Lecture 4: Learning

Probabilistic Graphical Models, Koller and Friedman:

- Chap 13 and Chap 17

- MAP inference, Max-Product
- Parameter Estimation, Max Likelihood Estimation, Sufficient Statistics, Bayesian Parameter Estimation, Conjugate Prior, Gaussian/Beta/Dirichlet Distributions,

MAP Inference

- Find the state that maximizes $P(\boldsymbol{x})$.
- Imagine we did message passing to get all the marginals.
- Could we then just do
  - For all i $x_i = argmax\ \ P(x_i)$?
- No!
- That is why the book introduces 'max-marginals'

	$x_1 = 0$	$x_1 = 1$
$x_2 = 0$	0.3	0.4
$x_2 = 1$	0.3	0.0
marginal $p(x_1)$	0.6	0.4

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Trick is to distribute the max instead of the sum

$$
\begin{aligned}
\max_x f(x) &= \max_{x_1,x_2,x_3,x_4} \phi(x_1,x_2)\phi(x_2,x_3)\phi(x_3,x_4) \\
&= \max_{x_1,x_2,x_3} \phi(x_1,x_2)\phi(x_2,x_3)\underbrace{\max_{x_4}\phi(x_3,x_4)}_{\gamma(x_3)} \\
&= \max_{x_1,x_2} \phi(x_1,x_2)\underbrace{\max_{x_3}\phi(x_2,x_3)\gamma(x_3)}_{\gamma(x_2)} \\
&= \max_{x_1}\underbrace{\max_{x_2}\phi(x_1,x_2)\gamma(x_2)}_{\gamma(x_1)} \\
&= \max_{x_1}\gamma(x_1)
\end{aligned}
$$

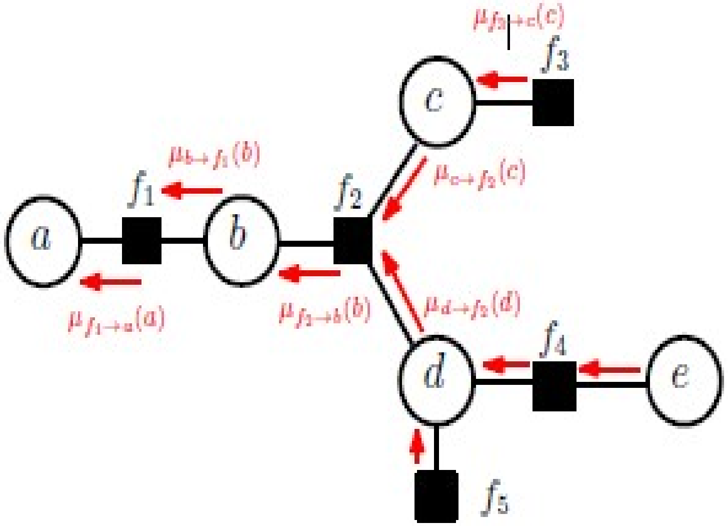
Figure from the lectures of: Bjoern Andres and Bernt Schiele

Then backtrack to the answer

$$
\begin{aligned}
x_1^* &= \operatorname*{argmax}_{x_1} \gamma(x_1) \\
x_2^* &= \operatorname*{argmax}_{x_2} \phi(x_1^*, x_2)\gamma(x_2) \\
x_3^* &= \operatorname*{argmax}_{x_3} \phi(x_2^*, x_3)\gamma(x_3) \\
x_4^* &= \operatorname*{argmax}_{x_4} \phi(x_3^*, x_4)\gamma(x_4)
\end{aligned}
$$

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Life in the Trees



$$\max_x f(x) = \max_{a,b,c,d,e} f_1(a,b) f_2(b,c,d) f_3(c) f_4(d,e) f_5(d)$$

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Life in the Trees is easy

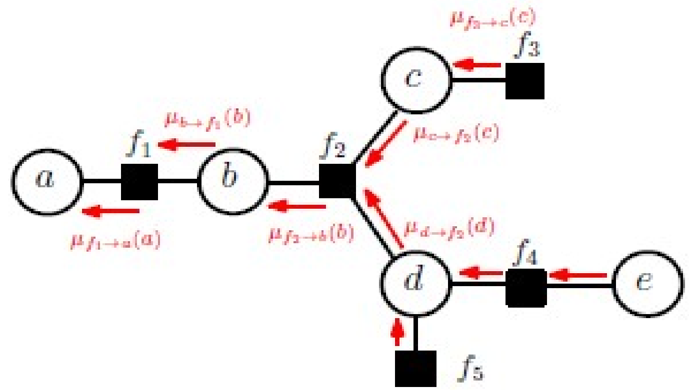


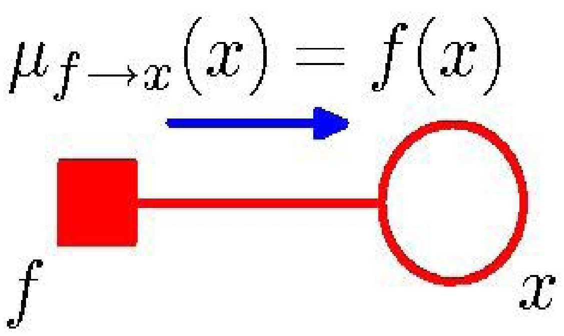
Table with a max over e *for each* d

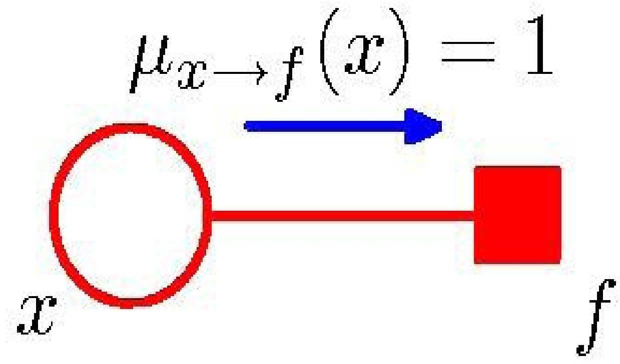
$$
\begin{aligned}
\max_x f(x) &= \max_{a,b,c,d,e} f_1(a,b) f_2(b,c,d) f_3(c) f_4(d,e) f_5(d) \\
&= \max_a \max_b f_1(a,b) \max_{c,d} f_2(b,c,d) f_3(c) \underbrace{f_5(d)}_{\mu_{f_5\to d}(d)} \underbrace{\max_e f_4(d,e)}_{\mu_{f_4\to d}(d)} \\
&= \max_a \max_b f_1(a,b) \max_{c,d} f_2(b,c,d) \underbrace{f_3(c)}_{\mu_{c\to f_2}(c)} \underbrace{\mu_{f_5\to d}(d)\mu_{f_4\to d}(d)}_{\mu_{d\to f_2}(d)} \\
&= \max_a \max_b f_1(a,b) \underbrace{\max_{c,d} f_2(b,c,d) \mu_{c\to f_2}(c) \mu_{d\to f_2}(d)}_{\mu_{f_2\to b}(b)} \\
&= \max_a \max_b f_1(a,b) \underbrace{\mu_{f_2\to b}(b)}
\end{aligned}
$$

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Max-Product Algorithm

- Pick a root
- Initialize marginals in leaf factor nodes to factors & messages from variable nodes to 1.





