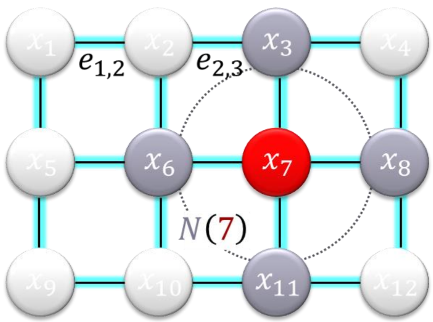
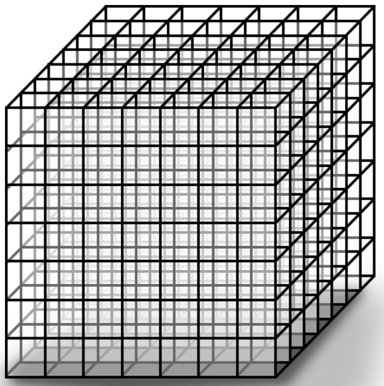
Machine Learning in Scientific Computing

CECAM/CSM/IRTG SCHOOL 2018





Max-Plank-Institut für Polymerforschung

Computational Science Mainz

Lecture 3.1.2

Markov Random Fields

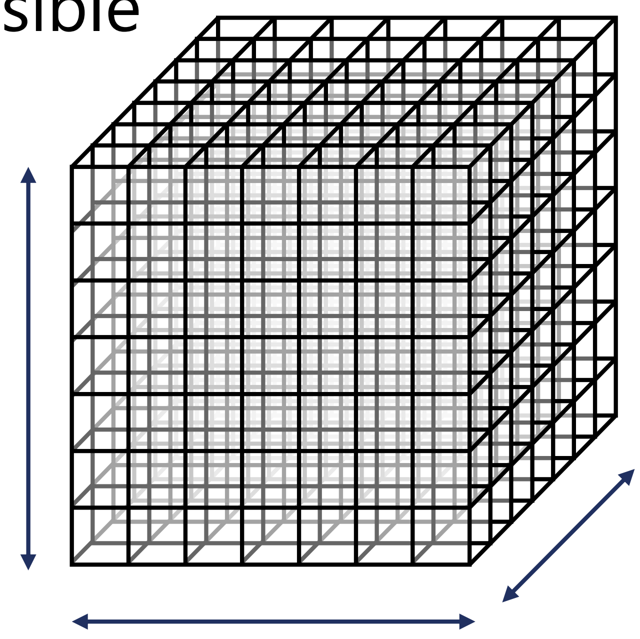
Michael Wand · Institut für Informatik · michael.wand@uni-mainz.de

Markov Random Fields and Graphical Models

Reducing dependencies

Problem:

- $p(x_1, x_2, \ldots, x_n)$ is to high-dimensional
- k States, n variables: $O(k^n)$ density entries
- General dependencies often infeasible



Reduction of Dependencies

- Parametric models (Gauss etc.)
- Markov Random Fields (MRFs)
- Deep Networks
- …

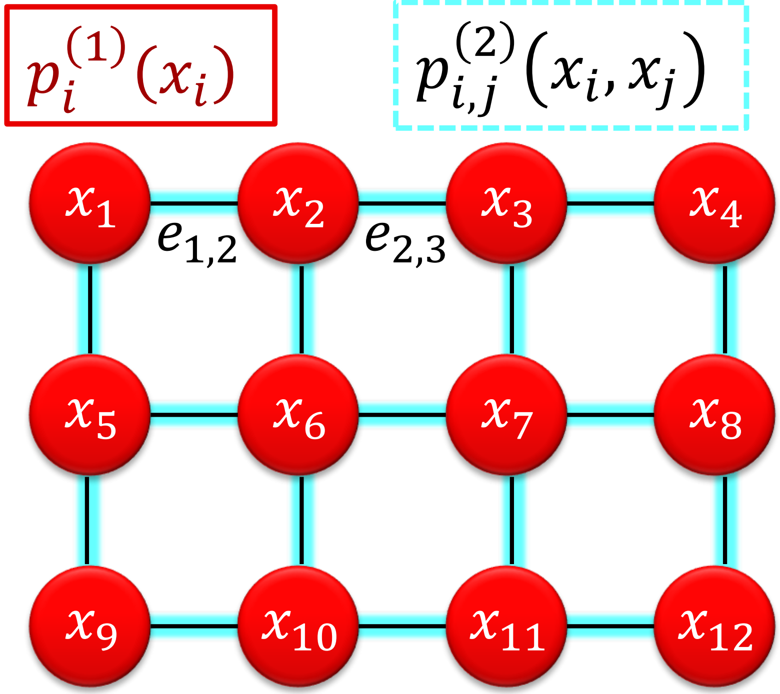
Graphical Models

Factorize Models

- Pairwise models:

$$p(x_1, \dots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} \underline{p_i^{(1)}(x_i)} \prod_{i,j \in E} \underline{p_{i,j}^{(2)}(x_i, x_j)}$$

- Model complexity:
  - $O(nk^2)$ parameters
- Higher order models:
  - Triplets, quadruples as factors
  - Local neighborhoods



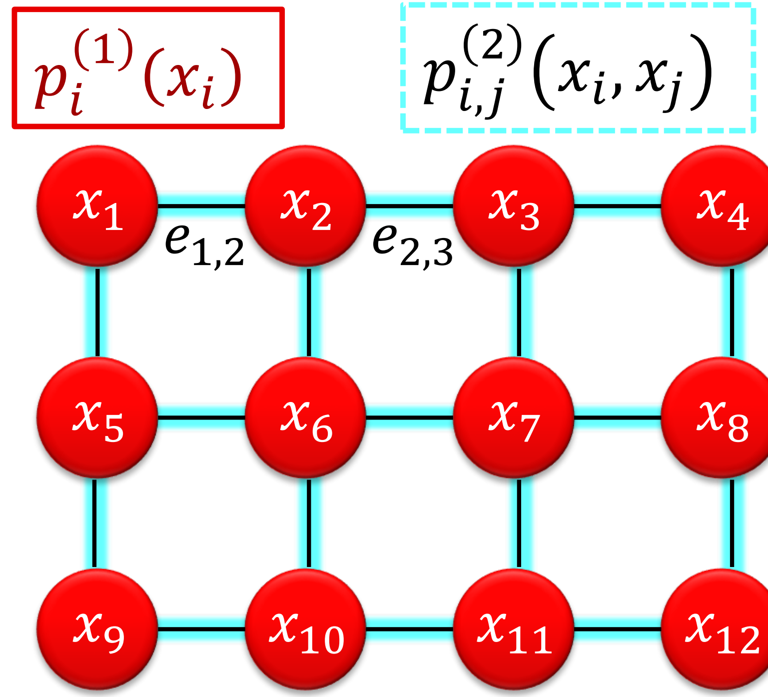
Graphical Models

Markov Random fields

- Factorize density in local “cliques”

