs: Gradient of the log-likelihood can be rewritten as: $\frac{\partial l}{\partial \theta_C} = \mathbb{E}_{\text{emp}}[\phi_C(y)] - \mathbb{E}_{p(y|\theta)}[\phi_C(y)] = \frac{1}{N}\sum_{i=1}^N \phi_C(y_i) - \mathbb{E}_{p(y|\theta)}[\phi_C(y)]$.

Stochastic maximum likelihood for fitting an MRF

1: Initialize parameters θ randomly; set $k = 0$, $\eta = 1$.
2: For each epoch:
3: Split observed data $\{y_i\}_{i=1}^N$ into minibatches B of size B.
4: For each minibatch B:
5: Draw S model samples: $\{y^{s,k} \sim p(y \mid \theta_k)\}_{s=1}^S$.
6: Compute model expectation: $\hat{\phi} = \frac{1}{S}\sum_{s=1}^S \phi(y^{s,k})$.
7: Compute per-sample gradients in minibatch: $g_{i,k} = \phi(y_i) - \hat{\phi}$, $i \in B$.
8: Minibatch gradient: $g_k = \frac{1}{B}\sum_{i\in B} g_{i,k}$.
9: Parameter update: $\theta_{k+1} = \theta_k - \eta g_k$.
10: Increment iteration: $k \leftarrow k+1$.Then, decrease step size η.

Iterative Proportional Fitting (IPF) for Tabular MRFs

1: Initialize clique potentials $\psi_C(y_C) = 1$ for all $c = 1,\ldots,C$.
2: **for** $c = 1$ to C **do**
3: **Model marginal:** $p_C(y_C \mid \psi) = \sum_{y\setminus y_C} p(y \mid \psi)$
4: **Empirical marginal:** $\hat{p}_C(y_C) = p_{\text{emp}}(y_C) = \frac{1}{N}\sum_{i=1}^N \delta(y_{C,i}, y_C)$
5: **IPF multiplicative update:** $\psi_C(y_C) \leftarrow \psi_C(y_C)\,\frac{\hat{p}_C(y_C)}{p_C(y_C|\psi)}$
6: **end for**

MIXTURE MODEL AND EM ALGORITHM

Mixture Models: Probabilistic models formed by taking linear combinations of more basic distributions.

GMM: The probability distribution of a mixture of Gaussians is given by the superposition of K Gaussian densities: $p(\mathbf{x}) = \sum_{k=1}^K \pi_k \mathscr{N}(\mathbf{x} \mid \mu_k, \Sigma_k)$, where $\sum_{k=1}^K \pi_k = 1$.

EM Algorithm

1: **Goal:** Maximize $\ln p(\mathbf{X}|\theta) = \ln \sum_{\mathbf{Z}} p(\mathbf{X},\mathbf{Z}|\theta)$
2: **while** Not Converged **do**
3: **E-Step**: compute posterior $p(\mathbf{Z}|\mathbf{X},\theta^{\text{old}})$
4: $\gamma(k) = \frac{\pi_k \mathscr{N}(\mathbf{x}|\mu_k,\Sigma_k)}{\sum_{j=1}^K \pi_j \mathscr{N}(\mathbf{x}|\mu_j,\Sigma_j)} = \frac{p(\mathbf{Z}=\mathbf{k}|\mathbf{X},\theta^{\text{old}})}{\sum_k p(\mathbf{Z}=\mathbf{k}|\mathbf{X},\theta^{\text{old}})}$
5: $Q(\theta,\theta^{\text{old}}) = \mathbb{E}_{\mathbf{Z}|\mathbf{X},\theta^{\text{old}}}[\ln p(\mathbf{X},\mathbf{Z}|\theta)] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X},\theta^{\text{old}}) \ln p(\mathbf{X},\mathbf{Z}|\theta) = \sum_k \gamma(k) \ln p(\mathbf{X},\mathbf{Z}|\theta)$
6: **M-Step**: $\theta^{\text{new}} = \arg\max_\theta\, Q(\theta,\theta^{\text{old}})$ (derivative = 0)
7: **Check**: $\ln p(\mathbf{X}|\theta^{\text{new}}) - \ln p(\mathbf{X}|\theta^{\text{old}}) < \varepsilon$
8: **end while**

HIDDEN MARKOV MODELS

Jiont probability: $p(z_{1:T},x_{1:T}) = p(z_1)\prod_{t=2}^T p(z_t \mid z_{t-1})\prod_{t=1}^T p(x_t \mid z_t)$

Transition matrix: Properties of the state transition matrix $A \in \mathbb{R}^{K\times K}$: (1)$A_{jk} \equiv p(z_{nk} = 1 \mid z_{n-1,j} = 1)$, (2)$0 \le A_{jk} \le 1$ with $\sum_k A_{jk} = 1$, (3)and there are $K(K-1)$ independent parameters.

Pairwise MRF: $p(\mathbf{x}) = \frac{1}{Z}\prod_{i\in V}\psi_i(x_i)\prod_{(i,j)\in E}\psi_{ij}(x_i,x_j)$

Viterbi Algorithm: The maximal probability of the joint distribution $p(x_1,\ldots,x_N,z_1,\ldots,z_N)$ is given by the max of $\omega(z_N)$ at the root node. $\max_{z_1,\ldots,z_N} p(x_1,\ldots,x_N,z_1,\ldots,z_N) = \max_{z_N} \omega(z_N)$.
$\omega(z_1) = \ln p(z_1) + \ln p(x_1 \mid z_1)$. $\omega(z_{n+1}) = \ln p(x_{n+1} \mid z_{n+1}) + \max_{z_n}\{\ln p(z_{n+1} \mid z_n) + \omega(z_n)\}$.

MONTE CARLO INFERENCE

Monte Carlo Principle: Draw samples $x^{(i)} \sim p(x)$ i.i.d.; define empirical measure $P_N(x) = \frac{1}{N}\sum_{i=1}^N \delta_{x^{(i)}}(x)$. For any function f, define $I(f) = \int f(x)p(x)dx$ and estimator $I_N(f) = \frac{1}{N}\sum_{i=1}^N f(x^{(i)})$. By law of large numbers, $I_N(f) \xrightarrow[N\to\infty]{\text{a.s.}} I(f)$. If $\text{Var}[f(X)] = \sigma_f^2 < \infty$, then $\text{Var}[I_N(f)] = \sigma_f^2/N$. By central limit theoremcentral limit theorem, $\sqrt{N}(I_N(f) - I(f)) \xrightarrow{d} \mathscr{N}(0,\sigma_f^2)$.
Summary: By independently sampling from the target distribution $p(x)$ to construct the empirical measure P_N and using the corresponding sample mean $I_N(f)$ to estimate the integral $I(f)$, one can rely on the law of large numbers and the central limit theorem as $N \to \infty$ to ensure consistency and asymptotic normality, thereby achieving a reliable numerical approximation of complex integrals or expectations.