Max-Product Algorithm

- Product step

$$\mu_{x \to f}(x) = \prod_{g \in \{\mathrm{ne}(x) \setminus f\}} \mu_{g \to x}(x)$$

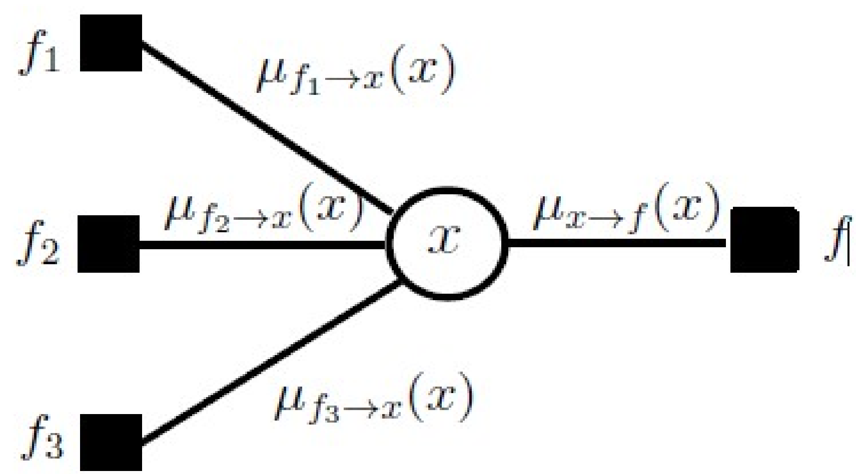


Figure from the lectures of: Bjoern Andres and Bernt Schiele

Max-Product Algorithm

- Max step (define $\phi_f(\chi_f) = f(y_1, y_2, y_3, x)$)

$$\mu_{f\to x}(x) = \max_{y\in\mathcal{X}_f\setminus x} \phi_f(\mathcal{X}_f) \prod_{y\in\{\text{ne}(f)\setminus x\}} \mu_{y\to f}(y)$$

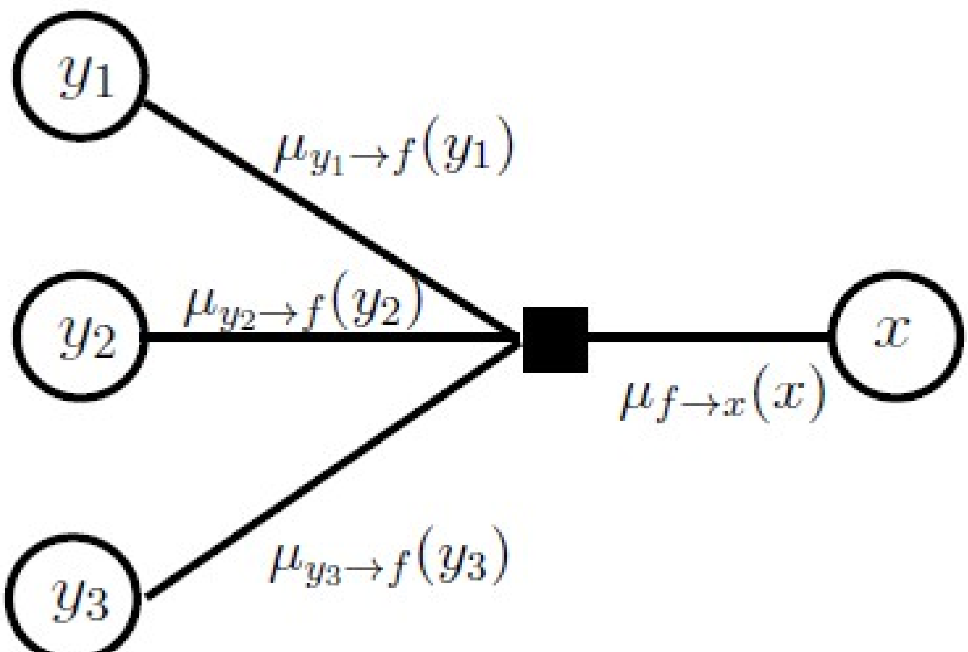


Figure from the lectures of: Bjoern Andres and Bernt Schiele

Max-Product Algorithm

- 'Max Marginal'
- This can be used to compute MAP solution.
- Taking log leads to max sum alg.

$$x^* = \underset{x}{\operatorname{argmax}} \prod_{f \in \mathrm{ne}(x)} \mu_{f \to x}(x)$$

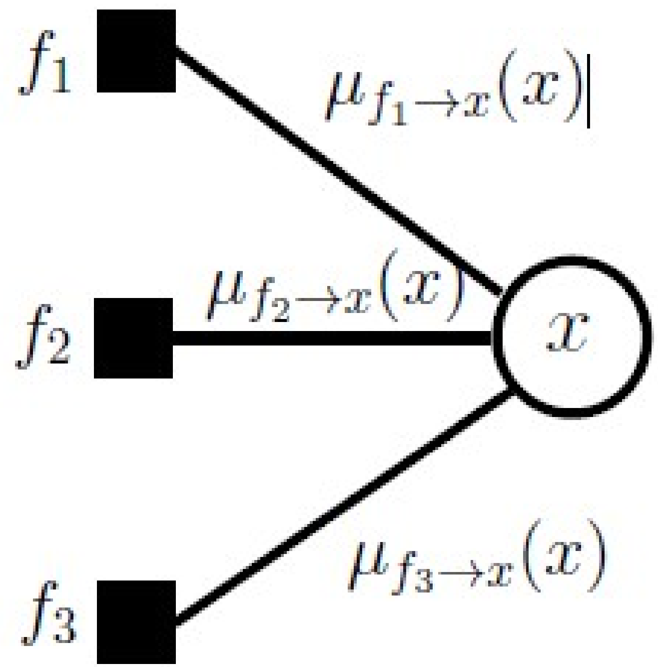
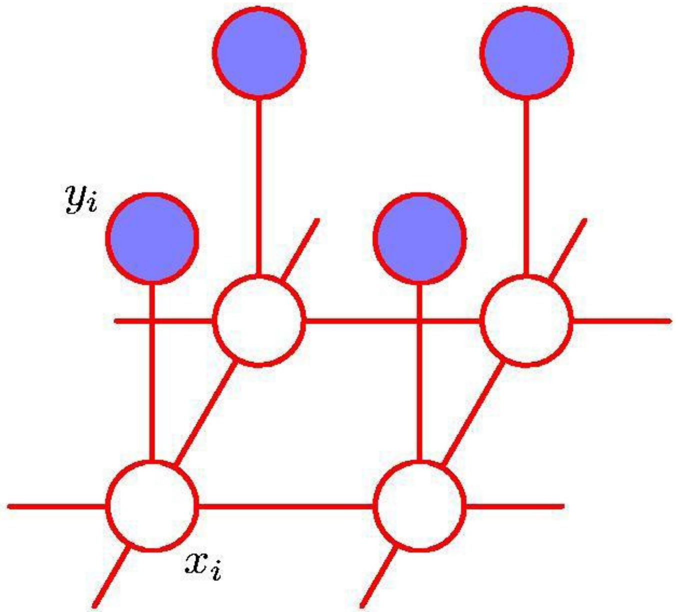


Figure from the lectures of: Bjoern Andres and Bernt Schiele

Tutorial 3: MRF-Graph Cuts



- Here the x_i are a hidden segment label.
- y_i are the observed image pixel value.
- So we want a to find the MAP,

  maximum a posteori, estimate **x** given **y.**
- Uses an exponential model for $\phi(x , y) \propto \exp(-E(\mathbf{x},\mathbf{y}))$
- The ‘Gibbs Energy’, E, Has terms for each type of edge above.
- The smoothness term $V(x_i, x_j)$ is a constant for neighboring pixels with different labels.
- MAP is same as minimize the ‘energy’ with respect to labels x
- Cleverly this can be transformed to a Graph Cut or Max Flow problem that is easy.