Graphical model

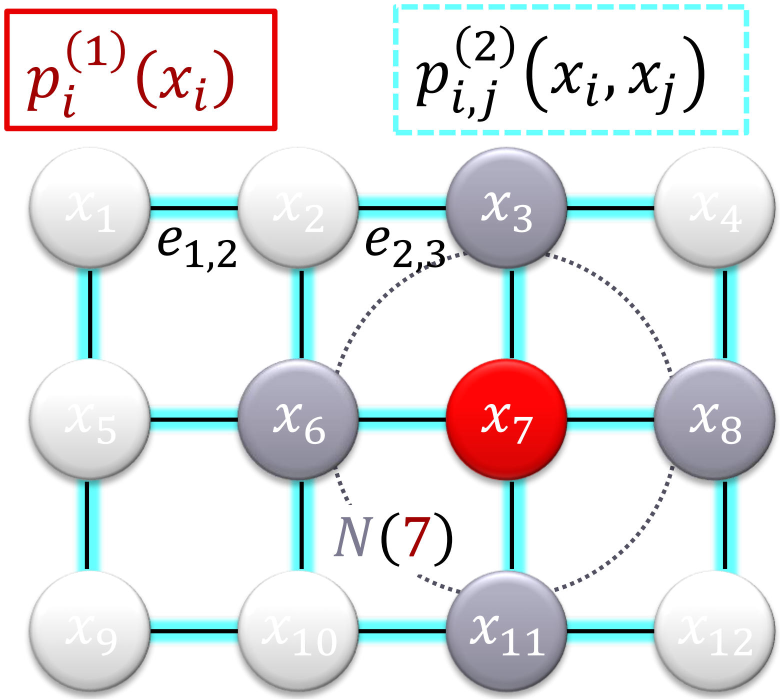
- Connect variables that are directly dependent
- Formal model: Conditional independence



Graphical Models

Conditional Independence

- A node is conditionally independent of all others given the values of its direct neighbors
- I.e. set these values to constants, x_7 is independent of all others



Formally

- $p(x_i | x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = p\big(x_i \big| \{x_j | j \in N(i)\}\big)$

Graphical Models

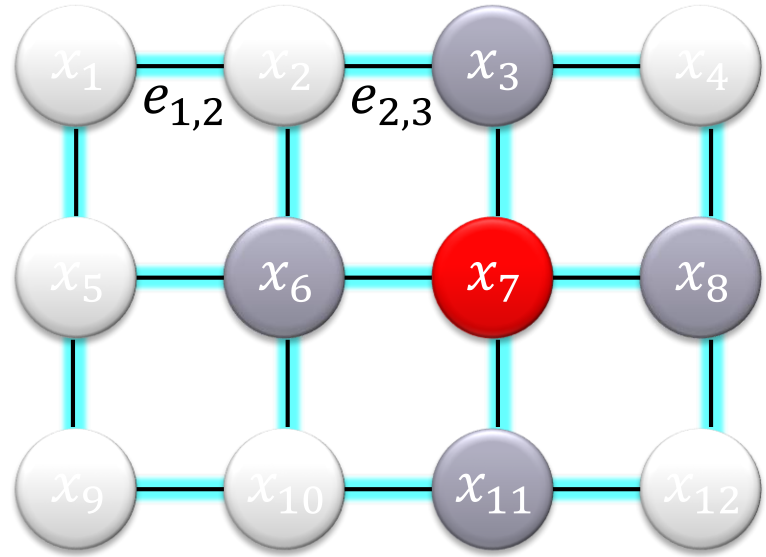
Theorem (Hammersley–Clifford):

- Assuming positive densities $p(x_i) > 0$

The therorem

- Given conditional independence as graph, density factors over cliques in the graph.
- And vice versa.

$p_i^{(1)}(x_i)$ $\quad$ $p_{i,j}^{(2)}(x_i, x_j)$

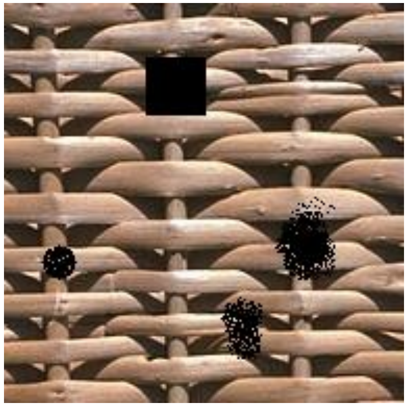


$$\Updownarrow$$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{i=1}^{n} p_i^{(1)}(x_i) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j)$$

Example: Texture Synthesis

original



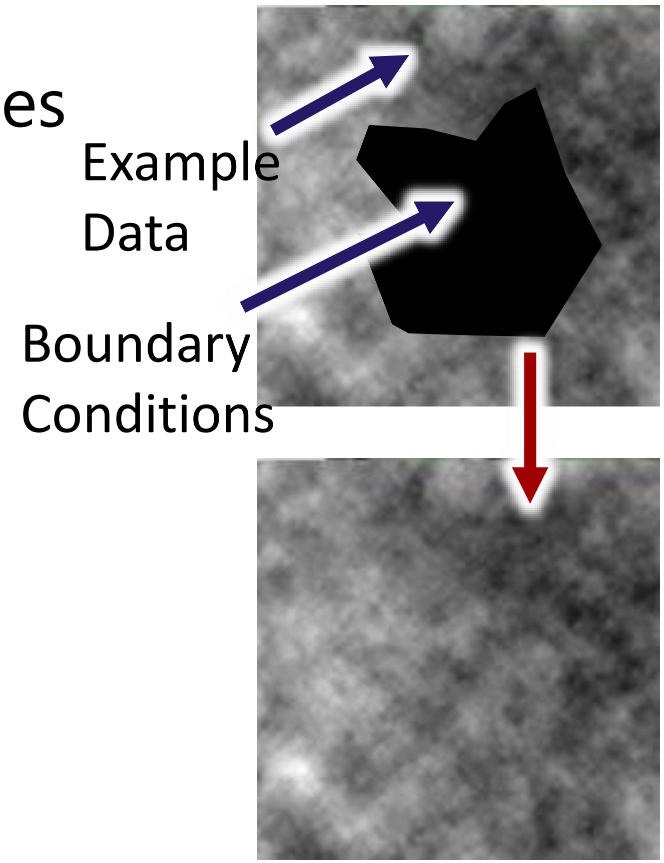
region selected

completion

Texture Synthesis

Idea

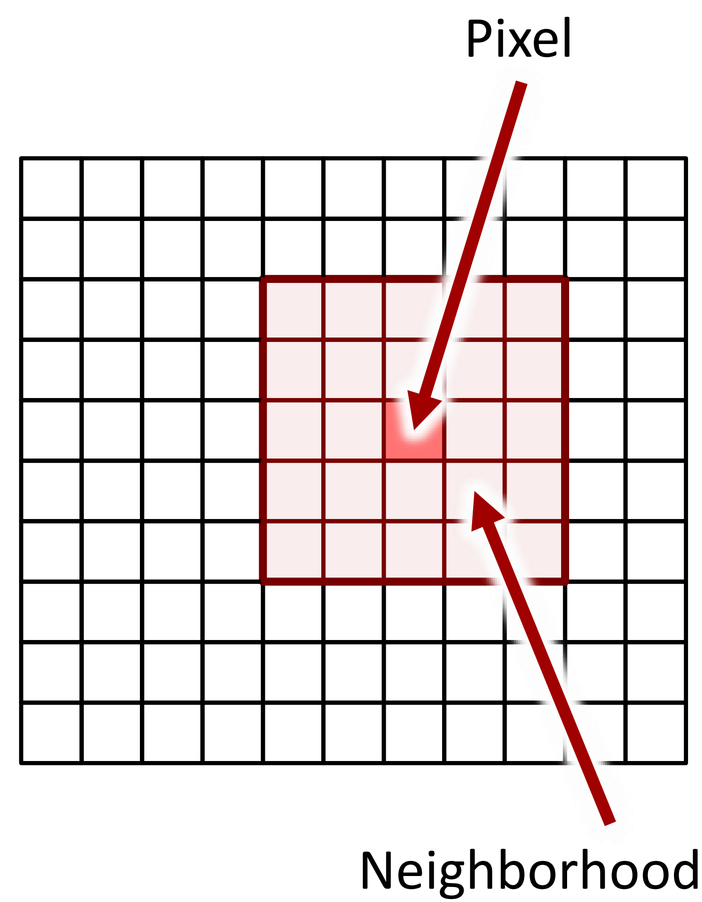
- One or more images as examples
- Learn image statistics
- Use knowledge:
  - Specify boundary conditions
  - Fill in texture



The Basic Idea

Markov Random Field Model

- Image statistics
- How pixels are colored depends on local neighborhood only (Markov Random Field)
- Predict color from neighborhood



A Little Bit of Theory…

Image statistics:

- An image of $n \times m$ pixels
- Random variable: $\mathbf{x} = [x_{11},\ldots,x_{nm}] \in [0, 1, \ldots, 255]^{n\times m}$
- Probability distribution:
  $p(\mathbf{x}) = p(x_{11}, \ldots, x_{nm})$

256 choices

⋮

256 choices

$256^{n\times m}$ probability values

Impossible to learn full images from examples!