Rejection Sampling:$p(z) = \frac{1}{Z_p}\tilde{p}(z)$, but $\tilde{p}(z)$ is difficult to sample so we need a common distribution $q(z)$. Sample $z^{(i)} \sim q(z)$ and $u \sim U(0,1)$, if $u < \frac{\tilde{p}(z^{(i)})}{kq(z^{(i)})}$, then accept $z^{(i)}$.

Importance Sampling: To estimate expectations under a target distribution $p(z)$, we draw samples from an easier proposal distribution $q(z)$ and correct the mismatch using importance weights. Using the identity $\mathbb{E}_p[f(z)] = \int f(z)\frac{p(z)}{q(z)}q(z)dz$, the Monte Carlo estimator becomes $\frac{1}{L}\sum_{l=1}^L \frac{p(z^{(l)})}{q(z^{(l)})} f(z^{(l)})$ with $z^{(l)} \sim q(z)$. When only unnormalized densities $\tilde{p}(z)$ and $\tilde{q}(z)$ are available, self-normalized importance sampling is used: $\mathbb{E}_p[f(z)] \approx \sum_{l=1}^L w_l f(z^{(l)})$, where $w_l = \frac{\tilde{p}(z^{(l)})/\tilde{q}(z^{(l)})}{\sum_m \tilde{p}(z^{(m)})/\tilde{q}(z^{(m)})}$.

Ancestral Sampling: Given a joint distribution factorized as $p(z) = \prod_{i=1}^M p(z_i \mid \text{pa}_i)$ in a Bayesian network, we can generate exact samples by drawing each variable in topological order. For root nodes, sample directly from $p(z_1)$; for each subsequent node, sample $z_i \sim p(z_i \mid \text{pa}_i)$ using already-sampled parent values. Repeating this for $i = 1,\ldots,M$ yields a complete sample from the joint distribution $p(z)$.

MCMC (Metropolis–Hastings):

1: Initialize $x^{(0)}$
2: **for** $i = 0$ to $N-1$ **do**
3: Sample $u \sim U_{[0,1]}$, $x' \sim q(x' \mid x^{(i)})$ ▷ draw acceptance threshold and proposal sample
4: **If** $u < \mathscr{A}(x',x^{(i)}) = \min\left\{1, \frac{\tilde{p}(x')q(x^{(i)}|x')}{\tilde{p}(x^{(i)})q(x'|x^{(i)})}\right\}$, $x^{(i+1)} = x'$
5: **end for**

Gibbs Sampling:

1: Initialize $\{x_i : i = 1,\ldots,M\}$
2: **for** $\tau = 1,\ldots,T$ **do**
3: Sample $x_1^{(\tau+1)} \sim p(x_1 \mid x_2^{(\tau)}, x_3^{(\tau)}, \ldots, x_M^{(\tau)})$, $x_2^{(\tau+1)} \sim p(x_2 \mid x_1^{(\tau+1)}, x_3^{(\tau)}, \ldots, x_M^{(\tau)})$...
4: Sample $x_M^{(\tau+1)} \sim p(x_M \mid x_1^{(\tau+1)}, x_2^{(\tau+1)}, \ldots, x_{M-1}^{(\tau+1)})$
5: **end for**

VARIANCE INFERENCE

Mean Field Approximation: Given a joint probability, the goal of MFA is to find a $q(z)$ for $p(z|x)$, which satisfies $q(z) = \prod q_i(z_i)$.

Mean Field Approximation

1: **Goal:** Derive $q_j^*(\mathbf{Z}_j)$ given Joint $p(\mathbf{X},\mathbf{Z})$, q_j^* is normalized q_j
2: $\ln q_j^*(\mathbf{Z}_j) = \mathbb{E}_{i\neq j}[\ln p(\mathbf{X},\mathbf{Z})] + \text{const}$
3: **while** Not Converged **do**
4: **Log-joint**:$\ln p = \sum \ln p(\text{factors})$ (Expand all terms)
5: **Drop irrelevant terms**: Keep terms with $\mathbf{Z}_j$, others $\to$ const
6: **Expectation**: Replace neighbors $\mathbf{Z}_{i\neq j} \to \mathbb{E}[\mathbf{Z}_i]$
7: **Match & update**: $-A\mathbf{Z}_j^2 + B\mathbf{Z}_j \Longrightarrow \mathscr{N}(B/2A, 1/2A)$, $(A-1)\ln \mathbf{Z}_j - B\mathbf{Z}_j \Longrightarrow \text{Gam}(A,B)$, $\sum Z_{jk} \ln \pi_k \Longrightarrow \text{Cat}(\pi)$
8: Compute new moment $\mathbb{E}[\mathbf{Z}_j]$ using new params
9: **end while**

Forward vs Backward KL-Divergence: KL is not symmetrical, minimize $KL(q\|p) \neq KL(p\|q)$. Reverse KL: $KL(q\|p) = \sum_z q(z)\ln\frac{q(z)}{p(z)}$, if $p(z) = 0$, $q(z) = 0 \Rightarrow q$ under-estimate p. Forward KL: $KL(p\|q) = \sum_z p(z)\ln\frac{p(z)}{q(z)}$, if $p(z) > 0$, $q(z) > 0 \Rightarrow q$ over-estimate p. Choose the Backward KL: $KL(q\|p)$ 1. Tractable lower bound. 2. Statically more sensible.