Graph Cuts for MAP

The corresponding factor graph has exponential factor nodes and only z variables

- *Nonzero* $e_i(z_i)$ *(energies):*

  $e_1(0) = 7 \quad e_2(1) = 2$

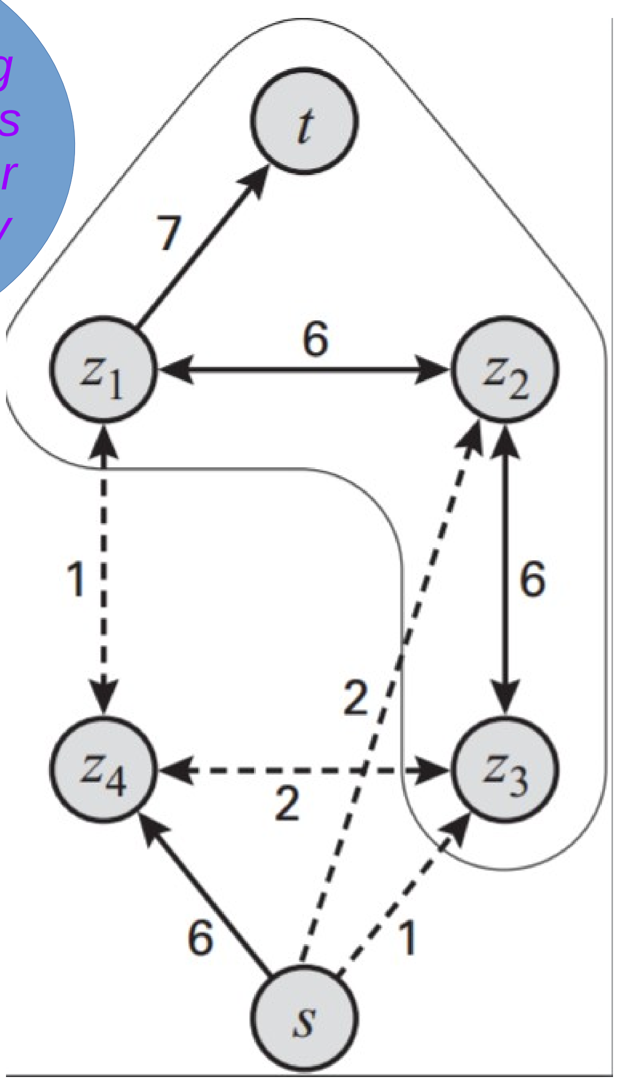
  $e_3(1) = 1 \quad e_4(1) = 6$

- Energies $e_{ij}(z_i, z_j) = \lambda_{ij}$ when $z_i = z_j$:

  $\lambda_{12} = 6 \quad \lambda_{23} = 6$

  $\lambda_{34} = 2 \quad \lambda_{14} = 1$

- Find $max_{zi} \{\Sigma_{ij} e_{ij}(z_i, z_j) + \Sigma_i e_i(z_i)\}$



Parameter Estimation

- Point Estimates: Trying to estimate one value of θ

  Commonly used:

  - Maximum Likelihood Esitmation MLE,
  - Maximum-A-Posteriori MAP
  - Minimum Expected Loss/Cost/Risk (Energy=MLE)

- Bayesian Estimation: Estimate the whole dist, $p(\theta \mid D)$

  Specify a prior distribution $p(\theta)$

  Integrate out the variable:

$$p(x \mid \mathcal{D}) = \int_\theta p(x \mid \theta) p(\theta \mid \mathcal{D}) d\theta$$

  Usually intractable because of the integral

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Bayesian Parameter Estimation

- Usually intractable because of the integral

$$p(x \mid \mathcal{D}) = \int_{\theta} p(x \mid \theta) p(\theta \mid \mathcal{D}) d\theta$$

- We can do it using Monte Carlo Approximation

$$p_N(x) = \frac{1}{N} \sum_{i=1}^{N} \delta_{x^{(i)}}(x),$$

$$I_N(f) = \frac{1}{N} \sum_{i=1}^{N} f\left(x^{(i)}\right) \xrightarrow[N \to \infty]{a.s.} I(f) = \int_{\mathcal{X}} f(x) p(x) dx.$$

Inference Problem

Given a dataset $\mathcal{D} = \{x_1, ..., x_n\}$:

Bayes Rule:

$$P(\theta|\mathcal{D}) = \frac{P(D|\theta)P(\theta)}{P(\mathcal{D})}$$

$$\begin{aligned} P(\mathcal{D}|\theta) & \quad \text{Likelihood function of } \theta \\ P(\theta) & \quad \text{Prior probability of } \theta \\ P(\theta|\mathcal{D}) & \quad \text{Posterior distribution over } \theta \end{aligned}$$

Computing posterior distribution is known as the **inference** problem.
But:

$$P(\mathcal{D}) \; = \; \int P(\mathcal{D}, \theta) d\theta$$

This integral can be very high-dimensional and difficult to compute.

Prediction

$$P(\theta|\mathcal{D}) = \frac{P(D|\theta)P(\theta)}{P(\mathcal{D})}$$

$P(\mathcal{D}\|\theta)$	Likelihood function of θ
$P(\theta)$	Prior probability of θ
$P(\theta\|\mathcal{D})$	Posterior distribution over θ

Prediction: Given $\mathcal{D}$, computing conditional probability of x requires computing the following integral:

$$\begin{aligned} P(x|\mathcal{D}) &= \int P(x|\theta,\mathcal{D})P(\theta|\mathcal{D})d\theta \\ &= \mathbb{E}_{P(\theta|\mathcal{D})}[P(x|\theta,\mathcal{D})] \end{aligned}$$

which is sometimes called **predictive distribution**.

Computing predictive distribution requires posterior $P(\theta|\mathcal{D})$.

Model Selection

Compare model classes, e.g. $\mathcal{M}_1$ and $\mathcal{M}_2$. Need to compute posterior probabilities given $\mathcal{D}$:

$$P(\mathcal{M}|\mathcal{D}) = \frac{P(\mathcal{D}|\mathcal{M})P(\mathcal{M})}{P(\mathcal{D})}$$

where

$$P(\mathcal{D}|\mathcal{M}) = \int P(\mathcal{D}|\theta, \mathcal{M})P(\theta|\mathcal{M})d\theta$$

is known as the **marginal likelihood** or **evidence**.

Is that it?