Simplification

Problem:

- Statistical dependencies
- Simple modell can express dependencies on all kinds of combinations

Markov Random Field:

- Each pixel is *conditionally independent* of the rest of the image given a small neighborhood
- In English: likelihood only depends on neighborhood, not rest of the image

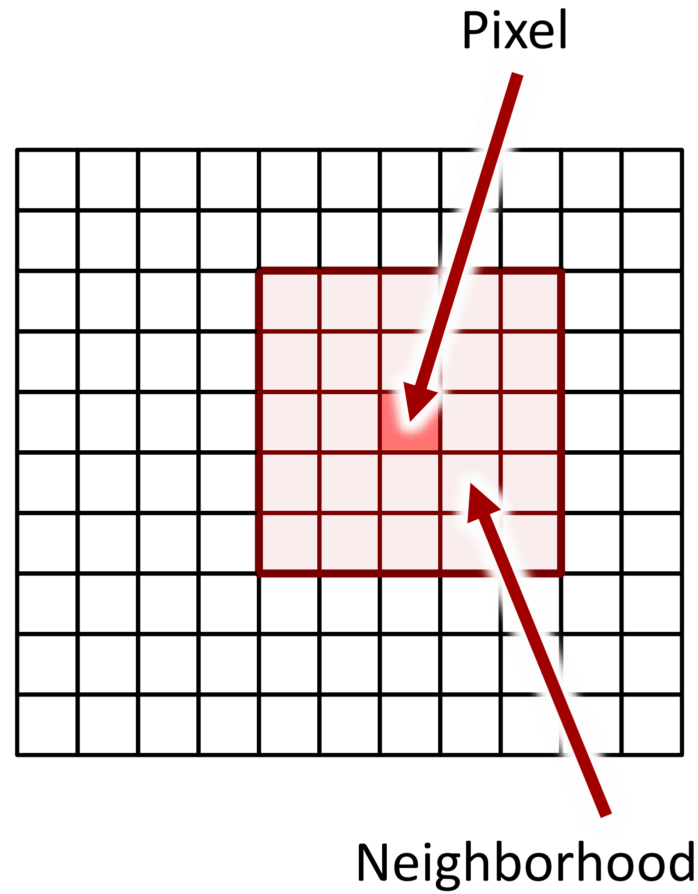
Markov Random Field

Example:

- Red pixel depends on light red region
- Not on black region
- If region is known, probability is fixed and independent of the rest

However:

- Regions overlap
- Indirect global dependency



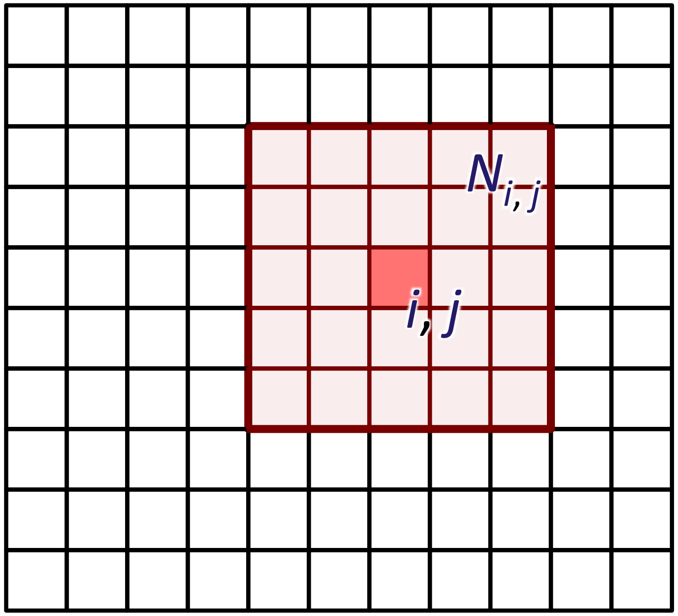
Texture Synthesis

Use for Texture Synthesis

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{i=1}^{n} \prod_{j=1}^{m} p_{i,j}(N_{i,j})$$

$$p_{i,j} = p_{i,j}(N_{i,j})$$

$$= p_{i,j}(x_{i-k,j-k} \dots, x_{i+k,j+k})$$



Inference

Inference Problem

- Computing $p(\mathbf{x})$ is trivial for known $\mathbf{x}$.
- Finding the $\mathbf{x}$ that maximizes $p(\mathbf{x})$ is very complicated.
- In general: NP-hard
- No efficient solution known (not even for images)

In practice

- Different approximation strategies
  ("heuristics", strict approximation is also NP-hard)

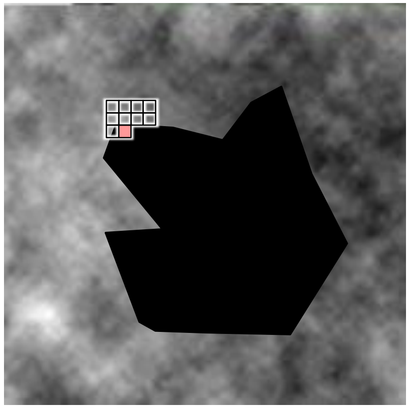
Simple Practical Algorithm

Here is the short story:

- Unknown pixels:
  consider known neighborhood
- Match to all of the known data
- Copy the pixel with the best matching neighborhood
- Region growing, outside in

Approximation only

- Can run into bad local minima



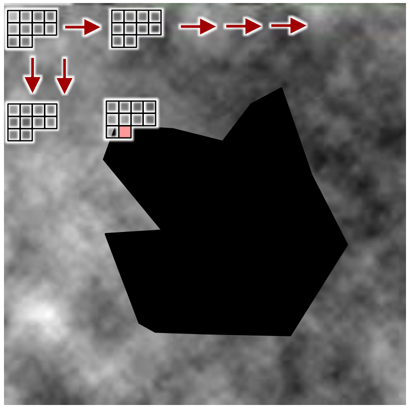


Image Analogies (Siggraph 2001)

CRF Image Segmentation

Reference (Image)

X. He, R.S. Zemel, M.A. Carreira-Perpinan:
Multiscale Conditional Random Fields for Image Labeling,
IEEE CVPR 2004.

Example: Weak Formulations of Differential Equations

Differential Equations

Example equation

$$\frac{d}{dt} f(t) = F(f(t), t)$$

Discretization

$$\frac{y_i - y_{i-1}}{h} = F(y_i, t_i)$$

Weak formulation (variational approach)

