Bayesian Networks and Belief Propagation

Mohammad Emtiyaz Khan
EPFL

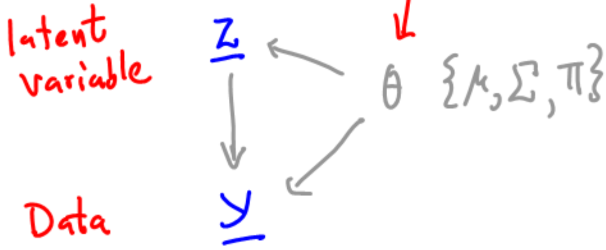
Nov 26, 2015

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Revision

Motivation

When the data is *structured*, e.g. time-series data, prediction can be improved by incorporating the structure into the model. This can be accomplished ~~by~~ using ideas from both probability theory and graph theory. The resulting field is called probabilistic graphical model.



$p_\theta(\underline{y}, \underline{z})$

Inference: Given $\underline{y}$ & θ, find $\underline{z}$.

Learning: Given $\underline{y}$, find θ

Bayes' rule:

$$p(\underline{y}, \underline{z}) = p(\underline{y}/\underline{z})\, p(\underline{z})$$

Joint Lik x prior

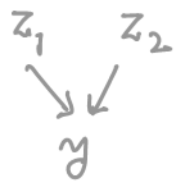
$$= p(\underline{z}/\underline{y}) \times p(\underline{y})$$

Post x Marg.Lik

We will learn about Bayesian network where the distribution *respects* a directed graphical structure. Our main goal is to learn *inference* of the latent variables using belief propagation.

An example:

$$p(y, z_1, z_2) = p(y/z_1, z_2)\, p(z_1)\, p(z_2)$$



Bayesian network

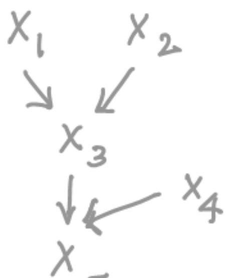
Given a directed acyclic graph G and parameters θ, a Bayesian network defines the joint distribution as follows,

$$p_\theta(\mathbf{x}) = \prod_{k=1}^{K} p_\theta(x_k|\mathrm{pa}_k),$$

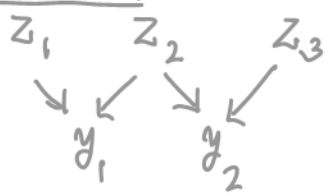
where pa_k are the parents of x_k.

Another example:

$$p(\underline{x}) = p(x_5/x_3, x_4)\, p(x_4)\, p(x_3/x_1, x_2)\, p(x_1)\, p(x_2)$$



Another example:



$$p(y_1, y_2, z_1, z_2, z_3) = p_\theta(z_1)\, p_\theta(z_2)\, p_\theta(z_3)\, p_\theta(y_1/z_1, z_2)\, p_\theta(y_2/z_2, z_3)$$

A simple example

Moby is the pet fish and Fluffy is the pet cat. One day, you ~~observed~~ found that Moby is missing. At the same time, you notice that Fluffy's food bowl is full. You wonder what happened.

$p(z_3=0) = 0.99$

$p(z_3=1) = 0.01$

Blue numbers are θ.

There are three possibilities: (1) Moby commited suicide (2) Fluffy ate Moby (3) Fluffy is sick.