- No it is not.
- The integrals can be intractable so we need to find a good way to approximate them.
- Monte Carlo Methods do just that (Lecture 6)
- We will not do that yet, but rather
- Some mathematically tractable problems

Max Likelihood Estimation

- Given
  - Training data : $= \{x_1,..., x_N\}$
  - Model parameterized probabilities, $P(x_i \mid \theta)$;

    (e.g. a factor graph with Gaussian factors)
- Problem : Find θ^* such that $P(\mathbf{x} \mid \theta^*)$ best fits the data

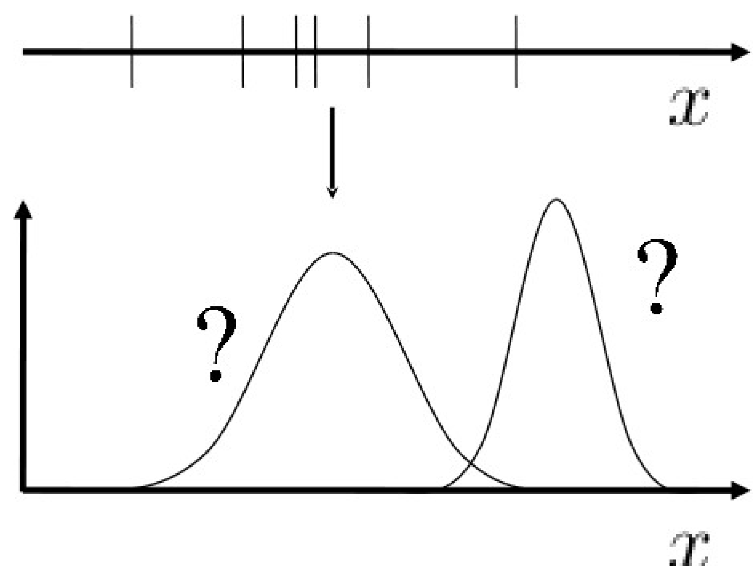


Figure from the lectures of: Bjoern Andres and Bernt Schiele

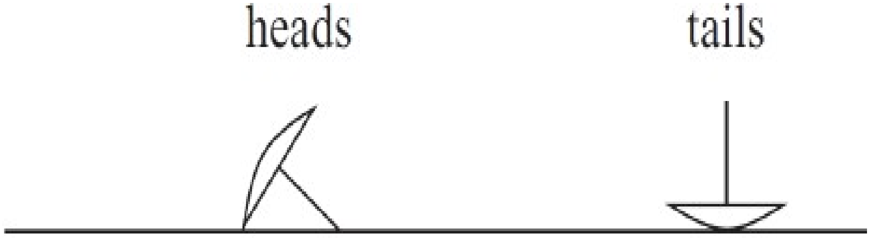
Maximum Likelihood Estimation

- Aim to estimate one single $\boldsymbol{\theta}$ (a point estimate)
- Likelihood of the data: $L(\boldsymbol{\theta}) = L(\boldsymbol{\theta}\,; D) = P(D \mid \boldsymbol{\theta})$
- Assume that the data is independent and identically distributed (iid)
- $L(\boldsymbol{\theta}) = P(D \mid \boldsymbol{\theta}) = \prod_{\mathrm{D}} P(\boldsymbol{x}_i \mid \boldsymbol{\theta})$
- $l(\boldsymbol{\theta}) = \ln P(D \mid \boldsymbol{\theta}) = \Sigma_{\mathrm{D}} \ln P(\boldsymbol{x}_i \mid \boldsymbol{\theta})$
  - Empirical expected log-likelihood;
  - Minus empirical 'log-loss' or energy or entropy;
  - ≤ 0 (0 is deterministic data and model)
  - More spread $\Rightarrow$ more negative

Thumbtack MLE

Training data : $\mathcal{D} = \{1,0,0,1,1,0...\}$

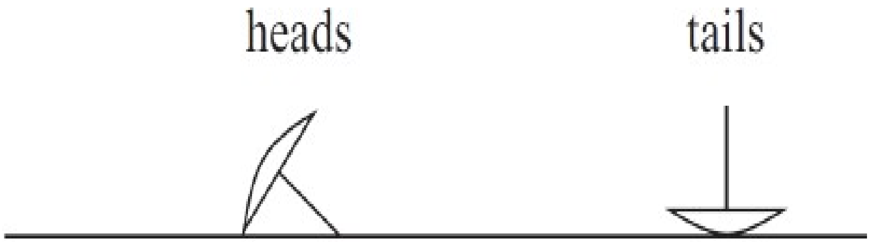
– $p(x_i = 1 | \theta) = \theta;$ $p(x_i = 0 | \theta) = 1\text{-}\theta$



Thumbtack MLE

Training data : $\mathcal{D} = \{1,0,0,1,1,0...\}$

- $p(x_i = 1 | \theta) = \theta;$ $p(x_i = 0 | \theta) = 1-\theta$
- $p(x | \theta) = \theta^x (1-\theta)^{1-x}$



Thumbtack MLE

Training data : $\mathcal{D} = \{1,0,0,1,1,0...\}$

- $p(x_i = 1 | \theta) = \theta;$ $\quad p(x_i = 0 | \theta) = 1 - \theta$
- $p(x | \theta) = \theta^x (1-\theta)^{1-x}$
- $l(\theta; D) = \Sigma_D \ln \theta^x (1-\theta)^{1-x}$

Thumbtack MLE

Training data : $\mathcal{D} = \{1,0,0,1,1,0...\}$

- $p(x_i = 1 | \theta) = \theta;$ $\quad\quad p(x_i = 0 | \theta) = 1\text{-}\theta$
- $p(x | \theta) = \theta^x (1\text{-}\theta)^{1\text{-}x}$
- $l(\theta; D) = \Sigma_D \ln \theta^x (1\text{-}\theta)^{1\text{-}x}$

$$= n_1 \ln \theta + n_0 \ln (1\text{-}\theta)$$

Thumbtack MLE

Training data : $\mathcal{D} = \{1,0,0,1,1,0...\}$

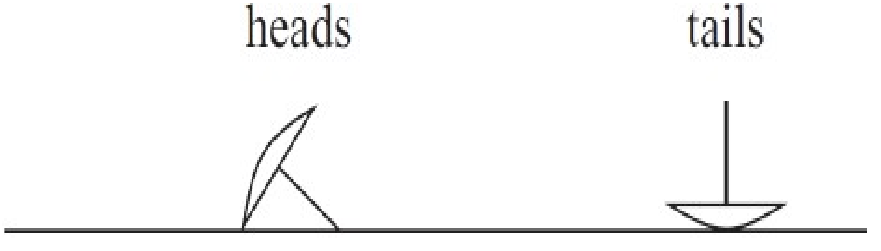
- $p(x_i = 1 | \theta) = \theta; \qquad p(x_i = 0 | \theta) = 1 - \theta$
- $p(x | \theta) = \theta^x (1-\theta)^{1-x}$
- $l(\theta; D) = \Sigma_D \ln \theta^x (1-\theta)^{1-x}$

$$= n_1 \ln \theta + n_0 \ln (1-\theta)$$

$$0 = n_1 / \theta - n_0 / (1-\theta)$$

$$n_1 (1-\theta) = n_0 \theta$$

$$\theta = n_1 / (n_1 + n_0)$$



Multinomial Distribution

We have N bins to choose between for x.

