Probabilistic Graphical Models

Brown University CSCI 2950-P, Spring 2013

Prof. Erik Sudderth

Lecture 11:
Inference & Learning Overview, Gaussian Graphical Models

Some figures courtesy Michael Jordan's draft textbook,
An Introduction to Probabilistic Graphical Models

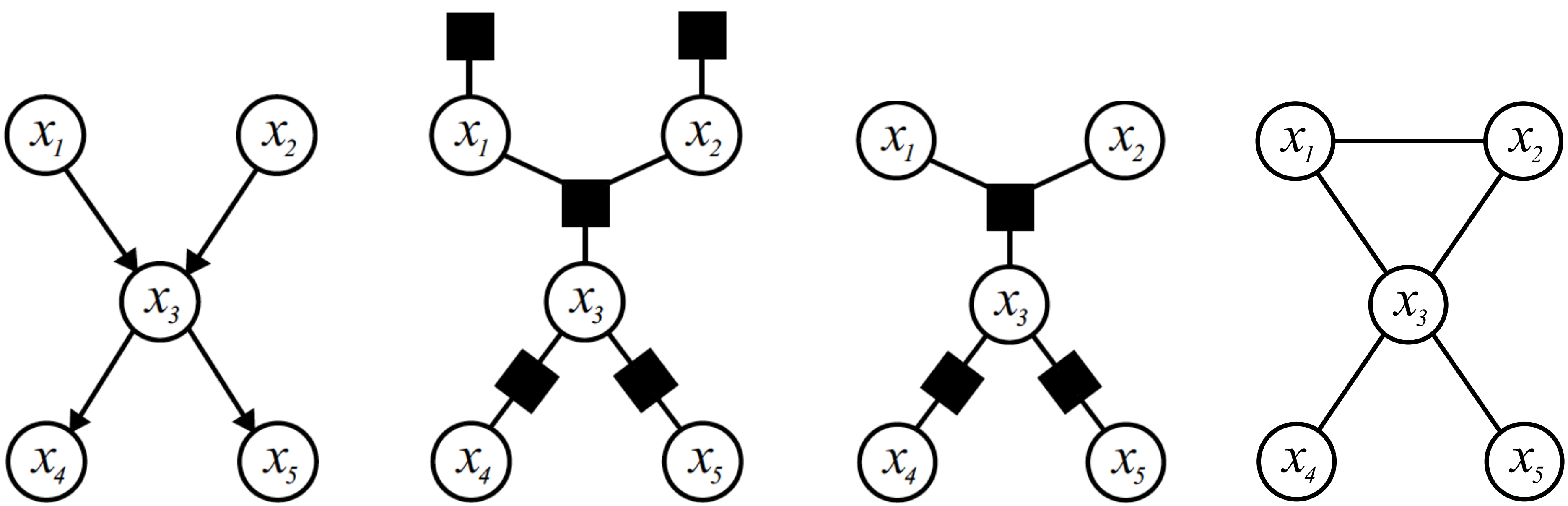
Graphical Models, Inference, Learning

Graphical Model: A factorized probability representation

- *Directed:* Sequential, causal structure for generative process
- *Undirected:* Associate features with edges, cliques, or factors

Inference: Given model, find marginals of hidden variables

- *Standardize:* Convert directed to equivalent undirected form
- *Sum-product BP:* Exact for any tree-structured graph
- *Junction tree:* Convert loopy graph to consistent clique tree



Undirected Inference Algorithms

	One Marginal	*All Marginals*
Tree	elimination applied recursively to leaves of tree	belief propagation or sum-product algorithm
Graph	elimination algorithm	junction tree algorithm: belief propagation on a junction tree

- A *junction tree* is a clique tree with special properties:
  - *Consistency:* Clique nodes corresponding to any variable from the original model form a connected subtree
  - *Construction:* Triangulations and elimination orderings

Graphical Models, Inference, Learning

Graphical Model: A factorized probability representation

- *Directed:* Sequential, causal structure for generative process
- *Undirected:* Associate features with edges, cliques, or factors

Inference: Given model, find marginals of hidden variables

- *Standardize:* Convert directed to equivalent undirected form
- *Sum-product BP:* Exact for any tree-structured graph
- *Junction tree:* Convert loopy graph to consistent clique tree

Learning: Given a set of *complete* observations of all variables

- *Directed:* Decomposes to independent learning problems: Predict the distribution of each child given its parents
- *Undirected:* Global normalization globally couples parameters: Gradients computable by inferring clique/factor marginals

Learning: Given a set of *partial* observations of some variables

- *E-Step:* Infer marginal distributions of hidden variables
- *M-Step:* Optimize parameters to match E-step and data stats