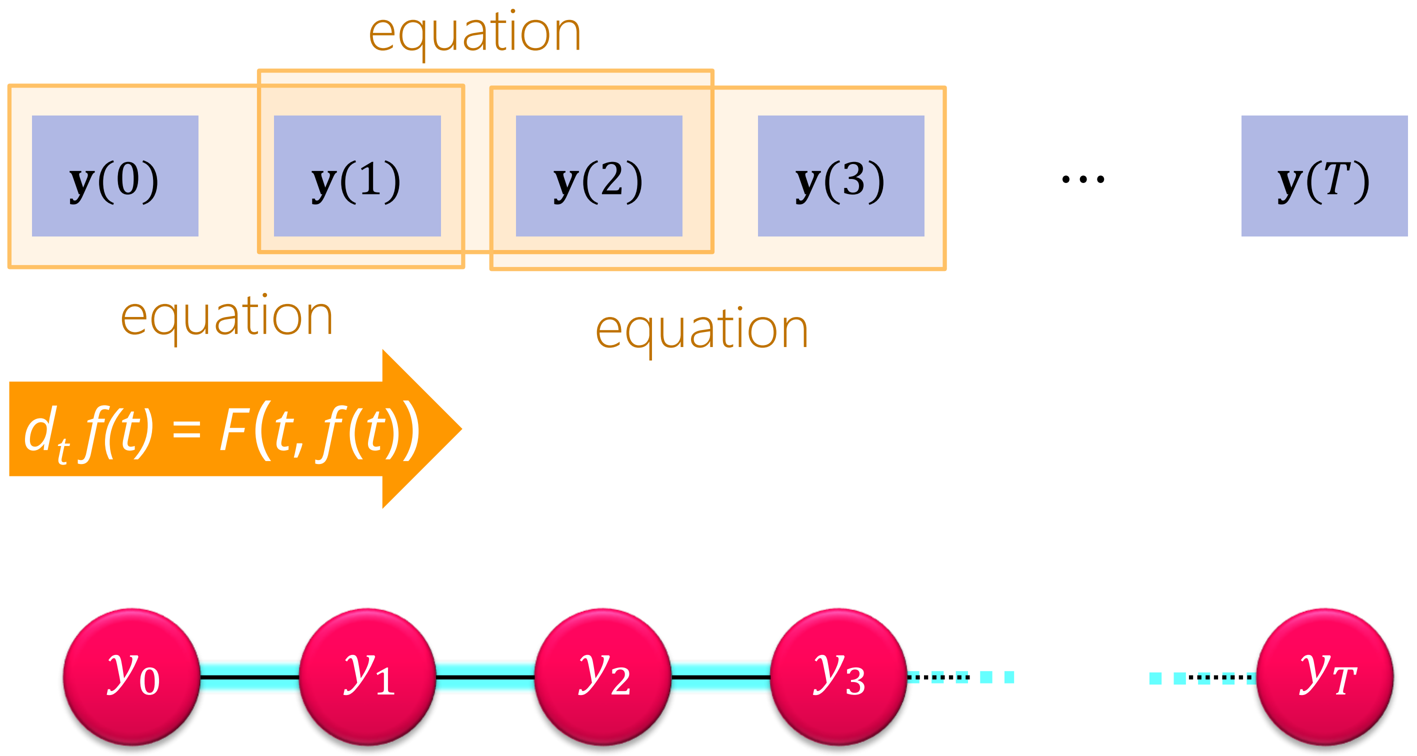
$$\left(\frac{y_i - y_{i-1}}{h} - F(y_i, t_i)\right)^2 \rightarrow min$$

$$\arg\max_{\mathbf{y}} \frac{1}{Z} \prod_{i=1}^{n-1} \exp\left(\frac{y_i - y_{i-1}}{h} - F(y_i, t_i)\right)^2$$