θ_k = probability of kth bin/outcome;

$\Sigma_K \theta_k = 1.$

- Same reasoning as for binomial case gives MLE:

$\theta_k = n_k /(n_1+n_2 + \ldots n_N)$

Gaussians

- $p(x \mid \mu, \sigma) = (2\pi\sigma)^{-½} \exp -(x - \mu)^2 / (2\sigma^2))$
- MLE:

  Remember 'Sufficient Statistics' and 'Moment matching'

  $\mu = \Sigma_m x_m / M$

  $\sigma^2 = \Sigma_m x_m^2 / M - \mu^2$

Gaussians

- $p(x \mid \mu, \sigma) = (2\pi\sigma^2)^{-½} \exp -(x - \mu)^2 / (2\sigma^2))$
- $l(\theta; D) = \Sigma_D ½[\ln(2\pi\sigma^2) + (x - \mu)^2 / \sigma^2]$

$$0 = \Sigma_D (x - \mu) / \sigma^2$$

$$\Rightarrow \mu = (1/M)\Sigma_D x$$

$$0 = \Sigma_D [1/(\sigma^2) - (x - \mu)^2 / (\sigma^2)^2]$$

$$= \Sigma_D [(\sigma^2) - (x - \mu)^2$$

$$\Rightarrow M\sigma^2 = \Sigma_D (x - \mu)^2 = \Sigma_D (x^2 - 2x\mu + \mu^2)$$

$$= \Sigma_D x^2 - 2M\mu\mu + M\mu^2 = \Sigma_D x^2 - M\mu^2$$

$$\sigma^2 = \Sigma_D x^2 / M - \mu^2$$

Sufficient Statistics

- Any two datasets with the same 'sufficient statistics', eg. $\Sigma_D \, \tau(\boldsymbol{\theta})$, will have the same likelihood for any choice of parameters $\boldsymbol{\theta}$.

- The value of these sufficient stats then are all we need to compute the MLE parameters.

  Examples:

  - counts per bin for multinomial,
  - moments for exponentials,...

MLE in Bayes Nets with Table CPDs

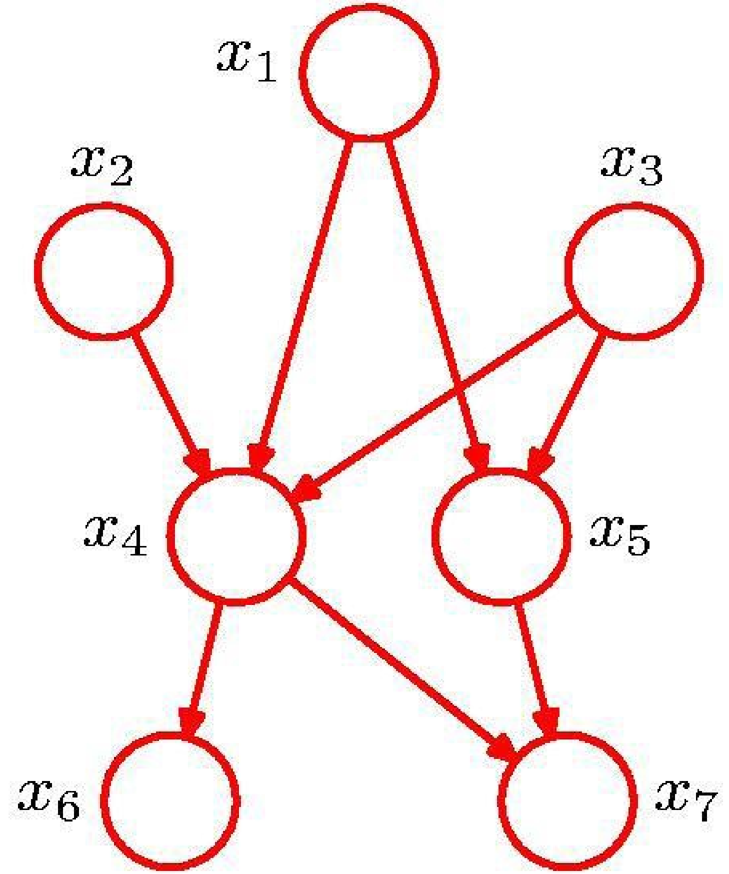
- Roots are easy:

  $P(x_1 = v_{1k} : ie\ k^{th}\ value) = \theta_k = n_k / \mathbf{\Sigma}_i n_i$

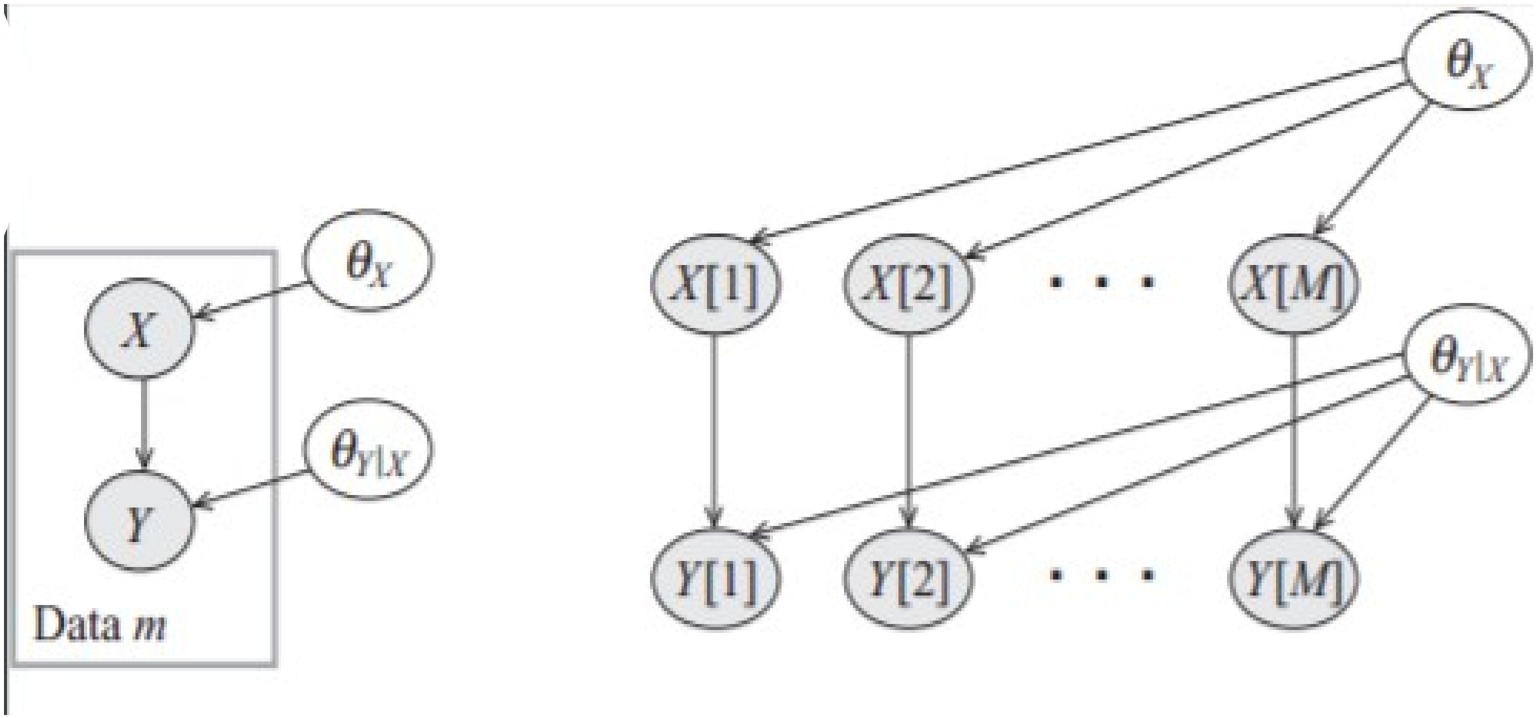
- For the others we treat them all as separate MLE problems given each possible assignment of values to parents.
  - Have to count cases such as:

    $x_1 = 1, x_2 = 4$ and $x_4 = 6$

- Called global decomposition into local likelihoods.