Learning for Undirected Models

- Undirected graph encodes dependencies within a single training example:

$$p(\mathcal{D} \mid \theta) = \prod_{n=1}^{N} \frac{1}{Z(\theta)} \prod_{f \in \mathcal{F}} \psi_f(x_{f,n} \mid \theta_f) \qquad \mathcal{D} = \{x_{\mathcal{V},1}, \ldots, x_{\mathcal{V},N}\}$$

- Given N independent, identically distributed, completely observed samples:

$$\log p(\mathcal{D} \mid \theta) = \left[\sum_{n=1}^{N} \sum_{f \in \mathcal{F}} \theta_f^T \phi_f(x_{f,n}) \right] - N A(\theta)$$

$$p(x \mid \theta) = \exp\left\{ \sum_{f \in \mathcal{F}} \theta_f^T \phi_f(x_f) - A(\theta) \right\}$$

$$\psi_f(x_f \mid \theta_f) = \exp\{\theta_f^T \phi_f(x_f)\} \qquad A(\theta) = \log Z(\theta)$$

Learning for Undirected Models

- Undirected graph encodes dependencies within a single training example:

$$p(\mathcal{D} \mid \theta) = \prod_{n=1}^{N} \frac{1}{Z(\theta)} \prod_{f \in \mathcal{F}} \psi_f(x_{f,n} \mid \theta_f) \qquad \mathcal{D} = \{x_{\mathcal{V},1}, \dots, x_{\mathcal{V},N}\}$$

- Given N independent, identically distributed, completely observed samples:

$$\log p(\mathcal{D} \mid \theta) = \left[\sum_{n=1}^{N} \sum_{f \in \mathcal{F}} \theta_f^T \phi_f(x_{f,n}) \right] - N A(\theta)$$

- Take gradient with respect to parameters for a single factor:

$$\nabla_{\theta_f} \log p(\mathcal{D} \mid \theta) = \left[\sum_{n=1}^{N} \phi_f(x_{f,n}) \right] - N \mathbb{E}_\theta[\phi_f(x_f)]$$

- Must be able to compute *marginal distributions* for factors in current model:
  - Tractable for tree-structured factor graphs via sum-product
  - For general graphs, use the junction tree algorithm to compute

Undirected Optimization Strategies

$$\log p(\mathcal{D} \mid \theta) = \left[\sum_{n=1}^{N} \sum_{f \in \mathcal{F}} \theta_f^T \phi_f(x_{f,n}) \right] - NA(\theta)$$

$$\nabla_{\theta_f} \log p(\mathcal{D} \mid \theta) = \left[\sum_{n=1}^{N} \phi_f(x_{f,n}) \right] - N\mathbb{E}_\theta[\phi_f(x_f)]$$

Gradient Ascent: Quasi-Newton methods like PCG, L-BGFS, …

- Gradients: Difference between statistics of observed data, and inferred statistics for the model at the current iteration
- Objective: Explicitly compute log-normalization (variant of BP)

Coordinate Ascent: Maximize objective with respect to the parameters of a single factor, keeping all other factors fixed

- Simple closed form depending on ratio between factor marginal for current model, and empirical marginal from data
- *Iterative proportional fitting* (IPF) and *generalized iterative scaling* algorithms

$$\psi_f^{(t+1)}(x_f) = \psi_f^{(t)}(x_f) \frac{\tilde{p}(x_f)}{p_f^{(t)}(x_f)}$$

Advanced Topics on the Horizon

Graph Structure Learning $\psi_f(x_f \mid \theta_f) = \exp\{\theta_f^T \phi_f(x_f)\}$

- Setting factor parameters to zero implicitly removes from model
- *Feature selection:* Search-based, sparsity-inducing priors, …
- *Topologies:* Tree-structured, directed, bounded treewidth, …

Approximate Inference: What if junction tree is intractable?

- Simulation-based (Monte Carlo) approximations
- Optimization-based (variational) approximations
- Inner loop of algorithms for approximate learning…

Alternative Objectives

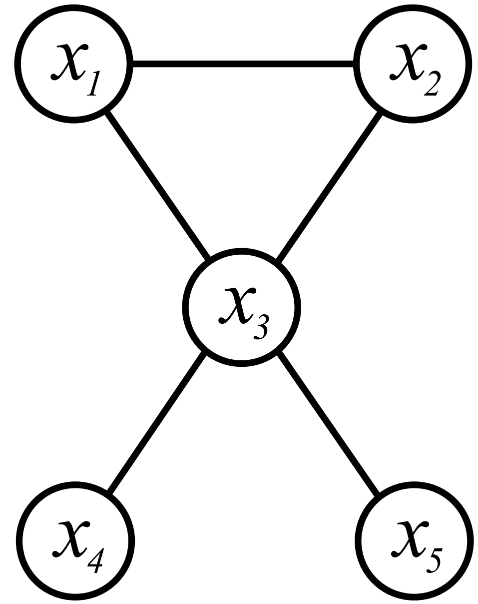
- Max-Product: Global MAP configuration of hidden variables
- Discriminative learning: CRF, max-margin Markov network,…