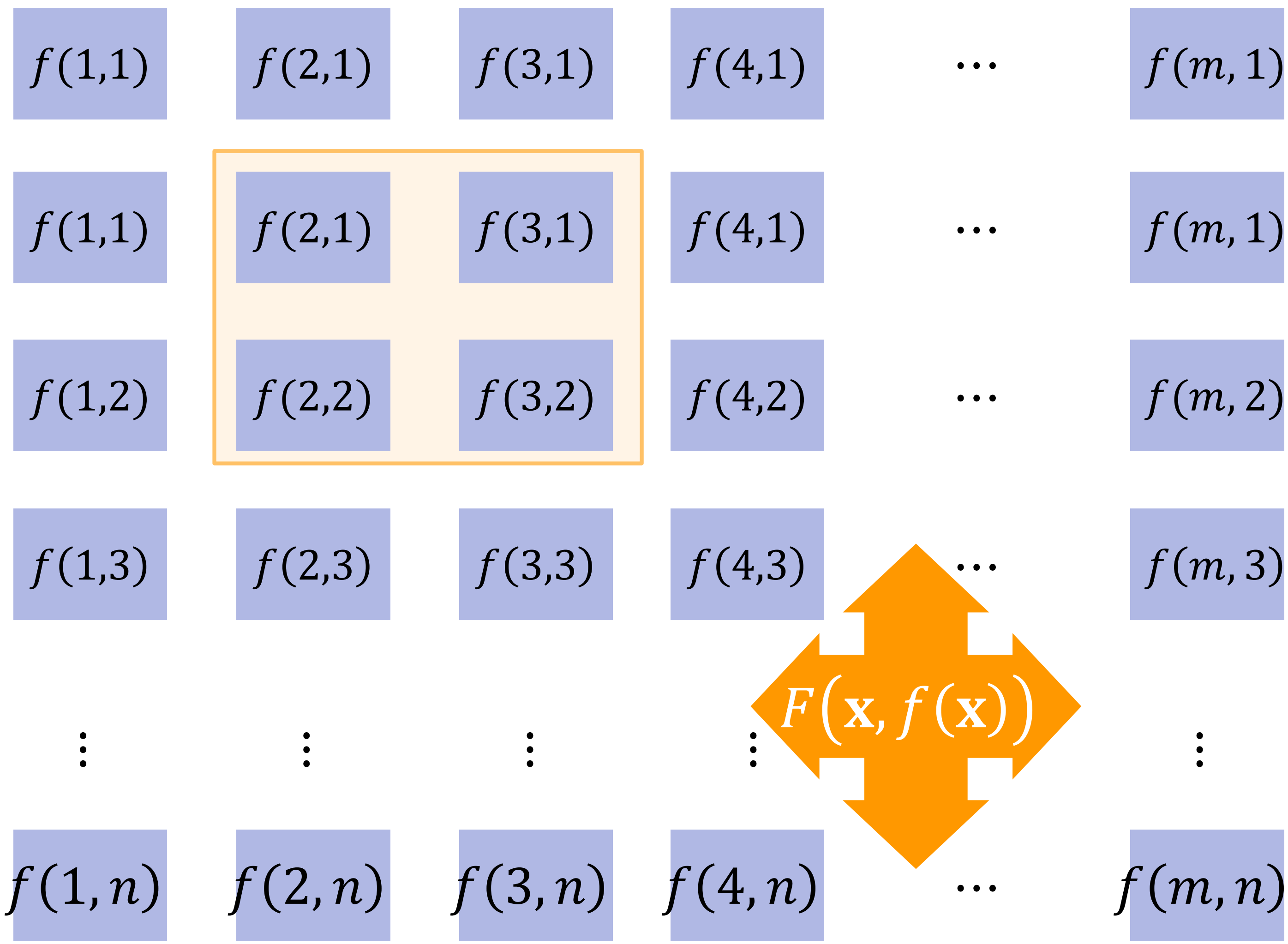
Gaussian MRF

Ordinary Differential Equations

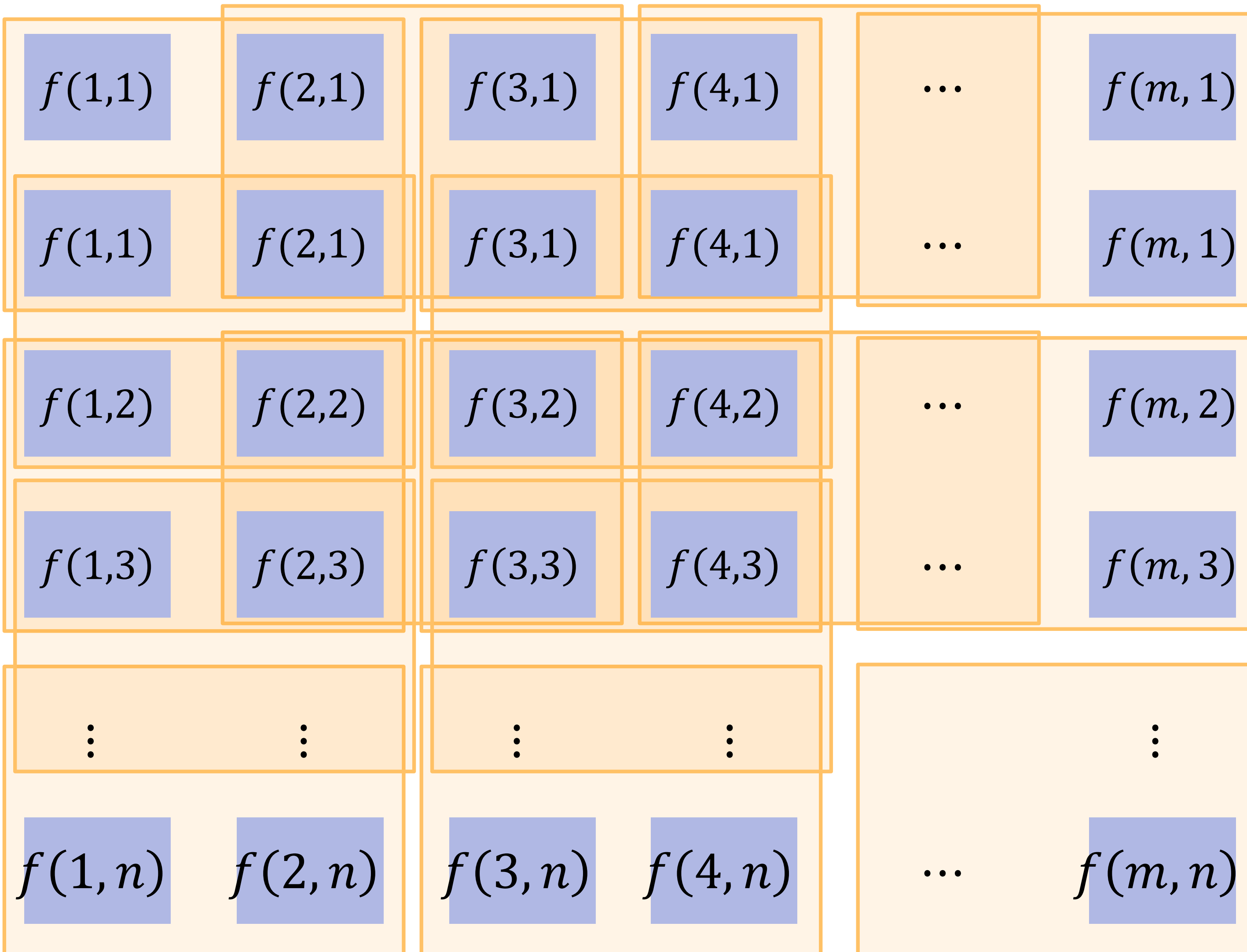
Causal Chain



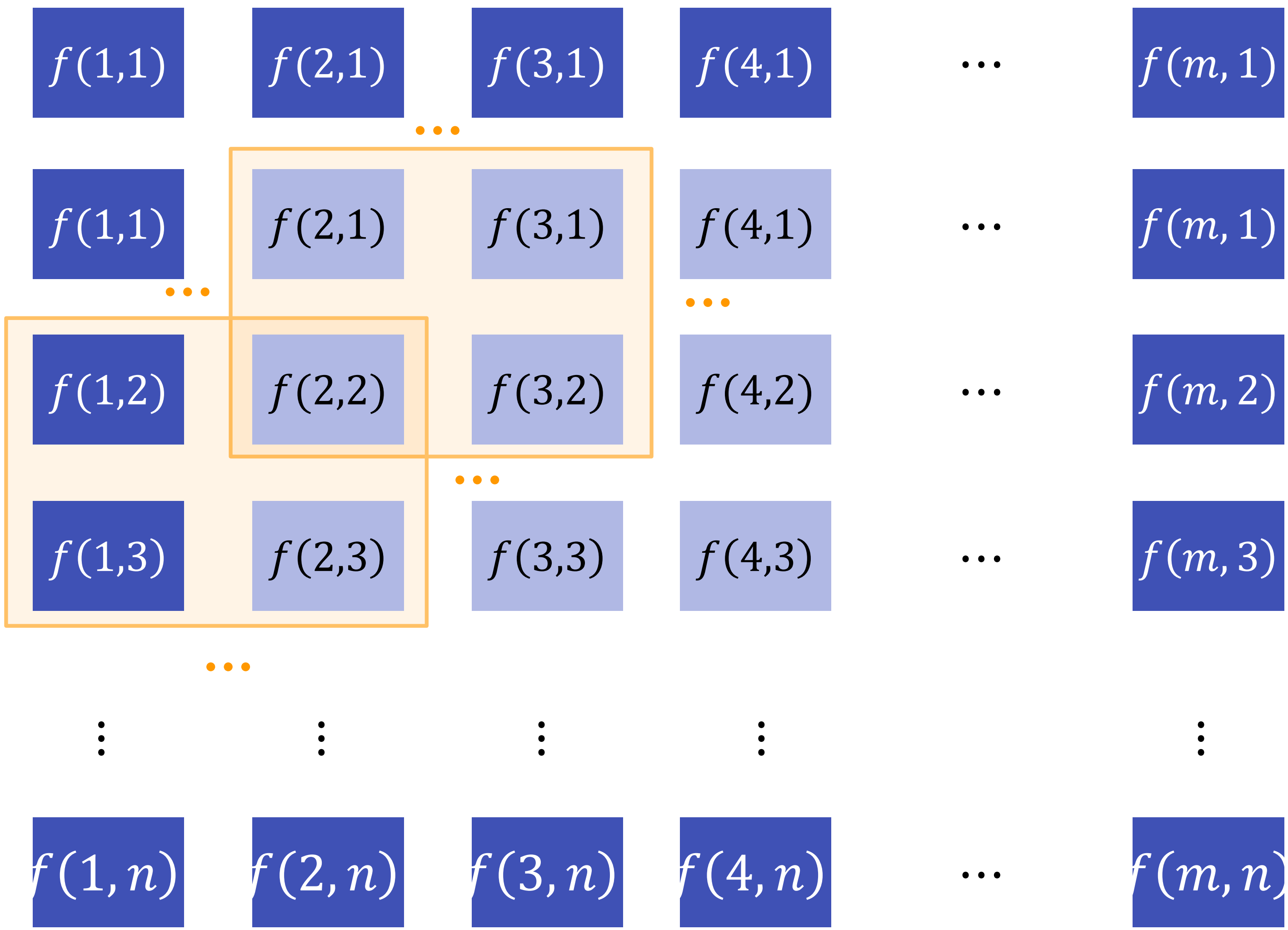
Structure of PDE



Structure of PDE



Boundary Value Problem



Inference in MRFs

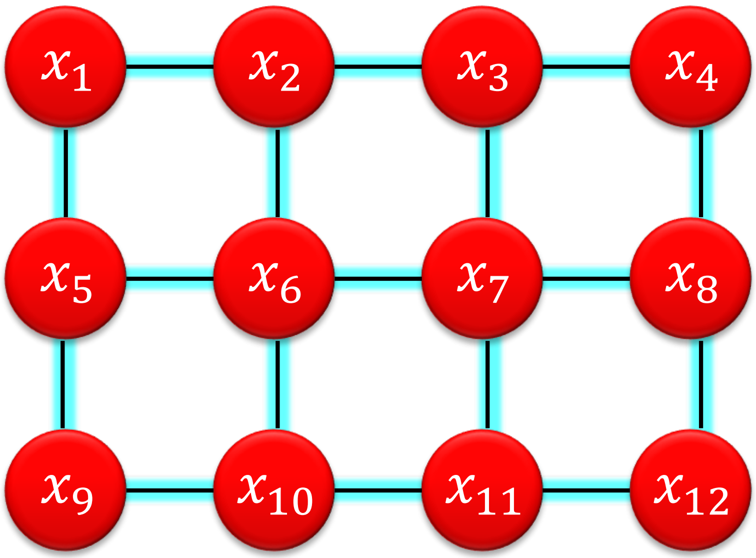
Inference

Model

$$p(x_1, \dots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} \underline{p_i^{(1)}(x_i)} \prod_{i,j \in E} \underline{p_{i,j}^{(2)}(x_i, x_j)}$$