$p(z_1)$

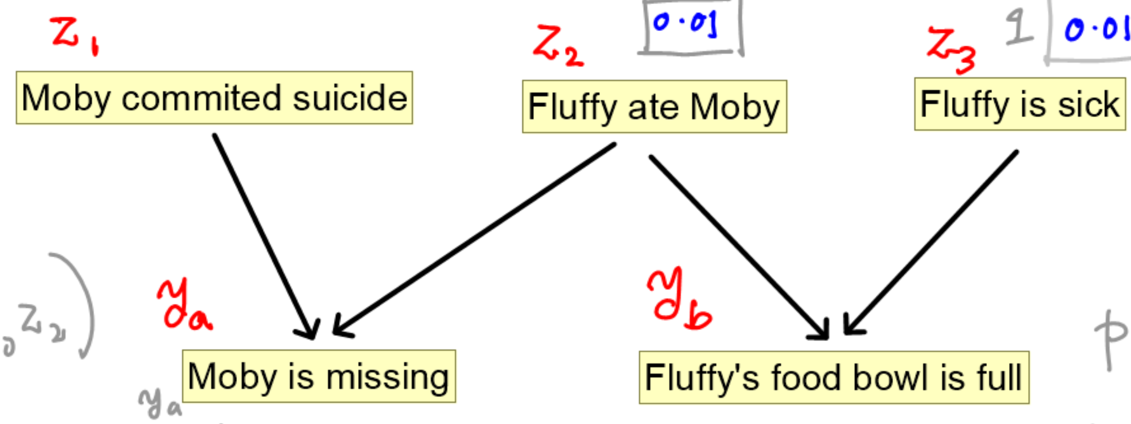
0.99
0.01

$p(z_2)$

0.99
0.01

$p(z_3)$

0	0.99
1	0.01



$p_\theta(y_a|z_1, z_2)$

z_1	z_2	y_a: 1	0
0	0	0.01	0.99
0	1	1	0
1	0	1	0
1	1	1	0

$p(y_b|z_2, z_3)$

z_2	z_3	y_b: 1	0
0	0	0.1	0.9
0	1	0.9	0.1
1	0	0.99	0.01
1	1	0.99	0.01

Given $y_a = 1, y_b = 0$, and $\boldsymbol{\theta}$, we wish to find:

1. Probability that "Moby committed suicide" i.e. $p(z_1|y_a, y_b)$.
2. Probability that "Fluffy ate Moby" i.e. $p(z_2|y_a, y_b)$.
3. Probability that "Fluffy is sick" i.e. $p(z_3|y_a, y_b)$.

There are the marginal posterior probability.

$$\underbrace{p(\underline{y}, \underline{z})}_{\text{Joint}} = \underbrace{p(\underline{y}/\underline{z})}_{\text{Lik}} \times \underbrace{p(\underline{z})}_{\text{Prior}} = \underbrace{p(\underline{z}/\underline{y})}_{\text{Post}} \times \underbrace{p(\underline{y})}_{\text{Marg. Lik}}$$

Posterior computation

Define $\mathbf{y} := \{y_a, y_b\}$. Using Bayes' rule, we can write the following:

$\underline{z} = \{z_1, z_2, z_3\}$

$$p(\mathbf{y}, z_1, z_2, z_3) = \underbrace{p(\mathbf{y}|z_1, z_2, z_3)}_{\text{Lik}}\underbrace{p(z_1, z_2, z_3)}_{\text{Prior}} = \underbrace{p(z_1, z_2, z_3|\mathbf{y})}_{\text{Post}}\underbrace{p(\mathbf{y})}_{\text{Marg. lik}}$$

$$p(z_1=1/\underline{y}) = \sum_{z_3=\{0,1\}} \sum_{z_2=\{0,1\}} p(z_1, z_2, z_3/\underline{y})$$

$$= p(z_1=1, z_2=0, z_3=0/\underline{y}) + p(z_1=1, z_2=1, z_3=0/\underline{y})$$

$$+ p(z_1=1, z_2=0, z_3=1/\underline{y}) + p(z_1=1, z_2=1, z_3=1/\underline{y})$$

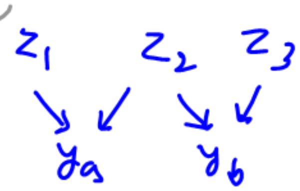
$$p(z_1=0/\underline{y}) =$$

To compute $p(z_1|y_a, y_b, y_c)$, we can marginalize z_2 and z_3.