Inference with Continuous Variables

- *Gaussian:* Closed form mean and covariance recursions
- *Non-Gaussian:* Variational and Monte Carlo approximations…

Pairwise Markov Random Fields

$$p(x) = \frac{1}{Z} \prod_{(s,t)\in\mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s\in\mathcal{V}} \psi_s(x_s)$$



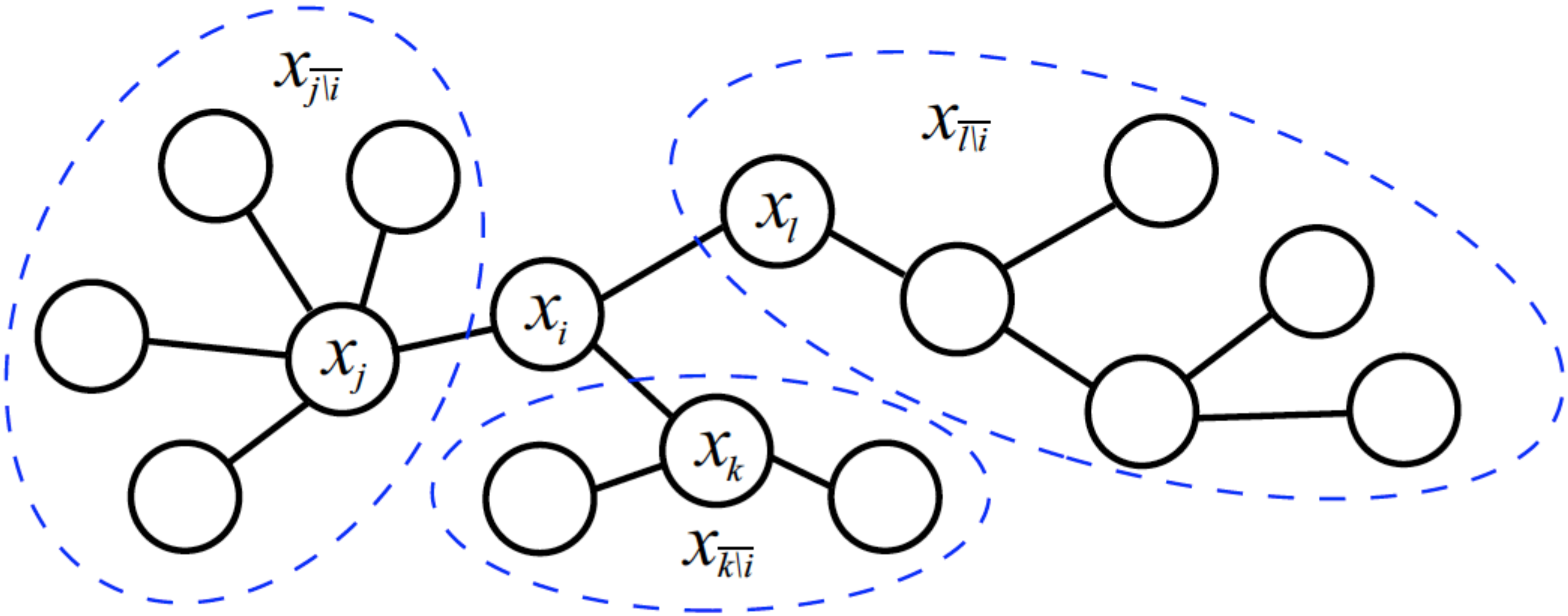
- Simple parameterization, but still expressive and widely used in practice
- Guaranteed Markov with respect to graph
- Any jointly Gaussian distribution can be represented by only *pairwise* potentials

$\mathcal{E} \longrightarrow$ set of undirected edges (s,t) linking pairs of nodes

$\mathcal{V} \longrightarrow$ set of N nodes or vertices, $\{1, 2, \ldots, N\}$

$Z \longrightarrow$ normalization constant (partition function)

Inference in Undirected Trees

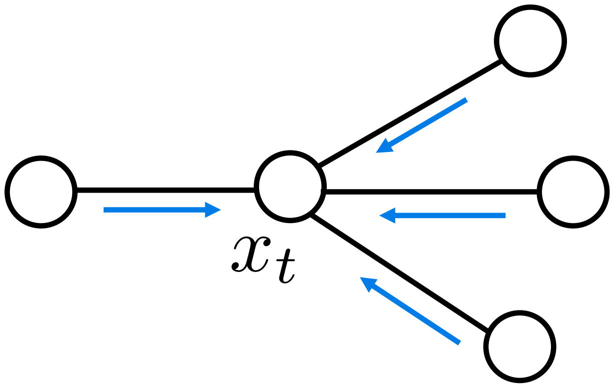


- For a tree, the maximal cliques are always pairs of nodes:

$$p(x) = \frac{1}{Z} \prod_{(s,t)\in\mathcal{E}} \psi_{st}(x_s, x_t) \prod_{s\in\mathcal{V}} \psi_s(x_s)$$