Inference

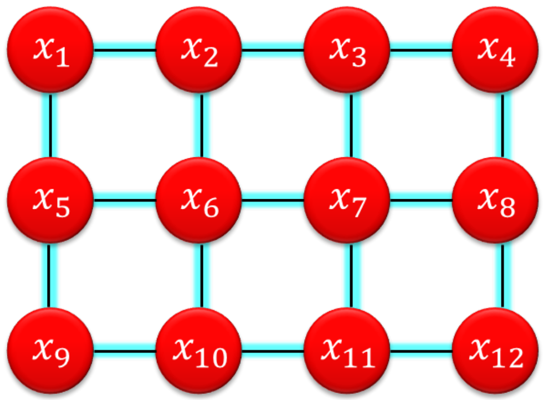
- Try all combinations of $(x_1, \dots, x_n)$
- Compute probability $p(x_1, \dots, x_n)$
- Determine maximum (or marginalize)



Inference

Complexity

- Brute-force-search / -integration:
  - Infeasible for large dimensions
  - Only toy-models
- Analytic maximization / integration
  - Special models only
  - Example: Gaussians
- Numerical maximization
  - All convex log-likelihoods
    - Gaussians (L_2-error)
    - L_1-errors
    - etc.



$$p(x_1, \dots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} p_i^{(1)}(x_i) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j)$$

Example:

Image Reconstruction

Image Reconstruction Model

Problem statement

- Measured 2D pixel image
- Distorted by noise
- Want to remove noise

Bayesian problem modeling

- Model of measurement process
- Prior distribution on images (this is Bayesian)

Inference: Maximum-a-posteriori

Model

Image

- $m_{i,j}$ with $i = 1 \ldots w, j = 1, \ldots, h$
- (continuous analogue: $f: [1, w] \times [1, h] \to \mathbb{R}$)

Probability space

- $\Omega = \mathbb{R}^{w \times h}$

Model

Bayes rule

$$P(M|D) \sim P(D|M) \cdot P(M)$$

Likelihood

- $P(D|M) = \prod_{i=1}^{w} \prod_{j=1}^{h} P(d_i|m_i)$ (i.i.d. noise)

$$= \prod_{i=1}^{w} \prod_{j=1}^{h} \mathcal{N}_{d_i,\sigma_D}(m_i) \text{ (Gaussian noise)}$$

$$= \prod_{i=1}^{w} \prod_{j=1}^{h} \left[\frac{1}{\sigma_D \sqrt{2\pi}} e^{-\frac{(m_i - d_i)^2}{2\sigma_D^2}} \right]$$

(Gaussian distribution)

Model

Likelihood

- $P(D|M) = \prod_{i=1}^{w} \prod_{j=1}^{h} \left[\frac{1}{\sigma_D \sqrt{2\pi}} e^{-\frac{(m_i - d_i)^2}{2\sigma_D^2}} \right]$

Neg-log-likelihood

$$E(D|M) \coloneqq -\ln P(D|M) = \sum_{i=1}^{w} \sum_{j=1}^{h} \frac{(m_i - d_i)^2}{2\sigma_D^2} + \cancel{\frac{wh}{\sigma_D \sqrt{2\pi}}}$$

independent of m_i

Model

Prior

- Assumption: Large image gradients are unlikely
- Gaussian distribution on Gradients
- Neg-log-likelihood: $\frac{1}{2\sigma^2}\|\nabla f\|^2$
- Discreet:

$$\mathrm{E}(M) \coloneqq -\ln P(M) = \sum_{i=1}^{w-1}\sum_{j=1}^{h-1}\frac{\left(m_{i+1,j}-m_{i,j}\right)^2+\left(m_{i,j+1}-m_{i,j}\right)^2}{2\sigma_M^2} + \cancel{\frac{wh}{\sigma_X\sqrt{2\pi}}}$$

independent of m_i

ground truth
reconstruction (Gaussian prior)
input (noisy)
reconstruction (Laplace prior)

Minimization Problem

Minimize

$$E(D|M) + \mathrm{E}(M)$$

$$= \sum_{i=1}^{w}\sum_{j=1}^{h}\frac{\left(m_{i,j} - d_{i,j}\right)^2}{2\sigma_D^2} + \sum_{i=1}^{w-1}\sum_{j=1}^{h-1}\frac{\left(m_{i+1,j} - m_{i,j}\right)^2 + \left(m_{i,j+1} - m_{i,j}\right)^2}{2\sigma_M^2}$$

Equivalent minimization objective

$$\sum_{i=1}^{w}\sum_{j=1}^{h}\left(m_{i.j} - d_{i,j}\right)^2 + \frac{\sigma_D^2}{\sigma_M^2}\sum_{i=1}^{w-1}\sum_{j=1}^{h-1}\left(m_{i+1,j} - m_{i,j}\right)^2 + \left(m_{i,j+1} - m_{i,j}\right)^2$$

Continuous

$$\int_{\Omega}\left(m(\mathbf{x}) - d(\mathbf{x})\right)^2 d\mathbf{x} + \frac{\sigma_M^2}{\sigma_D^2}\int_{\Omega}\|\nabla m(\mathbf{x})\|^2 d\mathbf{x}$$

Numerical Solution

Looks familiar?

- Solution via linear system

Variant

- Penalize l_1 norm instead of l_2 norm of gradients

$$\int_\Omega \left(m(\mathbf{x}) - d(\mathbf{x})\right)^2 d\mathbf{x} + \frac{\sigma_D^2}{\sigma_M^2} \int_\Omega \|\nabla m(\mathbf{x})\|^1 d\mathbf{x}$$

- Laplace distribution (single exponential)
- Yields sharper images (natural image statistics)

Technical Remark

Image prior

$$-\ln P(M) = \sum_{i=1}^{w-1} \sum_{j=1}^{h-1} \frac{\left(m_{i+1,j} - m_{i,j}\right)^2 + \left(m_{i,j+1} - m_{i,j}\right)^2}{2\sigma_M^2} + \frac{wh}{\sigma_M \sqrt{2\pi}}$$

- This is an "improper prior"
  - Does not integrate to one!
  - Infinite subspaces without penalty
- Formal fix
  - Assume broader prior on function value itself: $f \sim N_{0,\sigma_{very\ large}}$
- For MAP estimation, this does not matter
  - We just find a point of maximum density
  - Integration not required

Fancy Inference Schemes

Belief Propagation

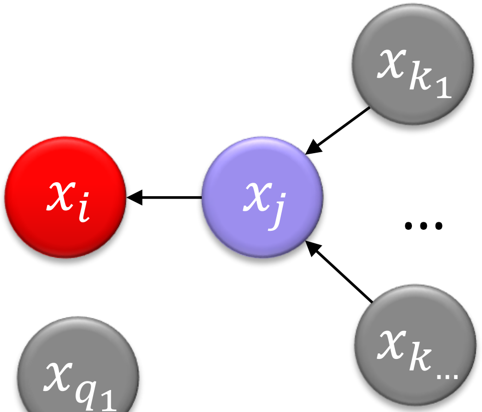
Model

$$p(\mathbf{x}) = \frac{1}{Z}\prod_{i=1}^{n} \underline{p_i^{(1)}(x_i)} \prod_{i,j\in E} \underline{p_{i,j}^{(2)}(x_i,x_j)}$$