Example 1: The Univariate Gaussian Observed: $D = \{x_1,\ldots,x_N\}$, Goal: infer posterior. Likelyhood: $p(D \mid \mu,\tau) = \left(\frac{\tau}{2\pi}\right)^{N/2}\exp\{-\frac{\tau}{2}\sum_n(x_n-\mu)^2\}$, conjugate prior: $p(\mu \mid \tau) = N(\mu \mid \mu_0,(\lambda_0\tau)^{-1})$, $p(\tau) = \text{Gam}(\tau \mid a_0,b_0)$, factorized: $q(\mu,\tau) = q_\mu(\mu)q_\tau(\tau)$, optimal solutions: $q_\mu^*(\mu) = N(\mu \mid \mu_N,\lambda_N^{-1})$, $\mu_N = \frac{\lambda_0\mu_0+N\bar{x}}{\lambda_0+N}$, $\lambda_N = (\lambda_0+N)E[\tau]$; $q_\tau^*(\tau) = \text{Gam}(\tau \mid a_N,b_N)$, $a_N = a_0 + \frac{N+1}{2}$, $b_N = b_0 + \frac{1}{2}E_\mu[\sum_n(x_n-\mu)^2] + \lambda_0(\mu-\mu_0)^2$. Solutions are coupled $\Rightarrow$ iteration approach: (1) Initial guess for $E[\tau]$, recompute $q_\mu^*(\mu)$. (2) Use revised $q_\mu^*(\mu)$ to extract the moments $E[\mu]$ and $E[\mu^2]$, use these to re-compute $q_\tau(\tau)$. (3) Use revised $q_\tau(\tau)$ to extract the moment $E[\tau]$, and use this to re-compute $q_\mu^*(\mu)$. (4) Repeat until convergence.

Example 3: Ising Model $p(x,y) = p(x)p(y \mid x)$, prior: $p(x) = \frac{1}{Z_0}\exp(E_0(x))$, $E_0(x) = -\sum_i\sum_{j\in\text{nbr}_i} W_{ij}x_ix_j$. Likelihood: $p(y \mid x) = \prod_i p(y_i \mid x_i) = \exp(\sum_i L_i(x_i))$, $\ln p(x,y) = \sum_i\{x_i\sum_{j\in\text{nbri}} x_{ij}x_j + L(x_i)\} - \ln Z_0$, optimized factor: $\log q_i(x_i) = E_{x_j}[\log p(x,y)] + \text{const}$, thus $q_i(x_i) \propto \exp(x_i\sum_{j\in\text{nbri}} W_{ij}\mu_j + L_i(x_i))$, $\mu_j = \sum_{x_j} x_j q_j(x_j)$.

MAP INFERENCE AND AUGMENTING PATHS

$E(w) = \sum_n U_n(w_n) + \sum_{(m,n)} P_{mn}(w_m,w_n)$

Cut and capacity: A cut is a node partition (S,T) on $\mathscr{G}$ such that $s \in S$ and $t \in T$, where $S \cap T = \varnothing$ and $S \cup T = V$. Capacity(S,T) is the sum of weights of edges leaving S.

MAP: Discussing different cases and first identify possible 2^n solutions (having different cuts), the solution with minimun $E(w)$ is the min-cut. The cut must sever the path from S to T. Whether a pixel belongs to the S or T depends on how it was cut. If the S was cut off, then it belongs to the T.

Converting MRF: In some cases, you may need to perform a graph transformation first, converting it into a standard form based on the number of observed variables. Considering S is labe '1' and T is labe '0', so we regard 'right' direction as $1 \to 0$.

Augment: Iterate using the shortest path first.

EXAMPLE 1: EM

We are given two biased coins, Coin A and Coin B, each with an unknown probability of landing heads. In a series of trials, we randomly choose one of the two coins and flip it a single time. However, we do not know which coin was chosen for each flip. Let us denote the random variables of choosing each coin as a 1-of-2 representation $Z_k \in \{0,1\}$, where $Z_k = 0 :=$ Coin A and $Z_k = 1 :=$ Coin B. The outcome of a coin flip is $X \in \{0,1\}$, where $X = 0 :=$ Head and $X = 1 :=$ Tail. We denote the probability of choosing Coin B as π, and the probability of getting a tail for Coin A and B as μ_0 and μ_1 respectively.

a) Given N i.i.d observations of the coin flips, draw the Bayesian network and write the joint probability distribution that represents the coin choosing and flipping process.

The probability of choosing each coin is $p(Z \mid \pi) = \prod_k \pi_k^{Z_k}$, where $\pi_0 = 1-\pi$ and $\pi_1 = \pi$.
The likelihood of outcome X given the coin is $p(X \mid \mu_k) = \mu_k^X(1-\mu_k)^{1-X}$.
Thus, the joint distribution is $p(X,Z) = p(Z)p(X \mid Z) = \prod_n \prod_k \pi_k^{Z_{nk}} p(X_n \mid \mu_k)^{Z_{nk}}$.

b) Using the EM algorithm, derive the expressions of μ_0 and μ_1 after the Maximization step.

The EM objective is $Q(\theta,\theta^{old}) = \sum_Z p(Z \mid X,\theta^{old})\ln p(X,Z \mid \theta)$.
Let responsibilities be $\gamma(Z_{nk}) = p(Z_{nk} = 1 \mid X_n,\theta^{old})$.
Then, $Q(\theta,\theta^{old}) = \sum_n\sum_k \gamma(Z_{nk})\{\ln\pi_k + \ln p(X_n \mid \mu_k)\}$.
Taking derivative w.r.t. μ_k: $\frac{\partial Q}{\partial \mu_k} = \sum_n \gamma(Z_{nk})\left(\frac{X_n}{\mu_k} - \frac{1-X_n}{1-\mu_k}\right) = 0$.
Solving gives $\mu_k = \frac{\sum_n \gamma(Z_{nk})X_n}{\sum_n \gamma(Z_{nk})}$, where $N_k = \sum_n \gamma(Z_{nk})$, then $\mu_k = \frac{1}{N_k}\sum_n \gamma(Z_{nk})X_n$.

c) The results of these flips are recorded as a sequence of heads (H) and tails (T): $\{H,T,H\}$. Taking equal probability of choosing either Coin A or B and the initial values of $\mu_0 = 0.6$ and $\mu_1 = 0.5$, perform one iteration of the EM algorithm to update μ_0 and μ_1.

The likelihood is $p(X \mid \mu_k) = \mu_k^X(1-\mu_k)^{1-X}$ and responsibilities: $\gamma(Z_{nk}) = \frac{\pi_k p(X_n|\mu_k)}{\sum_j \pi_j p(X_n|\mu_j)}$.