Belief Propagation (Integral-Product)

BELIEFS: Posterior marginals

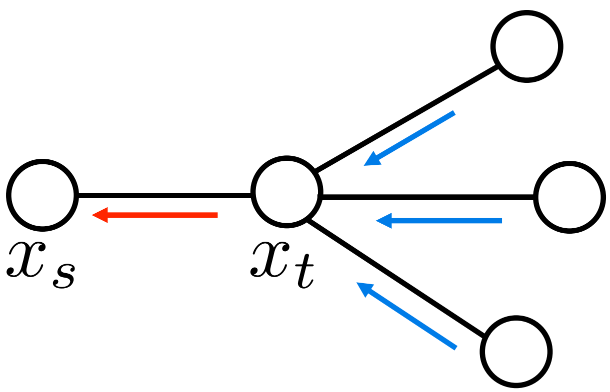


$$\hat{p}_t(x_t) \propto \psi_t(x_t) \prod_{u \in \Gamma(t)} m_{ut}(x_t)$$

$\Gamma(t) \longrightarrow$ neighborhood of node t (adjacent nodes)

MESSAGES: Sufficient statistics

$$m_{ts}(x_s) \propto \int_{x_t} \psi_{st}(x_s, x_t)\psi_t(x_t) \prod_{u \in \Gamma(t) \setminus s} m_{ut}(x_t)$$

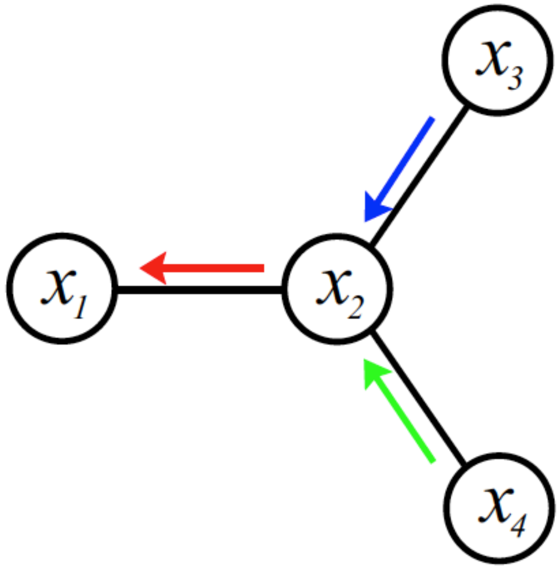


I) Message Product

II) Message Propagation

BP for Continuous Variables

Is there a finitely parameterized, closed form for the message and marginal functions?



Is there an analytic formula for the message integral, phrased as an update of these parameters?

$$p(x_1) \propto \iiint \psi_1(x_1)\psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)\, dx_4\, dx_3\, dx_2$$

$$\propto \psi_1(x_1) \iiint \psi_{12}(x_1,x_2)\psi_2(x_2)\psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)\, dx_4\, dx_3\, dx_2$$

$$\propto \psi_1(x_1) \int \psi_{12}(x_1,x_2)\psi_2(x_2)\left[\iint \psi_{23}(x_2,x_3)\psi_3(x_3)\psi_{24}(x_2,x_4)\psi_4(x_4)\, dx_4\, dx_3\right] dx_2$$

$$\propto \psi_1(x_1) \underbrace{\int \psi_{12}(x_1,x_2)\psi_2(x_2) \underbrace{\left[\int \psi_{23}(x_2,x_3)\psi_3(x_3)\, dx_3\right]}_{m_{32}(x_2)} \cdot \underbrace{\left[\int \psi_{24}(x_2,x_4)\psi_4(x_4)\, dx_4\right]}_{m_{42}(x_2)} dx_2}_{m_{21}(x_1) \propto \int \psi_{12}(x_1,x_2)\psi_2(x_2) m_{32}(x_2)\, m_{42}(x_2)\; dx_2}$$

Covariance and Correlation

Covariance:

$$\text{cov}[X,Y] \triangleq \mathbb{E}[(X-\mathbb{E}[X])(Y-\mathbb{E}[Y])] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$$

$$\text{cov}[\mathbf{x}] \triangleq \mathbb{E}\left[(\mathbf{x}-\mathbb{E}[\mathbf{x}])(\mathbf{x}-\mathbb{E}[\mathbf{x}])^T\right] = \begin{pmatrix} \text{var}[X_1] & \text{cov}[X_1,X_2] & \cdots & \text{cov}[X_1,X_d] \\ \text{cov}[X_2,X_1] & \text{var}[X_2] & \cdots & \text{cov}[X_2,X_d] \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}[X_d,X_1] & \text{cov}[X_d,X_2] & \cdots & \text{var}[X_d] \end{pmatrix}$$

$$\Sigma \in \mathbb{R}^{d\times d}$$

Always *positive semidefinite:* $u^T \Sigma u \geq 0 \text{ for any } u \in \mathbb{R}^{d\times 1}, u \neq 0$