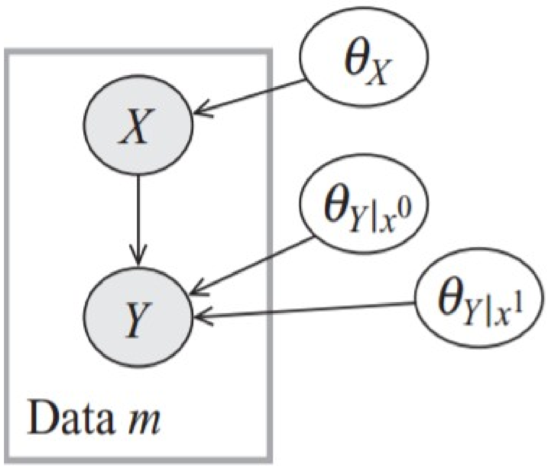


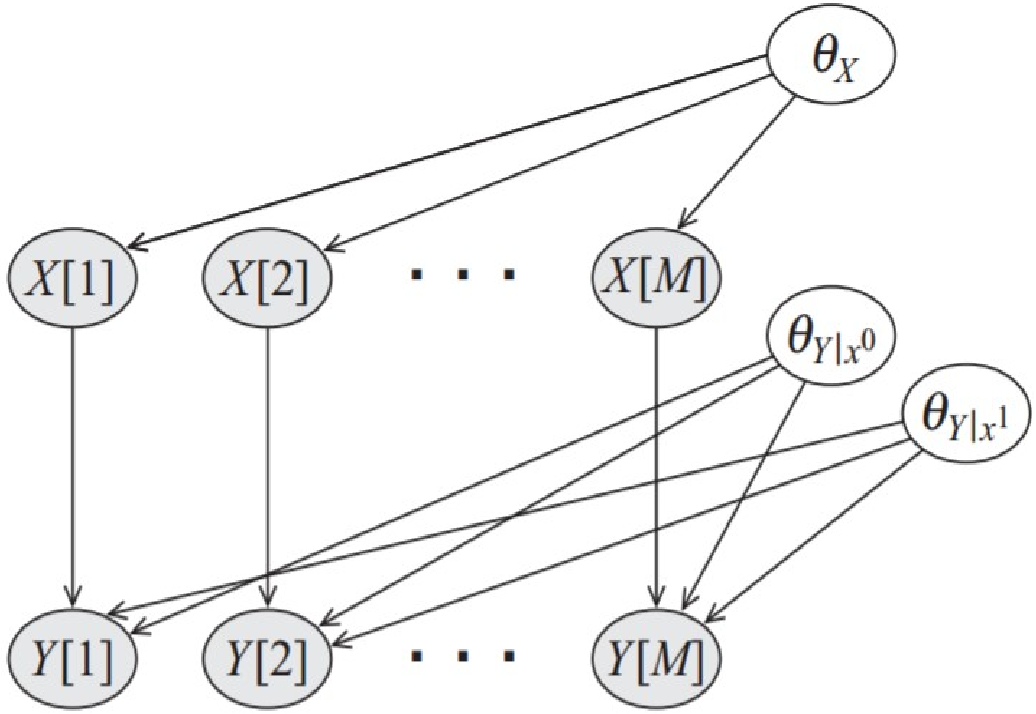
Global Decomposition



- Table CPDs
- $p(\theta_X, \theta_{Y|X} \mid D) = p(\theta_X \mid D)\, p(\theta_{Y|X} \mid D)$
- Looks ok.

Posteriori

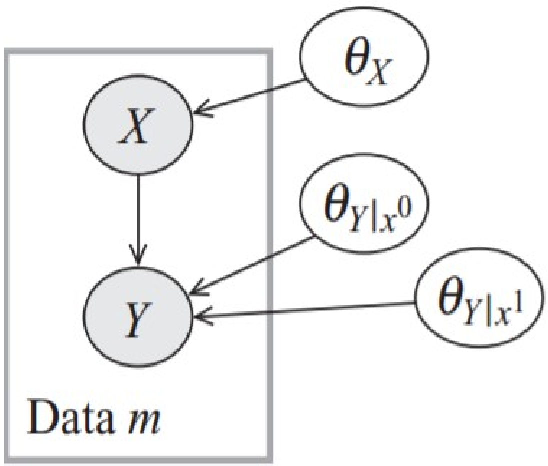


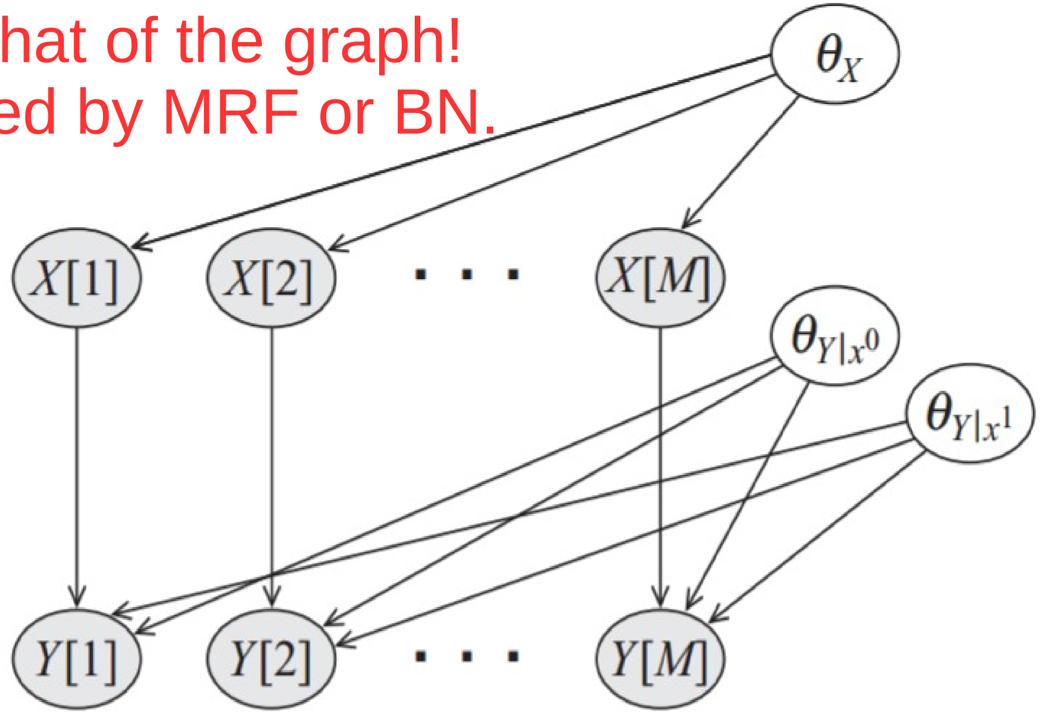


- $\theta_{Y|X0}$ and $\theta_{Y|X1}$ are not d-separated
- They are still independent | x since y depends on them disjointly, ie in different cases for x.