For $X_1 = 0$: $\gamma(Z_{1,0}) = \frac{0.5(1-\mu_0^{old})}{0.5(1-\mu_0^{old})+0.5(1-\mu_1^{old})} = \frac{1-0.6}{(1-0.6)+(1-0.5)} = \frac{0.4}{0.4+0.5} = 0.444$.

For $X_2 = 1$: $\gamma(Z_{2,0}) = \frac{0.5\mu_0^{old}}{0.5\mu_0^{old}+0.5\mu_1^{old}} = \frac{0.6}{0.6+0.5} = 0.545$.

For $X_3 = 0$: $\gamma(Z_{3,0}) = 0.444$.

Total responsibility: $N = 0.444 + 0.545 + 0.444 = 1.433$.

Parameter update: $\mu_0 = \frac{1}{N}\sum_n \gamma(Z_{n0})X_n = \frac{0.545}{1.433} = 0.380$.

EXAMPLE 2: EM

Figure 9.1 shows a Bayesian network with both binary and continuous state latent random variables, i.e., $Z \in \{0,1\}$ and $T \in \mathbb{R}$. In addition, $X = 0.5$ is the observed random variable. The maximum log-likelihood of T: $\arg\max_T \log p(T \mid X)$, can be obtained from the Expectation-Maximization (EM) algorithm. The EM algorithm iterates between the Expectation step that evaluates the expected complete data log-likelihood with respect to $p(Z \mid X,T^{old})$ and the Maximization step that maximizes T over the expected complete data log-likelihood with respect to $p(Z \mid X,T^{old})$. T^{old} is the value of T from the previous iteration of the EM algorithm. $\{\lambda = 0.1, w_{a0} = 0.5, w_{a1} = 0.5, w_{b0} = 0.8, w_{b1} = 0.2, \tau_a = 1.0, \tau_b = 1.2, U = 0.6\}$ are known hyperparameters of the following distributions:
$p(Z) = \lambda^Z(1-\lambda)^{(1-Z)}$, $p(X \mid T,Z) = \mathscr{N}(X \mid w_{a0}+w_{a1}T,\tau_a)^Z\mathscr{N}(X \mid w_{b0}+w_{b1}T,\tau_b)^{(1-Z)}$,
$\mathscr{N}(X \mid w_0+w_1T,\tau) = \sqrt{\frac{\tau}{2\pi}}\exp\{-0.5\tau(X-w_0-w_1T)^2\}$, $p(T) = U$.

a. Derive the expression for the posterior $p(Z \mid X,T^{old})$ from the Bayesian Network.

b. Derive the expression for T that maximizes the expected complete data log-likelihood with respect to $p(Z \mid X, T^{old})$.

c. Given the initial value of $T = 2.0$, find the value of T in the next EM iteration.

Answer:

Joint distribution: $p(X,Z,T) = p(T)p(Z)p(X \mid T,Z) = p(Z)p(X \mid T,Z)$.

a. $p(Z \mid X, T^{old}) = \dfrac{p(Z)p(X \mid T,Z)}{\sum_Z p(Z)p(X \mid T,Z)} = \dfrac{p(Z)p(X \mid T,Z)}{\sum_Z p(Z)p(X \mid T,Z)}$.

$= \dfrac{\lambda^Z(1-\lambda)^{(1-Z)}\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)^Z \mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)^{(1-Z)}}{\sum_Z \lambda^Z(1-\lambda)^{(1-Z)}\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)^Z \mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)^{(1-Z)}}$.

$= \dfrac{\lambda^Z(1-\lambda)^{(1-Z)}\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)^Z \mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)^{(1-Z)}}{\lambda\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)+(1-\lambda)\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)}$.

Let's define:

$\gamma(Z=0) = p(Z=0 \mid X, T^{old}) = \dfrac{(1-\lambda)\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)}{\lambda\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)+(1-\lambda)\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)}$,

$\gamma(Z=1) = p(Z=1 \mid X, T^{old}) = \dfrac{\lambda\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)}{\lambda\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)+(1-\lambda)\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)}$.

b. $Q = \sum_Z p(Z \mid X, T^{old}) \ln p(X,Z \mid T) = \sum_Z \gamma(Z) \ln p(Z)p(X \mid T,Z)$

$= \gamma(Z=0)\ln p(Z=0)p(X \mid T,Z=0) + \gamma(Z=1)\ln p(Z=1)p(X \mid T,Z=1)$

$= \gamma(Z=0)\ln\{(1-\lambda)\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)\} + \gamma(Z=1)\ln\{\lambda\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)\}$

$0 = \gamma(Z=0)\dfrac{\partial}{\partial T}\{\ln(1-\lambda) + \ln\mathcal{N}(X \mid w_{b0}+w_{b1}T, \tau_b)\} + \gamma(Z=1)\dfrac{\partial}{\partial T}\{\ln\lambda + \ln\mathcal{N}(X \mid w_{a0}+w_{a1}T, \tau_a)\}$

$T = \dfrac{X(\gamma(Z=0)w_{b1}\tau_b + \gamma(Z=1)w_{a1}\tau_a) - \gamma(Z=0)w_{b0}w_{b1}\tau_b - \gamma(Z=1)w_{a0}w_{a1}\tau_a}{\gamma(Z=0)w_{b1}^2\tau_b + \gamma(Z=1)w_{a1}^2\tau_a}$

c. Just use $T = 2.0$ to compute items in T.

EXAMPLE 3: VARIANCE INFERENCE

Given a dataset $\mathcal{D} = \{x_1, \ldots, x_N\}$ of observed values of a random variable X, which are assumed to be drawn independently from a univariate Gaussian distribution: $p(x \mid \mu, \tau) = \mathcal{N}(x \mid \mu, \tau^{-1})$ parameterized by the mean μ and precision τ. The prior distributions for μ and τ are given by a univariate Gaussian distribution: $p(\mu \mid \tau) = \mathcal{N}(\mu \mid \mu_0, (\lambda_0\tau)^{-1})$ and a Gamma distribution: $p(\tau) = \text{Gam}(\tau \mid a_0, b_0)$, respectively. $\{\mu_0, \lambda_0, a_0, b_0\}$ are hyperparameters of the prior distributions. (i) Using variation inference and mean field approximation, derive the distributions $q(\mu)$ and $q(\tau)$ that best approximate the conditional distributions $p(\mu \mid \mathcal{D})$ and $p(\tau \mid \mathcal{D})$.