Often *positive definite:* $u^T \Sigma u > 0 \text{ for any } u \in \mathbb{R}^{d\times 1}, u \neq 0$

Correlation:

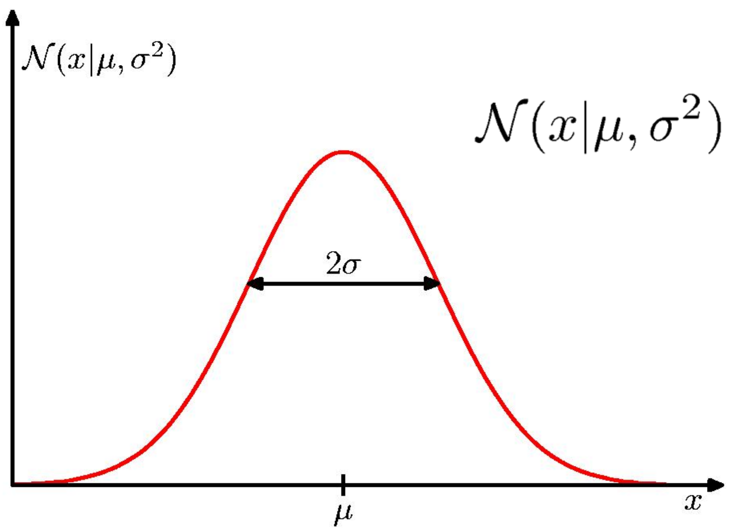
$$\text{corr}[X,Y] \triangleq \frac{\text{cov}[X,Y]}{\sqrt{\text{var}[X]\,\text{var}[Y]}} \qquad -1 \leq \text{corr}[X,Y] \leq 1$$

$$\mathbf{R} = \begin{pmatrix} \text{corr}[X_1,X_1] & \text{corr}[X_1,X_2] & \cdots & \text{corr}[X_1,X_d] \\ \vdots & \vdots & \ddots & \vdots \\ \text{corr}[X_d,X_1] & \text{corr}[X_d,X_2] & \cdots & \text{corr}[X_d,X_d] \end{pmatrix}$$

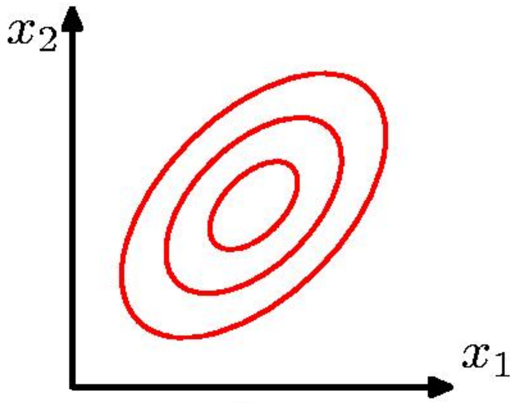
Independence:

$$p(X,Y) = p(X)p(Y) \Rightarrow \text{cov}[X,Y] = 0 \Leftrightarrow \text{corr}[X,Y] = 0$$