Posteriori

Case where I-map is not that of the graph!
And can not be represented by MRF or BN.





- $\theta_{Y|X0}$ and $\theta_{Y|X1}$ are not d-separated
- They are still independent | x since y depends on them disjointly, ie in different cases for x.

Bayesian Parameter Estimation

- Aim to estimate $p(\theta \mid D)$
- $p(\theta \mid D) = P(D \mid \theta) \; p(\theta) \,/\, P(D)$
- We need a prior, $p(\theta)$ and might want to normalize, $P(D)$.

Bayesian Parameter Estimation

- Aim to estimate $p(\boldsymbol{\theta} \mid D)$
- $p(\boldsymbol{\theta} \mid D) = P(D \mid \boldsymbol{\theta})\ \ p(\boldsymbol{\theta}) / P(D)$
- We need a prior, $p(\boldsymbol{\theta})$ and might want to normalize, $P(D)$.
- Giant mental leap: What do we mean by $p(\boldsymbol{\theta})$?

This is a probability of a probability!?
E.g. The probability that the probability of heads is 0.5!?
You're blowing my mind!!!!!

Bayesian Parameter Estimation

- Aim to estimate $p(\theta \mid D)$
- $p(\theta \mid D) = P(D \mid \theta)\ p(\theta) / P(D)$
- We need a prior, $p(\theta)$ and might want to normalize, $P(D)$.
- Giant mental leap: What do we mean by $p(\boldsymbol{\theta})$?

This is a probability of a probability!?
E.g. The probability that the probability of heads is 0.5!?
You're blowing my mind!!!!!

Joke from the Lectures of: Veselin Stoyanov, Alexandre Klementiev and Shane Bergsman

Bayesian Parameter Estimation

- Predict by integrating:

$$p(\boldsymbol{x}) = \int P(\boldsymbol{x} \mid \boldsymbol{\theta}) \; p(\boldsymbol{\theta} \mid \boldsymbol{D}) \, \mathrm{d}\boldsymbol{\theta}$$

- Nice if we pick a 'conjugate' prior.

Conjugate Prior

- The prior, $p(\boldsymbol{\theta})$, will have its own 'hyper-parameters α that span a family.
- If we can always find new hyper-parameters, α' to describe the posteori, $p(\boldsymbol{\theta} \mid D)$, then we say we have a conjugate prior.
- Depends on form of $P(D \mid \boldsymbol{\theta})$.

Conjugate Prior

- Binomial distribution:

$$P(D \mid \theta) = \Pi_i\, \theta^{x_i}(1-\theta)^{1-x_i}$$

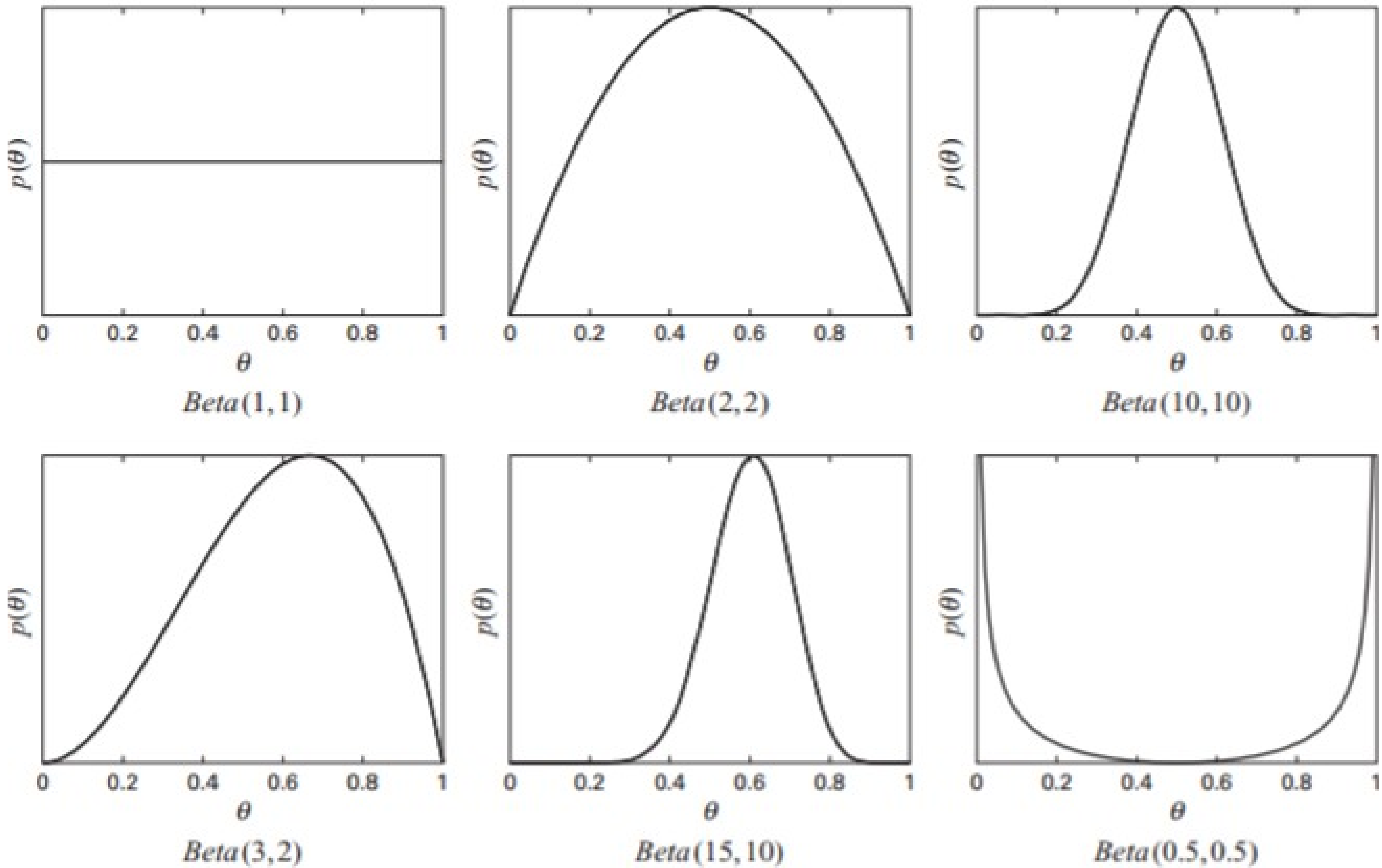
- Conjugate prior is the Beta Distribution:

$$p(\theta) = \beta(\alpha_0, \alpha_1) = \gamma\theta^{\alpha_1-1}(1-\theta)^{\alpha_0-1}$$