(ii) Derive the expressions for the expected mean μ and precision τ under the approximated distributions $q(\mu)$ and $q(\tau)$.

(iii) Derive the expressions for the mean μ and precision τ that *maximize the log-likelihood*.

(iv) Comment on the difference** between the mean μ and precision τ obtained from variational inference and maximum log-likelihood.

1. Gamma distribution: $\text{Gam}(\tau \mid a,b) = \dfrac{1}{\Gamma(a)} b^a \tau^{a-1} \exp(-b\tau)$, where $\Gamma(a)$ is the Gamma function;

2. Gaussian distribution: $p(x \mid \mu, \tau^{-1}) = \left(\dfrac{\tau}{2\pi}\right)^{1/2} \exp\left\{-\dfrac{\tau}{2}(x-\mu)^2\right\}$;

3. Completing the square: $ax^2 + bx + c = 0 \implies a(x+d)^2 + e = 0$, where $d = \dfrac{b}{2a}, e = c - \dfrac{b^2}{4a}$.

Solution

i) Joint probability distribution: $p(D,\mu,\tau) = p(D \mid \mu,\tau)\,p(\mu \mid \tau)\,p(\tau)$

$= \prod_n \mathcal{N}(x_n \mid \mu, \tau^{-1})\,\mathcal{N}(\mu \mid \mu_0, (\lambda_0\tau)^{-1})\,\text{Gam}(\tau \mid a_0, b_0)$ Mean field approximation: $q(\mu,\tau) = q_\mu(\mu)\,q_\tau(\tau)$ The optimal factors $q_\mu(\mu)$ and $q_\tau(\tau)$ are obtained from variational inference: $\ln q_j^*(Z_j) = \mathbb{E}_{i \neq j}\left[\ln p(X,Z)\right] + \text{const.}$

We get: $\ln q_\mu^*(\mu) = \mathbb{E}_\tau[\ln p(D,\mu,\tau)] + \text{const}$

$= \mathbb{E}_\tau\left[\ln\left(\left(\frac{\tau}{2\pi}\right)^{N/2}\exp\{-\frac{\tau}{2}\sum_n(x_n-\mu)^2\}\right) + \ln\mathcal{N}(\mu \mid \mu_0, (\lambda_0\tau)^{-1})\right] + \text{const}$

$= \mathbb{E}_\tau\left[\ln\left(\left(\frac{\tau}{2\pi}\right)^{N/2}\exp\{-\frac{\tau}{2}\sum_n(x_n-\mu)^2\}\right) + \ln\left(\left(\frac{\lambda_0\tau}{2\pi}\right)^{1/2}\exp\{-\frac{\lambda_0\tau}{2}(\mu-\mu_0)^2\}\right)\right] + \text{const}$

$= \mathbb{E}_\tau\left[\sum_n\left(-\frac{\tau}{2}(x_n-\mu)^2\right) - \frac{\lambda_0\tau}{2}(\mu-\mu_0)^2\right] + \text{const} \;=\; -\frac{\mathbb{E}_\tau[\tau]}{2}\left[\sum_n(x_n-\mu)^2 + \lambda_0(\mu-\mu_0)^2\right] + \text{const}$

$= -\dfrac{\mathbb{E}_\tau[\tau]}{2}\left[\mu^2(\lambda_0+N) - 2\mu\left(\lambda_0\mu_0 + \sum_n x_n\right) + \left(\lambda_0\mu_0^2 + \sum_n x_n^2\right)\right] + \text{const}$

Let $a = (\lambda_0 + N)$, $b = 2(\lambda_0\mu_0 + \sum_n x_n)$ and $c = (\lambda_0\mu_0^2 + \sum_n x_n^2)$, we further let $d = \dfrac{b}{2a}$ and $e = c - \dfrac{b^2}{4a^2}$, we get: $\ln q_\mu^*(\mu) = -\dfrac{\mathbb{E}_\tau[\tau]}{2}\left[a(\mu+d)^2 + e\right] + \text{const}$, where e can be put into the constant terms since it is independent of μ.

Thus, we get: $\ln q_\mu^*(\mu) = -\dfrac{(\lambda_0+N)\mathbb{E}_\tau[\tau]}{2}\left[\left(\mu - \dfrac{\lambda_0\mu_0+\sum_n x_n}{\lambda_0+N}\right)^2\right] + \text{const}$. Taking exponential on both sides and normalize, we have: $\tilde{q}_\mu^*(\mu) = \exp\left(-\dfrac{(\lambda_0+N)\mathbb{E}_\tau[\tau]}{2}\left[\left(\mu - \dfrac{\lambda_0\mu_0+\sum_n x_n}{\lambda_0+N}\right)^2\right]\right)$, which is a Gaussian distribution, i.e., $q_\mu^*(\mu) = \mathcal{N}\left(\mu \;\middle|\; \dfrac{\lambda_0\mu_0+\sum_n x_n}{\lambda_0+N}, ((\lambda_0+N)\mathbb{E}_\tau[\tau])^{-1}\right)$.

ii)Expected mean: $\mathbb{E}_{q(\mu)}[\mu] = \int q(\mu)\,\mu\,d\mu$. Expected precision: $\mathbb{E}_{q(\tau)}[\tau] = \int q(\tau)\,\tau\,d\tau$

iii)Maximum log-likelihood: $\arg\max_\mu \ln p(D \mid \mu, \tau) = \arg\max_\mu \left\{\ln\left(\left(\frac{\tau}{2\pi}\right)^{N/2}\exp\left(-\frac{\tau}{2}\sum_n(x_n-\mu)^2\right)\right)\right\} = \arg\max_\mu\left\{-\frac{\tau}{2}\sum_n(x_n-\mu)^2\right\}$

Taking partial derivatives w.r.t. μ and equate to 0, we get: $\tau(\sum_n x_n - \mu N) = 0 \Rightarrow \mu = \dfrac{\sum_n x_n}{N}$

$\arg\max_\tau\left\{\ln\left(\left(\frac{\tau}{2\pi}\right)^{N/2}\exp\left(-\frac{\tau}{2}\sum_n(x_n-\mu)^2\right)\right)\right\} = \arg\max_\tau\left(\frac{N}{2}\ln\left(\frac{\tau}{2\pi}\right) - \frac{\tau}{2}\sum_n(x_n-\mu)^2\right)$

Taking partial derivatives w.r.t. τ and equate to 0, we get:

$\left(\frac{N}{2}\right)\left(\frac{2\pi}{\tau}\right)\left(\frac{1}{2\pi}\right) - \frac{1}{2}\sum_n(x_n-\mu)^2 = 0 \Rightarrow \tau = \dfrac{N}{\sum_n(x_n-\mu)^2}$

iv)Solution obtained from variational inference is an approximation and often under-estimate. However, maximum log-likelihood ignores prior and would not be accurate if number of observations is low.