Gaussian Distributions



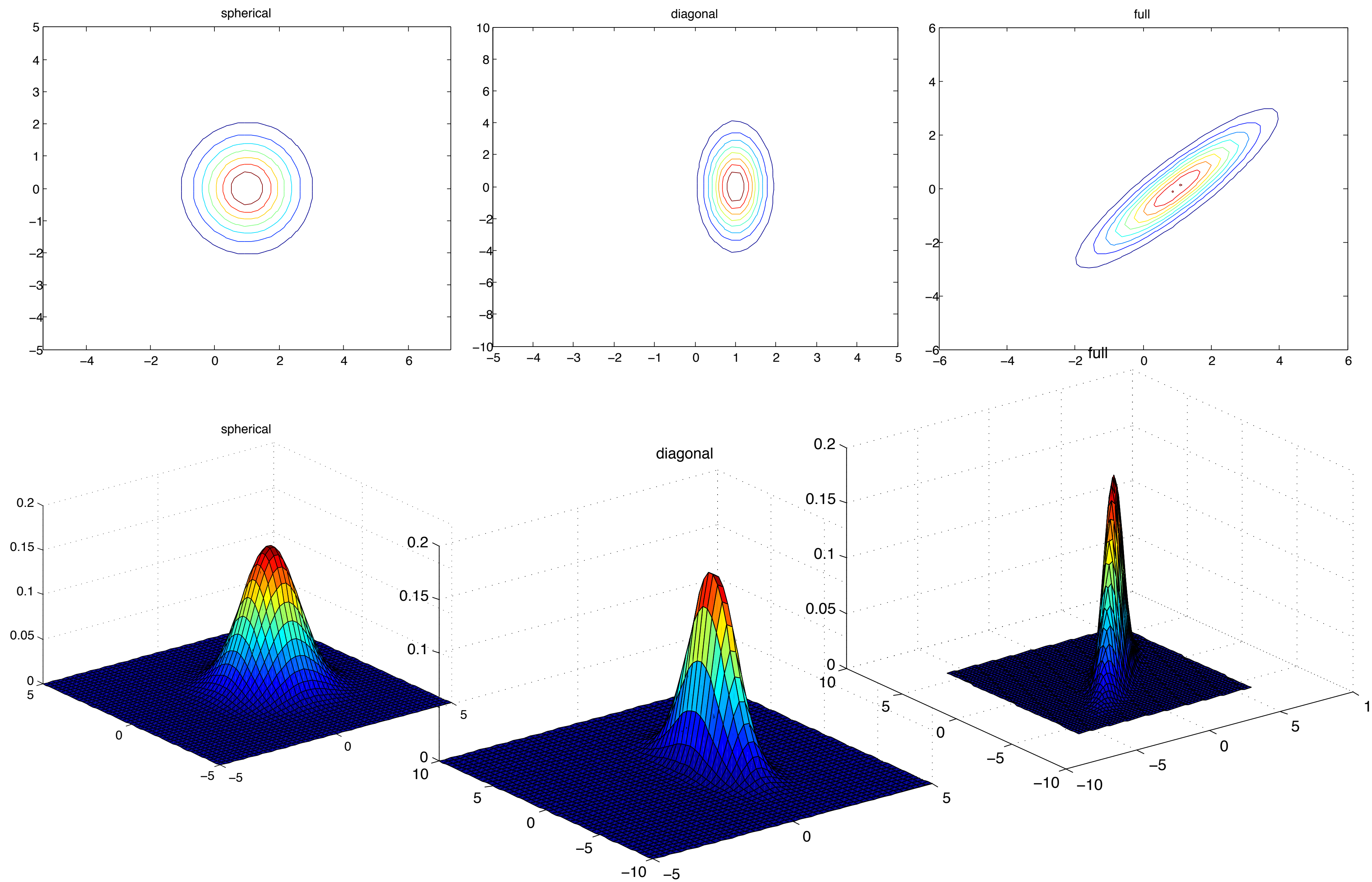
$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$



$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}$$

- Simplest joint distribution that can capture arbitrary mean & covariance
- Justifications from *central limit theorem* and *maximum entropy* criterion
- Probability density above assumes covariance is *positive definite*
- ML parameter estimates are *sample mean* & *sample covariance*

Two-Dimensional Gaussians



Gaussian Geometry

- Eigenvalues and eigenvectors:

$$\Sigma u_i = \lambda_i u_i, i = 1, \ldots, d$$

- For a *symmetric* matrix:

$$\lambda_i \in \mathbb{R} \qquad u_i^T u_i = 1 \qquad u_i^T u_j = 0$$

$$\Sigma = U \Lambda U^T = \sum_{i=1}^{d} \lambda_i u_i u_i^T$$

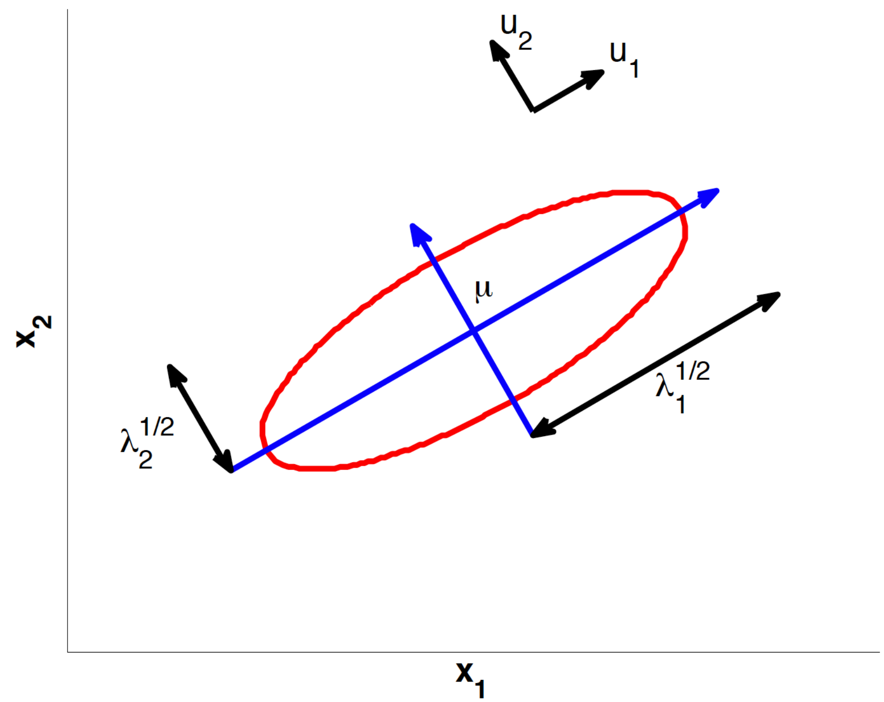
- For a *positive semidefinite* matrix:

$$\lambda_i \geq 0$$

- For a *positive definite* matrix:

$$\lambda_i > 0$$

$$\Sigma^{-1} = U \Lambda^{-1} U^T = \sum_{i=1}^{d} \frac{1}{\lambda_i} u_i u_i^T$$



$$\Delta^2 = (\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})$$

$$\Delta^2 = \sum_{i=1}^{D} \frac{y_i^2}{\lambda_i}$$

$$y_i = u_i^T (x - \mu)$$

Probabilistic PCA & Factor Analysis

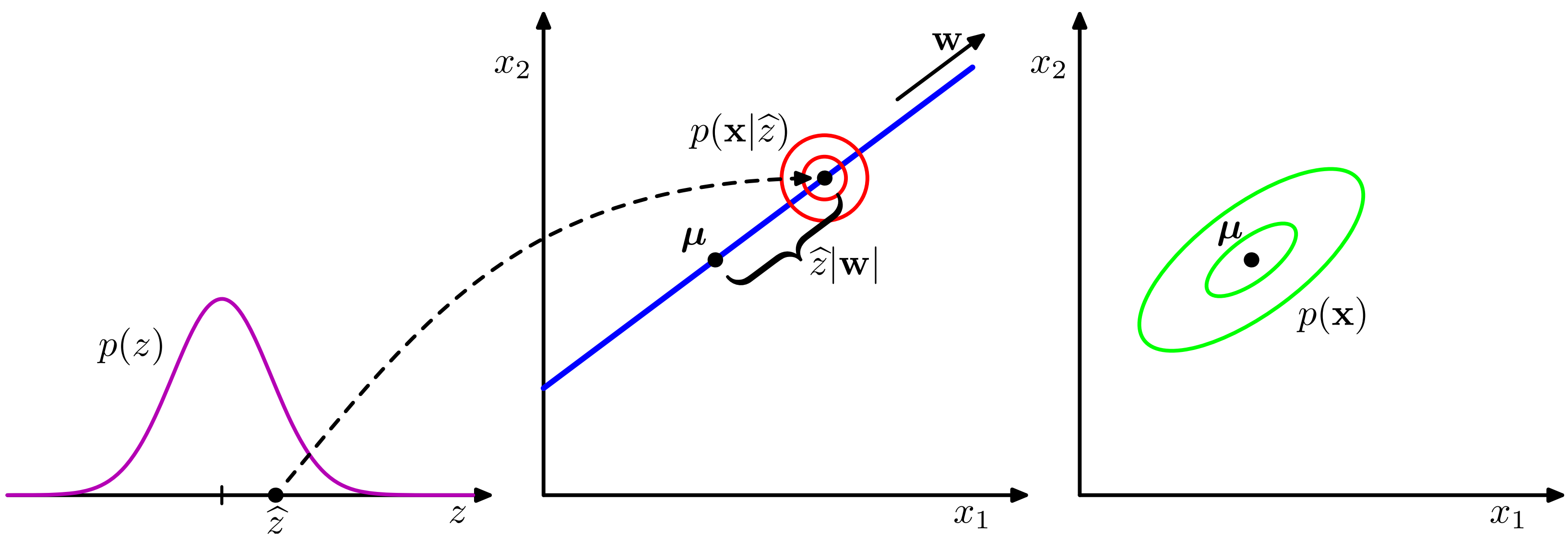
- **Both Models:** Data is a linear function of low-dimensional latent coordinates, plus Gaussian noise

$$p(x_i \mid z_i, \theta) = \mathcal{N}(x_i \mid W z_i + \mu, \Psi) \qquad p(z_i \mid \theta) = \mathcal{N}(z_i \mid 0, I)$$