$$p(z_1|\mathbf{y}) = \sum_{z_3=\{0,1\}} \sum_{z_2=\{0,1\}} \underbrace{p(z_1, z_2, z_3|\mathbf{y})}_{\text{Post}}$$

$$= \sum_{z_3=\{0,1\}} \sum_{z_2=\{0,1\}} \frac{\overbrace{p(\mathbf{y}|z_1, z_2, z_3)}^{\text{Lik}} \times \overbrace{p(z_1, z_2, z_3)}^{\text{Prior}}}{\underbrace{p(\mathbf{y})}_{\text{Marg. Lik} \leftarrow \text{constant}}}$$

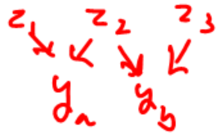
$$\propto \sum_{z_3=\{0,1\}} \sum_{z_2=\{0,1\}} \underbrace{p(\mathbf{y}|z_1, z_2, z_3)p(z_1, z_2, z_3)}_{p(\underline{y}, \underline{z}) \text{ Joint}}$$



Expanding we get the following:

$$p(z_1|\mathbf{y}) \propto \sum_{z_3} \sum_{z_2} p(y_a|z_1, z_2)p(y_b|z_2, z_3)p(z_1)p(z_2)p(z_3)$$

Flow chart of computation $p(z_1 = 1 | y_a = 1, y_b = 0)$



$p(y_a, y_b, z_1, z_2, z_3)$

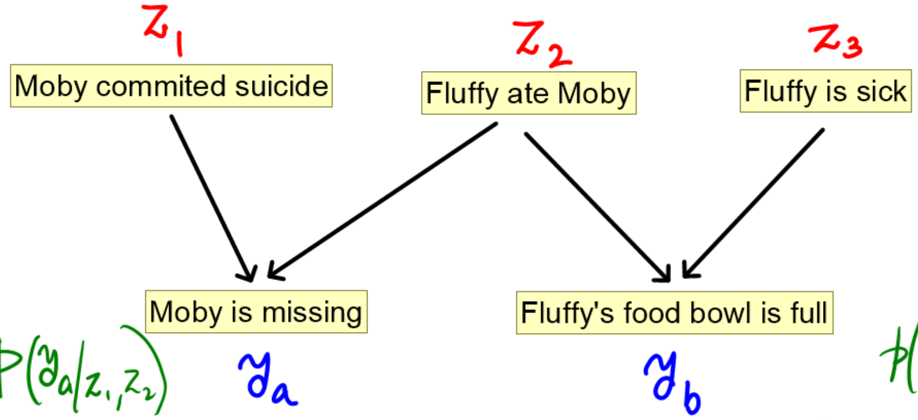
$y_a = 1$
$y_b = 0$
$z_1 = 1$

$p(z_1)$		$p(y_a \mid z_1, z_2)$		$p(z_2)$		$p(y_b \mid z_2, z_3)$		$p(z_3)$	Values of z_2	Values of z_3
0.01	×	1	×	0.99	×	0.9	×	0.99	0	0
0.01	×	1	×	0.99	×	0.1	×	0.01	0	1
	×		×		×		×		1	0
	×		×		×		×		1	1

+

Homework: Fill this.

$p(z_1) = \begin{bmatrix} 0.99 \\ 0.01 \end{bmatrix}$ $p(z_2) = \begin{bmatrix} 0.99 \\ 0.01 \end{bmatrix}$ $p(z_3) = \begin{bmatrix} 0.99 \\ 0.01 \end{bmatrix}$



$p(y_a | z_1, z_2)$

z_1	z_2	$y_a = 1$	$y_a = 0$
0	0	0.01	0.99
0	1	1	0
1	0	1	0
1	1	1	0

$p(y_b | z_2, z_3)$

z_2	z_3	$y_b = 1$	$y_b = 0$
0	0	0.1	0.9
0	1	0.9	0.1
1	0	0.99	0.01
1	1	0.99	0.01

What is the computational complexity for D unknowns?