EXAMPLE 4: VARIANCE INFERENCE

Figure 13.1 shows a three-node undirected graphical model, where $X_i \in \mathbb{R}_{\geq 0}$, $\psi(X_1,X_2) = \exp\{-\alpha X_1 X_2\}$, $\psi(X_i) = \exp\{-\beta X_i\}$, and $\alpha = 0.5$ and $\beta = 2.5$ are constants.

(a) Using variational inference, find the expressions of the expectation of X_1, X_2, and X_3 under $q(X_1)$, $q(X_2)$, and $q(X_3)$, respectively, where

$$q(X_1,X_2,X_3) = q(X_1)q(X_2)q(X_3)$$

is the mean-field approximation of the posterior distribution $p(X_1,X_2,X_3)$.

(b) Taking the initial expected values of X_2 and X_3 under $q(X_2)$ and $q(X_3)$ to be 2.0 and 1.0, respectively, find the mean-field approximation $q(X_1,X_2,X_3)$ after one iteration.

1. $\int \exp\{kx\}\,dx = \dfrac{1}{k}\exp(kx) + \text{const}, \quad k \neq 0.$

2. $\int x\exp\{kx\}\,dx = \left(\dfrac{kx-1}{k^2}\right)\exp(kx), \quad k \neq 0.$

(a) Joint probability.

$p(\mathbf{X}) = \frac{1}{Z}\prod_{i<j}\psi(X_i,X_j)\prod_k\psi(X_k) = \frac{1}{Z}\exp\left[-\alpha(X_1X_2 + X_1X_3 + X_2X_3) - \beta(X_1+X_2+X_3)\right]$.

(b) Mean-field update of X_1. $\ln q^*(X_1) = \mathbb{E}_{q_2q_3}[\ln p] = -X_1\alpha(\mu_2+\mu_3) - \beta X_1 + c = -\lambda_1 X_1 + c$, where $\lambda_1 = \alpha(\mu_2+\mu_3) + \beta, \qquad \mu_1 = \mathbb{E}[X_1] = \frac{1}{\lambda_1}$.

Numerical update: Parameters: $\alpha = 0.5$, $\beta = 0.25$. Initial: $\mu_2 = 0.2$, $\mu_3 = 0.1$.

1. Update X_1: $\lambda_1 = 0.5(0.2) + 0.5(0.1) + 0.25 = 0.4, \qquad \mu_1 = \frac{1}{0.4} = 2.5$.

2. Update X_2 (use new μ_1): $\lambda_2 = \alpha\mu_1 + \alpha\mu_3 + \beta = 0.5(2.5) + 0.5(0.1) + 0.25 = 1.55, \qquad \mu_2 = \frac{1}{1.55} \approx 0.64$.

3. Update X_3 (use new μ_1, μ_2): $\lambda_3 = \alpha\mu_1 + \alpha\mu_2 + \beta = 0.5(0.25) + 0.5(0.64) + 0.25 = 0.695$.

$q(\mathbf{X}) = q^*(X_1)\,q^*(X_2)\,q^*(X_3) \propto \exp\left[-0.4X_1 - 1.55X_2 - 0.695X_3\right]$.

EXAMPLE 5: HMM

a. Figure 8.1 shows a homogeneous hidden Markov Model (HMM) over three time steps. The latent random variables are Y_1, Y_2, Y_3, where $Y_n \in \{0,1,2\}$, and the observed random variables are X_1, X_2, X_3, where $X_n \in \mathbb{R}$. The prior probability of the random variable Y_1 is $p(Y_1 \mid \pi) = \prod_k \pi_k^{Y_1^k}$, where $\pi = \{0.2, 0.5, 0.3\}$. Furthermore, the transition probability is given by:

$$p(Y_n \mid Y_{n-1}, A) = \prod_k\prod_j A_{jk}^{Y_{n-1}^j Y_n^k}, \quad A = \begin{bmatrix} 0.2 & \alpha & \beta \\ 0.1 & 0.6 & 0.3 \\ 0.4 & 0.5 & 0.1 \end{bmatrix},$$

and the emission probabilities of the respective observed random variables X_n are:

Given that the minimum probability of the joint distribution $p(Y_1,Y_2,Y_3,X_1,X_2,X_3)$ is 0.000216 and occurs at $Y_1 = 0, Y_2 = 1, Y_3 = 0$, find the unknown values α and β in the transition probability.

b. Figure 8.2 shows an undirected graphic model with six random variables $X_1, X_2, X_3, X_4, X_5, X_6$, where $X_i \in \{0,1,2\}$. The potential $\psi(X_i, X_j)$ between any pair of nodes X_i and X_j, where $i < j$, is given in Table 2.2. Given $X_1 = 0$, $X_3 = 1$ and $X_5 = 2$, find the states of X_2, X_4, X_6 that maximize the joint distribution $p(X_1,X_2,X_3,X_4,X_5,X_6)$.