Messages

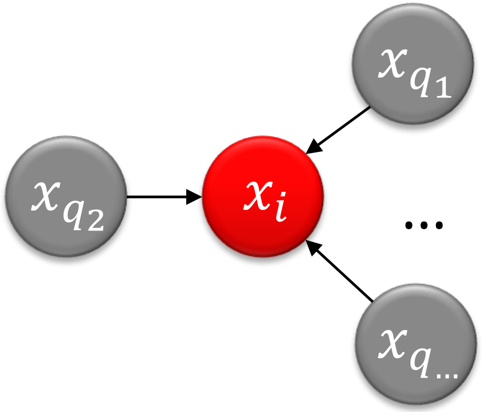
- Maximum a posteriori inference?
- Use "max-marginal":

$$m_{j,i}(x_i) = \max_{x_j=1\ldots k} p_i^{(1)}(x_j)\, p_{i,j}^{(2)}(x_i,x_j) \prod_{q\in N(j)\setminus x_i} m_{q,j}(x_q)$$



$$b_i(x_i) = \frac{1}{z_i} p_i^{(1)}(x_i) \prod_{q\in N(i)} m_{q,i}(x_q)$$

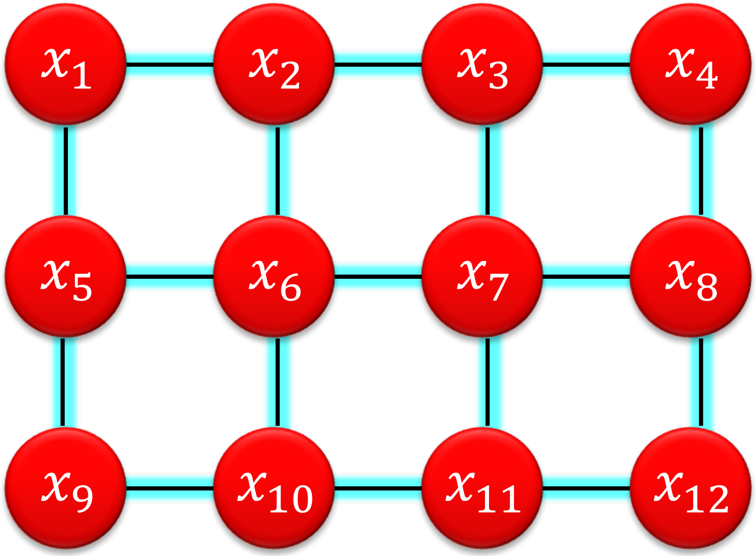
most likely state = $\arg\max_{x_i} b_i(x_i)$



Loopy BP

Loopy BP

- Loops? Which loops?
- Just run BP on loopy graph
- Arbitrary order

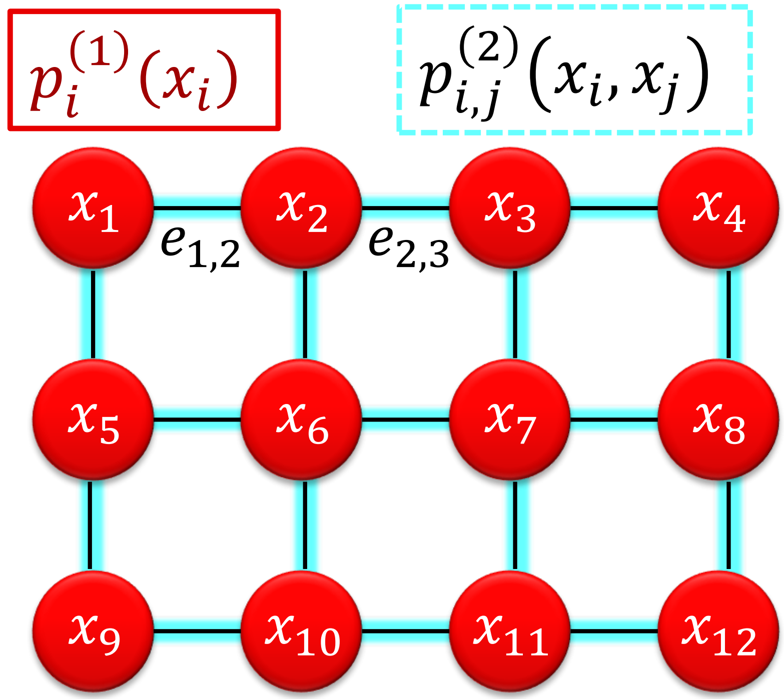


Result

- No guarantees (results can be wrong)
- Most often, results will be wrong
- Frequently still a reasonable approximation
  - Problem dependent, but worth a try. Popular 10-20 years ago.

Graph Cut

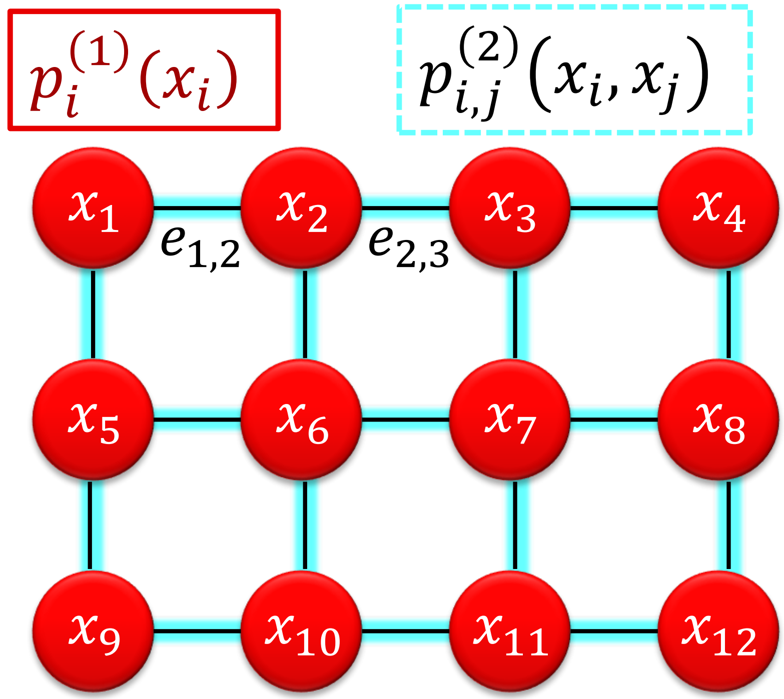
Pairwise MRF



Pairwise MRF

$$p(x_1, \dots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} \underline{p_i^{(1)}(x_i)} \prod_{i,j \in E} \underline{p_{i,j}^{(2)}(x_i, x_j)}$$