- $P(\theta \mid D) \propto P(D \mid \theta)\ p(\theta) = \Pi_i\, \theta^{xi}(1-\theta)^{1-xi}\ \theta^{\alpha_1-1}(1-\theta)^{\alpha_0-1}$

$$P(\theta \mid D) = \gamma\theta^{\alpha_1+n_1-1}(1-\theta)^{\alpha_0+n_2-1} = \beta(\alpha_0+n_0, \alpha_1+n_1)$$

Beta Distribution



Conjugate Prior

- $P(\theta \mid D) = \gamma\theta^{\alpha_1+n1-1}\,(1-\theta)^{\alpha_0+n_0-1} = \beta(\alpha_0 + n_0, \alpha_1 + n_1)$
- Integrating:

$P(x=1) = \int P(x=1 \mid \theta)\; p(\theta)\, d\theta = \alpha_1/(\alpha_1+\alpha_0)$

- Posteriori $P(\theta \mid D) = \beta(\alpha_0+n_0, \alpha_1+n_1)$

$P(x=1 \mid D) = (\alpha_1+n_1)/(\alpha_1+n_1+\alpha_0+n_0)$; Laplace's one

Dirichlet Distribution

- Dirichlet are conjugate for multinomial models.

$$p(\theta) = \text{Dirichlet}(\alpha_0, \ldots, \alpha_k) \propto \prod_k \theta_k^{\alpha_k - 1}$$

- $P(x = x_k) = \alpha_k / \Sigma_i \alpha_i$
- $p(\theta \mid D) = \text{Dirichlet}(\alpha_0 + n_0, \ldots, \alpha_k + n_k)$

Gaussian

- Gaussian s are conjugate for Gaussian models.

MAP Estimation

$p(\theta^* \mid D) = \max_\theta p(\theta \mid D);$

θ^* is the MAP estimate.

$P(\boldsymbol{x} \mid D) = \int P(\boldsymbol{x} \mid \theta) p(\theta \mid D) d\theta$

$\approx P(\boldsymbol{x} \mid \theta^*)$ as data gets big

MLE and MAP are representation dependent

if they involve probability density, p, not probabilities, P.

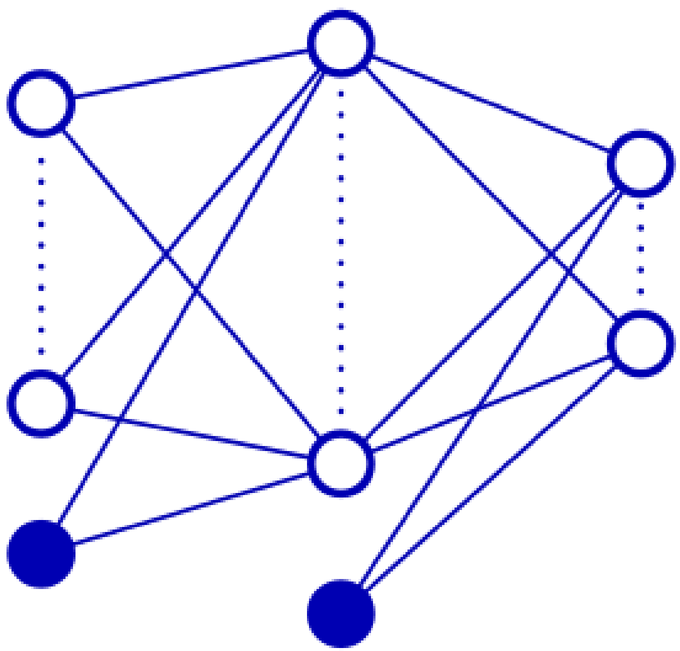
Example with Bayesian Neural Nets

We can as in Tutorial 10 use a NN to generate a mean conditional on some data:

- $\mathbf{Y} = f(\mathbf{w}, \mathbf{X})$
- Then form a conditional distribution by assuming it to be Gaussian with this mean and some Covariance.
- That then becomes our model of likelihood of some data.

Bayesian Neural Nets

Regression problem: Given a set of $i.i.d$ observations $\mathbf{X} = \{\mathbf{x}^n\}_{n=1}^N$ with corresponding targets $\mathcal{D} = \{t^n\}_{n=1}^N$.



Likelihood:

$$p(\mathcal{D}|\mathbf{X}, \mathbf{w}) = \prod_{n=1}^{N} \mathcal{N}(t^n | y(\mathbf{x}^n, \mathbf{w}), \beta^2)$$

The mean is given by the output of the neural network:

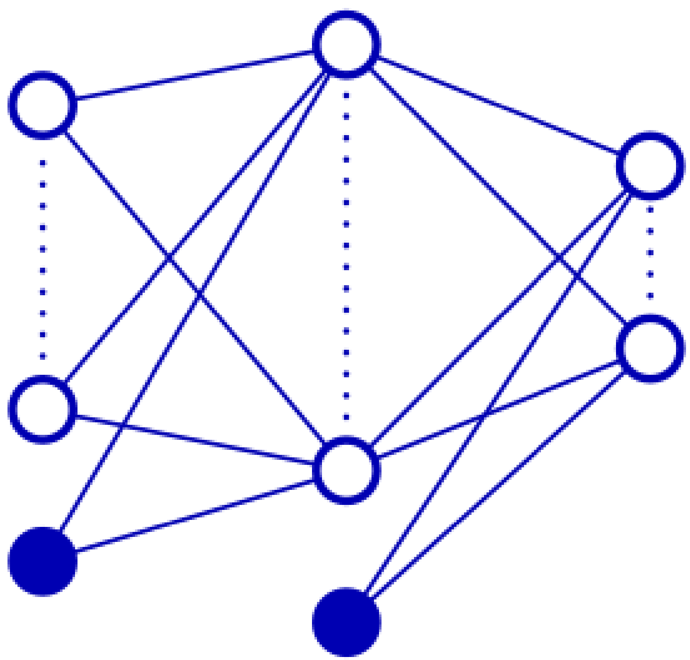
$$y_k(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M} w_{kj}^2 \sigma\Big(\sum_{i=0}^{D} w_{ji}^1 x_i\Big)$$

where $\sigma(x)$ is the sigmoid function.

Gaussian prior over the network parameters: $p(\mathbf{w}) = \mathcal{N}(0, \alpha^2 I)$.

Figure: Ruslan Salakhutdinov, BCS and CSAIL, MIT

Bayesian Neural Nets



Likelihood:

$$p(\mathcal{D}|\mathbf{X},\mathbf{w}) = \prod_{n=1}^{N} \mathcal{N}(t^n|y(\mathbf{x}^n,\mathbf{w}),\beta^2)$$

Gaussian prior over parameters:

$$p(\mathbf{w}) = \mathcal{N}(0,\alpha^2 I)$$

Posterior is analytically intractable:

$$p(\mathbf{w}|\mathcal{D},\mathbf{X}) = \frac{p(\mathcal{D}|\mathbf{w},\mathbf{X})p(\mathbf{w})}{\int p(\mathcal{D}|\mathbf{w},\mathbf{X})p(\mathbf{w})d\mathbf{w}}$$

Remark: Under certain conditions, Radford Neal (1994) showed, as the number of hidden units go to infinity, a Gaussian prior over parameters results in a Gaussian process prior for functions.

Notice this analysis only gives an evaluation of the model pdf at a given **w**. It can be used with Importance sampling to compute integrals.

Figure: Ruslan Salakhutdinov, BCS and CSAIL, MIT