Answer: a. Joint probability: $p(X,Y) = p(Y_1)\prod_{n=2}p(Y_n \mid Y_{n-1})\prod_{n=1}p(X_n \mid Y_n)$

$\min_Y p(X,Y) = \min_Y p(Y_1)\prod_{n=2}p(Y_n \mid Y_{n-1})\prod_{n=1}p(X_n \mid Y_n)$

$= \min_{Y_1}\min_{Y_2}\min_{Y_3} p(Y_1)p(Y_2 \mid Y_1)p(Y_3 \mid Y_2)p(X_1 \mid Y_1)p(X_2 \mid Y_2)p(X_3 \mid Y_3)$

$= \min_{Y_3} p(X_3 \mid Y_3)\min_{Y_2} p(X_2 \mid Y_2)p(Y_3 \mid Y_2)\min_{Y_1} p(Y_1)p(X_1 \mid Y_1)p(Y_2 \mid Y_1)$

Given that the minimum probability equals 0.000216 and occurs at $Y_1 = 0, Y_2 = 1, Y_3 = 0$, this imply:

$\min_Y p(X,Y) = 0.000216$

$p(Y_1=0)p(Y_2=1 \mid Y_1=0)p(Y_3=0 \mid Y_2=1)p(X_1 \mid Y_1=0)p(X_2 \mid Y_2=1)p(X_3 \mid Y_3=0)$

$= (0.2)(\alpha)(0.1)(0.3)(0.4)(0.3) = 0.00072\alpha = 0.000216 \Rightarrow \alpha = 0.3$

Since each row of the transition matrix sums to one, we have $0.2 + \alpha + \beta = 1 \Rightarrow \beta = 0.5$.

b. Joint probability: $p(X) = \frac{1}{Z}\psi(X_1=0,X_2)\psi(X_1=0,X_4)\psi(X_2,X_3=1)\psi(X_2,X_5=2)\psi(X_3=1,X_6)\psi(X_4,X_5=2)\psi(X_5=2,X_6)$

$\max_{X_2}\max_{X_4}\max_{X_6} p(X) = \max_{X_2}\psi(X_1=0,X_2)\psi(X_2,X_3=1)\psi(X_2,X_5=2)\max_{X_4}\psi(X_1=0,X_4)\psi(X_4,X_5=2)\max_{X_6}\psi(X_3=1,X_6)\psi(X_5=2,X_6)$

$\max_{X_6}\psi(X_3=1,X_6)\psi(X_5=2,X_6) = \max[(2)(3),(4)(6),(8)(9)] = \max[6,24,72] = 72 \quad (X_6=2)$

$\max_{X_4}\psi(X_1=0,X_4)\psi(X_4,X_5=2) = \max[(1)(7),(5)(8),(7)(9)] = \max[7,40,63] = 63 \quad (X_4=2)$

$\max_{X_2}\psi(X_1=0,X_2)\psi(X_2,X_3=1)\psi(X_2,X_5=2) = \max[(1)(5)(7),(5)(4)(8),(7)(6)(9)] = \max[35,160,378] = 378 \quad (X_2=2)$

EXAMPLE 6: SAMPLING

The Bayesian network shown in Figure 7.1 has five random variables X_1, X_2, X_3, X_4, X_5, where $x_i \in \{0,1,2\}$ for $i = 1,2$ and $x_i \in \{0,1\}$ for $i = 3,4,5$. The conditional probability tables are:

X_1	X_2	$p(x_2 \mid x_1)$
0	0	0.1
1	0	0.4
2	0	0.8
0	1	0.2
1	1	0.4
2	1	0.1

X_3	X_4	$p(x_4 \mid x_3)$
0	0	0.2
1	0	0.5
2	0	0.7

X_1	$p(x_1)$
0	0.1
1	0.4

X_2	X_3	$p(x_3 \mid x_2)$
0	0	0.3
1	0	0.4
2	0	0.1

X_3	X_4	X_5	$p(x_5 \mid x_3, x_4)$
0	0	0	0.6
0	1	0	0.3
1	0	0	0.1
1	1	0	0.8

(a) Given the following numbers drawn from a uniform distribution $u \sim \mathcal{U}(0,1)$:

$$u = [\,0.4387 \quad 0.4898 \quad 0.7513 \quad 0.4984 \quad 0.2760\,],$$

generate one set of samples from the joint distribution $p(x_1,x_2,x_3,x_4,x_5)$ using Gibbs sampling. Use $x_1 = 0, x_2 = 0, x_3 = 0, x_4 = 0, x_5 = 0$ as the initialization. Show all your workings clearly.

(a) Conditional Expressions:

$p(x_1 \mid x_2,x_3,x_4,x_5) = \dfrac{p(x_1,x_2,x_3,x_4,x_5)}{p(x_2,x_3,x_4,x_5)} = \dfrac{p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1)p(x_4 \mid x_2)p(x_5 \mid x_3,x_4)}{p(x_4 \mid x_2)p(x_5 \mid x_3,x_4)\sum_{x_1}p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1)} \propto$

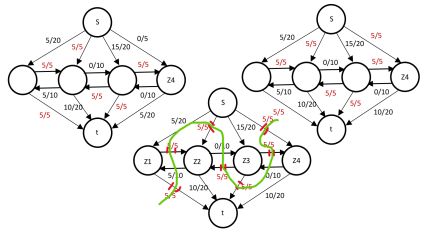
$p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1)$

$p(x_1 = 0 \mid \mathbf{x}_{-1}) \propto (0.1)(0.1)(0.3) = 0.003, p(x_1 = 1 \mid \mathbf{x}_{-1}) \propto (0.4)(0.4)(0.4) = 0.064, p(x_1 = 2 \mid \mathbf{x}_{-1}) \propto (0.5)(0.8)(0.1) = 0.04$

$p(x_1 = 0 \mid \ldots) = \frac{0.003}{0.107} = 0.028 p(x_1 = 1 \mid \ldots) = \frac{0.064}{0.107} = 0.598 p(x_1 = 2 \mid \ldots) = \frac{0.04}{0.107} = 0.374$

Sampling: Given $u = 0.4387$: CDF(0) = 0.028, CDF(1) = 0.028 + 0.598 = 0.626

Since $0.028 < u < 0.626 \implies \mathbf{x_1} = \mathbf{1}$.



EXAMPLE 8: PARAMETER LEARNING

Figure 6.1 shows a Markov Random Field (MRF) with two random variables X_1 and X_2, where $x_i \in \{0,1\}$. Furthermore, let $\phi_1(x_1)$ and $\phi_2(x_2)$ denote the unary potentials, and $\psi_{12}(x_1,x_2)$ denote the pairwise potential. Given the observations over 14 trials as shown in Table 6.1, find the unknown value of $\psi_{12}(x_1=0,x_2=0)$ in the potential tables shown in Table 6.2. Show all your workings clearly.

Answer: Joint probability:

$p(x_1,x_2) = \prod_n \frac{1}{Z_p}\phi_1(x_{1,n})\phi_2(x_{2,n})\psi_{12}(x_{1,n},x_{2,n})$, where $Z_p = \sum_{x_1}\sum_{x_2}\phi_1(x_1)\phi_2(x_2)\psi_{12}(x_1,x_2)$.