$$p(x_i \mid \theta) = \mathcal{N}(x_i \mid \mu, WW^T + \Psi)$$

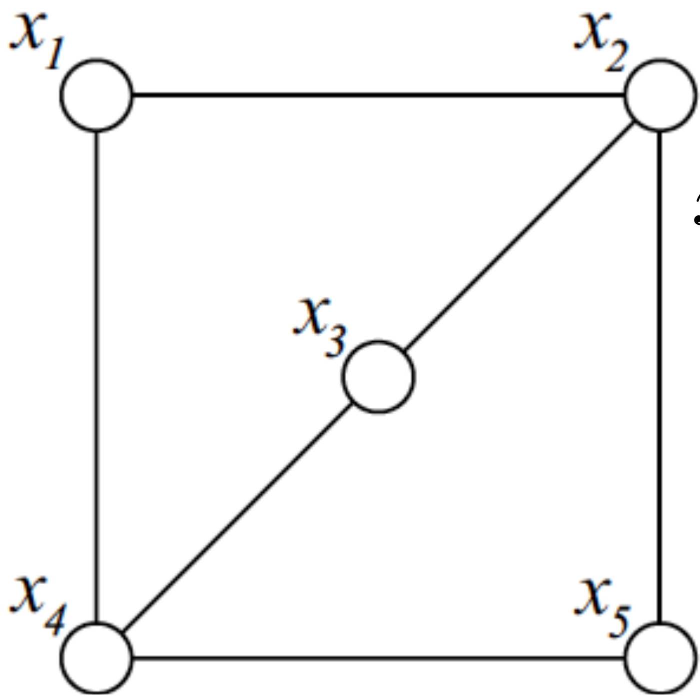
low rank covariance parameterization

- **Factor analysis:** Ψ is a general diagonal matrix
- **Probabilistic PCA:** $\Psi = \sigma^2 I$ is a multiple of identity matrix



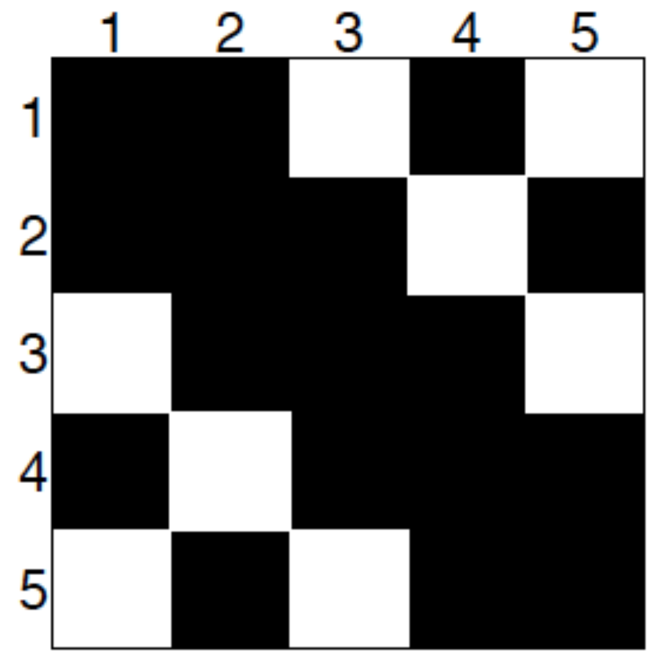
C. Bishop, Pattern Recognition & Machine Learning

Gaussian Graphical Models



$$x \sim \mathcal{N}(\mu, \Sigma)$$

$$J = \Sigma^{-1}$$



$$p(x) = \frac{1}{Z} \prod_{(s,t)\in\mathcal{E}} \psi_{s,t}(x_s, x_t)$$

$$\sum_{t\in N(s)} J_{s(t)} = J_{s,s}$$

$$\psi_{s,t}(x_s, x_t) = \exp\left\{ -\frac{1}{2} \begin{bmatrix} x_s^T & x_t^T \end{bmatrix} \begin{bmatrix} J_{s(t)} & J_{s,t} \\ J_{t,s} & J_{t(s)} \end{bmatrix} \begin{bmatrix} x_s \\ x_t \end{bmatrix} \right\}$$

Gaussian Potentials

$$p(x) = \frac{1}{Z}\exp\left\{-\frac{1}{2}x^T P^{-1} x\right\} = \frac{1}{Z}\prod_{s=1}^{N}\prod_{t=1}^{N}\exp\left\{-\frac{1}{2}x_s^T J_{s,t} x_t\right\} =$$

$$\frac{1}{Z}\prod_{(s,t)\in\mathcal{E}}\exp\left\{-\frac{1}{2}\begin{bmatrix}x_s^T & x_t^T\end{bmatrix}\begin{bmatrix}J_{s(t)} & J_{s,t}\\ J_{t,s} & J_{t(s)}\end{bmatrix}\begin{bmatrix}x_s\\ x_t\end{bmatrix}\right\} = \frac{1}{Z}\prod_{(s,t)\in\mathcal{E}}\psi_{s,t}(x_s, x_t)$$

$$Z = \left((2\pi)^N \det P\right)^{1/2}$$

$$p(x) = \frac{1}{Z}\prod_{(s,t)\in\mathcal{E}}\psi_{s,t}(x_s, x_t) \qquad \sum_{t\in N(s)} J_{s(t)} = J_{s,s}$$

$$\psi_{s,t}(x_s, x_t) = \exp\left\{-\frac{1}{2}\begin{bmatrix}x_s^T & x_t^T\end{bmatrix}\begin{bmatrix}J_{s(t)} & J_{s,t}\\ J_{t,s} & J_{t(s)}\end{bmatrix}\begin{bmatrix}x_s\\ x_t\end{bmatrix}\right\}$$