Neg-Log Likelihood



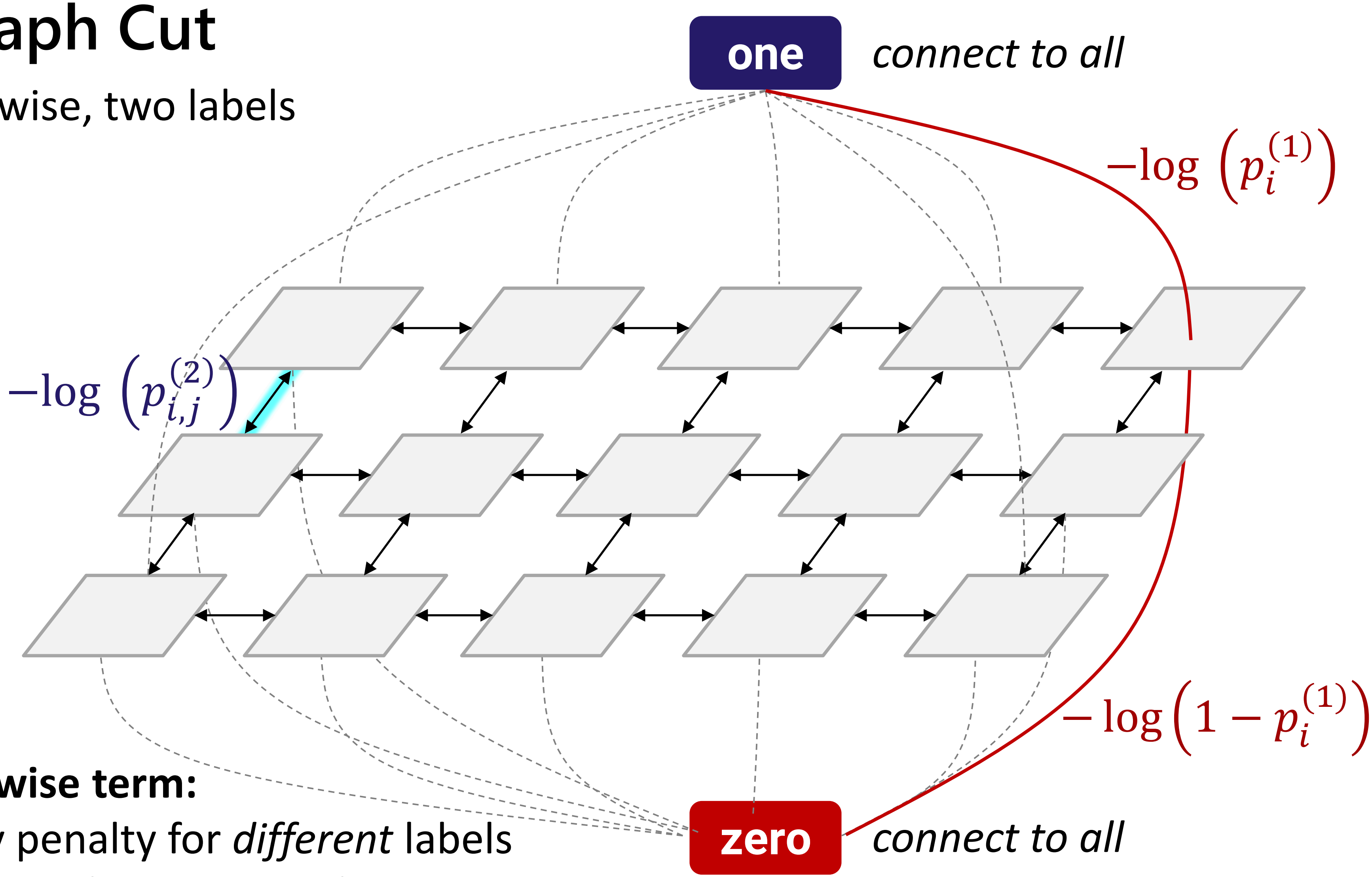
Pairwise MRF

$$-\log\big(p(x_1, \dots, x_n)\big) = \sum_{i=1}^{n} \underline{\left[-\log\left(p_i^{(1)}(x_i)\right)\right]} \sum_{i,j \in E} \underline{\left[-\log\left(p_{i,j}^{(2)}(x_i, x_j)\right)\right]}$$

Pairwise MRF Inference

Graph Cut

Pairwise, two labels



Pairwise term:
Only penalty for *different* labels possible (submodular!)

Example Application

"Grab-Cut"

- Rother, Kolmogorov, Blake
  Siggraph 2004

Probabilistic Model

- Per pixel: "Gaussian Mixture"
  of pixel colors
  - Foreground (red box)
    and background (rest)
- Pairwise: Neighboring Pixels
  - Different Label (f/b) incur fixed cost
- Graph-Cut Inference

MCMC

More General Tools

MCMC (Markov Chain Monte Carlo)

- Gibbs Sampling
- Metropolis algorithm
- Many variants

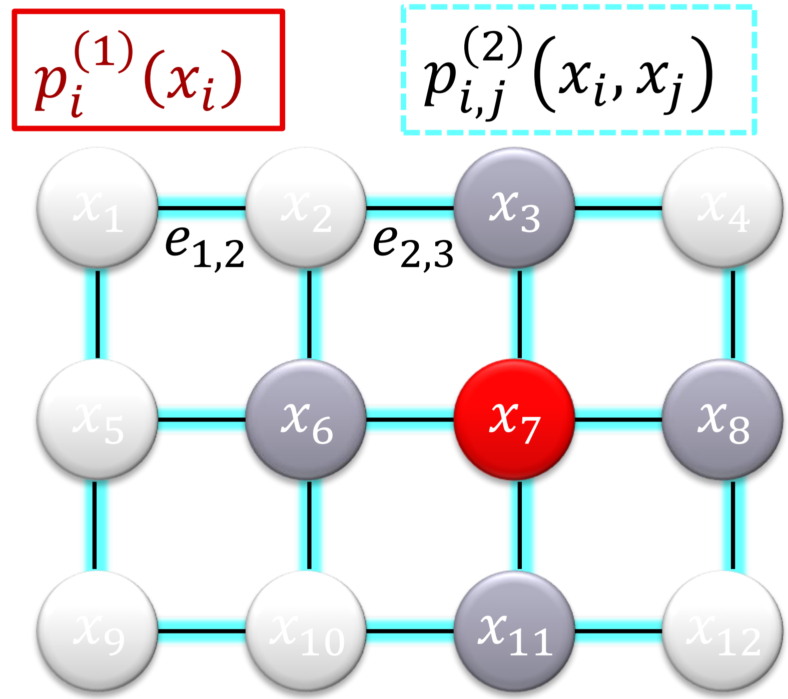
Idea

Naiive Sampling

- Infeasible (exponential)

Gibbs Sampling

- Random initialization
- Select random node
  - Fix neighbors
  - Compute local distribution
  - Sample
- Repeat



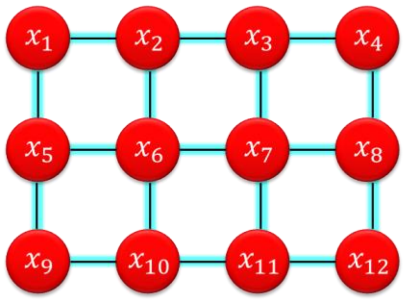
Convergence: Mixing times (hard to estimate)

Literature

S. Geman & D Geman:
Stochastic Relaxation, Gibbs Distributions,
and the Bayesian Restoration of Images.
In: IEEE Transactions on Pattern Analysis and
Machine Intelligence (PAMI) 6(6), Nov. 1984.

Learning MRF-Models from training data

Learning



Maximum Likelihood:

Maximize

$$p(x_1, \dots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} p_i^{(1)}(x_i) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j)$$

$$= \frac{\prod_{i=1}^{n} p_i^{(1)}(x_i) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j)}{\int_{x_1,\dots,x_n} \prod_{i=1}^{n} p_i^{(1)}(x_i) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j)\, dx_1 \cdots dx_n}$$

$$= \frac{P_X^{(1)}(\theta) P_X^{(2)}(\theta)}{Z(\theta)}$$

(θ denoting set of learnable parameters,
X denoting training data)

$Z(\theta)$ is numerically intractable for high dimensions, general distributions

In Practice

Learning MRFs in practice

- Tractable models, e.g. Gaussian
  - Often too simple
- Ignore Z
  - kind of wrong
  - but easy to implement
- Approximate Z
  - Mean-field theory; general: Variational Bayes
  - Replace integral by maximum (convex Z; still kind of wrong)
- Direct discriminative learning (ignore Bayes rule)