Log form: $\ln p = \sum_n \ln\phi_1(x_{1,n}) + \sum_n \ln\phi_2(x_{2,n}) + \sum_n \ln\psi_{12}(x_{1,n},x_{2,n}) - N\ln Z_p$.

Note that: $\sum_n \ln\phi(x_{1,n}) = \sum_{x_1} N(x_1)\ln\phi(x_1)$.

$\ln p = \sum_{x_1} N(x_1)\ln\phi_1(x_1) + \sum_{x_2} N(x_2)\ln\phi_2(x_2) + \sum_{x_1}\sum_{x_2} N(x_1,x_2)\ln\psi_{12}(x_1,x_2) - N\ln Z_p$.

To find $\arg\max_{\psi_{12}} \ln p$: $\dfrac{\partial \ln p}{\partial\psi_{12}(0,0)} = \dfrac{N(0,0)}{\psi_{12}(0,0)} - \dfrac{N}{Z_p}\dfrac{\partial Z_p}{\partial\psi_{12}(0,0)} = 0$. $\dfrac{\partial Z_p}{\partial\psi_{12}(0,0)} = \phi_1(0)\phi_2(0)$.

Thus: $\dfrac{N(0,0)}{\psi_{12}(0,0)} = \dfrac{N}{Z_p}\phi_1(0)\phi_2(0)$.

$k = Z_p = \phi_1(0)\phi_2(0)\psi_{12}(0,0) + \phi_1(0)\phi_2(1)\psi_{12}(0,1) + \phi_1(1)\phi_2(0)\psi_{12}(1,0) + \phi_1(1)\phi_2(1)\psi_{12}(1,1)$.

Given counts: $N(0,0) = 5, N(0,1) = 1, N(1,0) = 4, N(1,1) = 4$.

Solve: $\dfrac{5}{\psi_{12}(0,0)} - \dfrac{14}{k}(2)(1) = 0 \Rightarrow \dfrac{5}{\psi_{12}} = \dfrac{28}{k}, k = 2\psi_{12} + 10$.

Thus: $\dfrac{5}{\psi_{12}} = \dfrac{28}{2\psi_{12}+10}$ $5(2\psi_{12}+10) = 28\psi_{12}$ $10\psi_{12} + 50 = 28\psi_{12}$ $18\psi_{12} = 50$ $\psi_{12}(0,0) = \dfrac{25}{9}$.

EXAMPLE 9: SAMPLING

Consider the simple 3-node graph shown in Fig. 5.1 in which the observed node X is given by a Gaussian distribution $\mathcal{N}(x \mid \mu, \tau^{-1})$ with mean μ and precision τ. Suppose that the marginal distributions over the mean and precision are given by $\mathcal{N}(\mu \mid \mu_0, s_0)$ and $\text{Gam}(\tau \mid a, b)$, where $\text{Gam}(\cdot,\cdot)$ denotes a gamma distribution. Write down expressions for the conditional distributions $p(\mu \mid x,\tau)$ and $p(\tau \mid x,\mu)$ that would be required to apply Gibbs sampling to the posterior distribution $p(\mu,\tau \mid x)$.

Answer:

$p(\mu,\tau \mid x) = \dfrac{p(\mu,x,\tau)}{\int p(\mu,x,\tau)\,d\mu} = \dfrac{p(\mu)p(\tau)p(x \mid \mu,\tau)}{\int p(\mu)p(\tau)p(x \mid \mu,\tau)\,d\mu} = \dfrac{p(\mu)p(x \mid \mu,\tau)}{\int p(\mu)p(x \mid \mu,\tau)\,d\mu}$.

$p(x \mid \mu,\tau) = C_x e^{-0.5\tau(x-\mu)^2}$, $p(\mu \mid \mu_0, s_0) = C_\mu e^{-0.5s_0(\mu-\mu_0)^2}$, so $p(\mu)p(x \mid \mu,\tau) = C_xC_\mu e^{-0.5[\mu^2(\tau+s_0) - 2\mu(\tau x - s_0\mu_0) + (x^2 + s_0\mu_0^2)]}$.

$p(\mu \mid x,\tau) = \dfrac{\exp\{-0.5[\mu^2(\tau+s_0) - 2\mu(\tau x - s_0\mu_0)]\}}{\int\exp\{-0.5[\mu^2(\tau+s_0) - 2\mu(\tau x - s_0\mu_0)]\}d\mu}$.

Let $\alpha = \tau + s_0$, $\beta = \tau x - s_0\mu_0$. Completing the square gives exponent $-a\mu^2 + b\mu$ with $a = \alpha/2, b = \beta$.

Using $\int_{-\infty}^{\infty} e^{-a\mu^2 + b\mu}d\mu = \sqrt{\pi/a}e^{b^2/(4a)}$, we obtain $p(\mu \mid x,\tau) = \sqrt{\alpha/(2\pi)}\exp[-(\alpha/2)\mu^2 + \beta\mu + \beta^2/(2\alpha)]$, i.e. a Gaussian.

Next, $p(\tau \mid x,\mu) = \dfrac{p(\mu)p(\tau)p(x \mid \mu,\tau)}{\int p(\mu)p(\tau)p(x \mid \mu,\tau)d\tau} = \dfrac{p(\tau)p(x \mid \mu,\tau)}{\int p(\tau)p(x \mid \mu,\tau)d\tau}$.

$p(x \mid \mu,\tau) = C_x e^{-0.5\tau(x-\mu)^2}$, $p(\tau \mid a_0,b_0) = C_\tau \tau^{a_0-1}e^{-b_0\tau}$, so $p(\tau)p(x \mid \mu,\tau) = C_xC_\tau\tau^{a_0-1}e^{-\tau[0.5(x-\mu)^2 + b_0]}$.

Thus $p(\tau \mid x,\mu) = \dfrac{\tau^{a_0-1}e^{-\alpha\tau}}{\int_0^\infty \tau^{a_0-1}e^{-\alpha\tau}d\tau}$, where $\alpha = 0.5(x-\mu)^2 + b_0$.

$\int_0^\infty \tau^n e^{-\alpha\tau}d\tau = \Gamma(n+1)/\alpha^{n+1}$, we get $p(\tau \mid x,\mu) = \dfrac{\alpha^{a_0}}{\Gamma(a_0)}\tau^{a_0-1}e^{-\alpha\tau}$, a Gamma distribution.