$O(2^M E)$

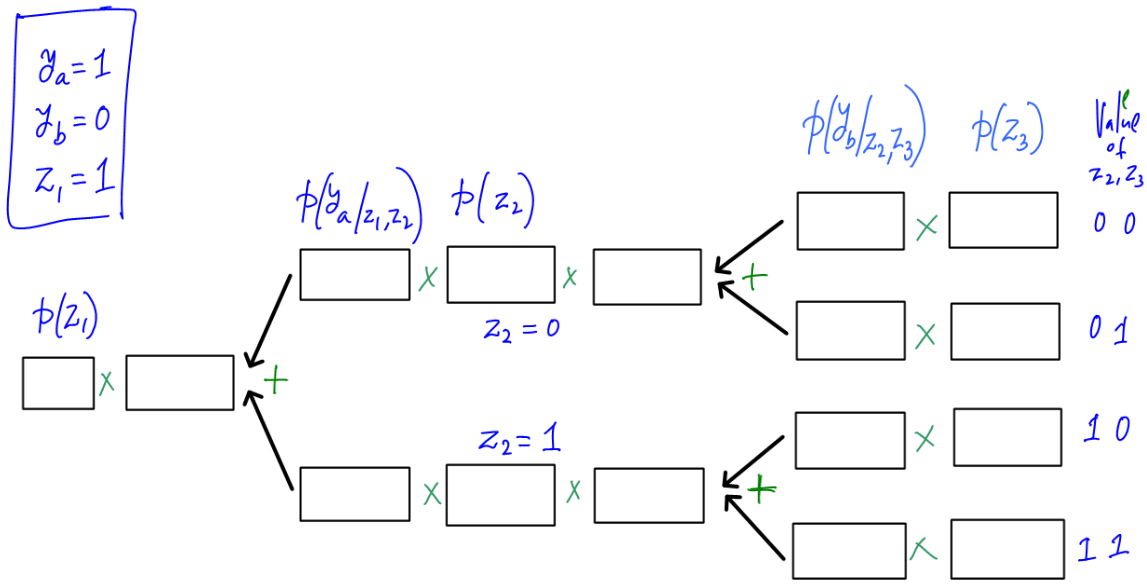
E = # Edges

Sum-product algorithm

We can reduce the complexity drastically by using the structure of the problem. We will use the sum-product algorithm.

$$\sum_{z_3}\sum_{z_2} p(y_a|z_1,z_2)p(y_b|z_2,z_3)p(z_1)p(z_2)p(z_3)$$
$$= p(z_1)\Big\{\sum_{z_2} p(y_a|z_1,z_2)p(z_2)\Big[\sum_{z_3} p(y_b|z_2,z_3)p(z_3)\Big]\Big\}$$

Flow chart of computation for $p(z_1 = 1|y_a = 1, y_b = 0)$



We can repeat the same procedure for z_2 and z_3.

$$p(z_1|\mathbf{y}) \propto p(z_1)\Big\{\sum_{z_2} p(y_a|z_1,z_2)p(z_2)\Big[\sum_{z_3} p(y_b|z_2,z_3)p(z_3)\Big]\Big\}$$
$$p(z_2|\mathbf{y}) \propto p(z_2)\Big\{\sum_{z_1} p(y_a|z_1,z_2)p(z_1)\Big[\sum_{z_3} p(y_b|z_2,z_3)p(z_3)\Big]\Big\}$$
$$p(z_3|\mathbf{y}) \propto p(z_3)\Big\{\sum_{z_2} p(y_b|z_2,z_3)p(z_2)\Big[\sum_{z_1} p(y_b|z_1,z_2)p(z_1)\Big]\Big\}$$

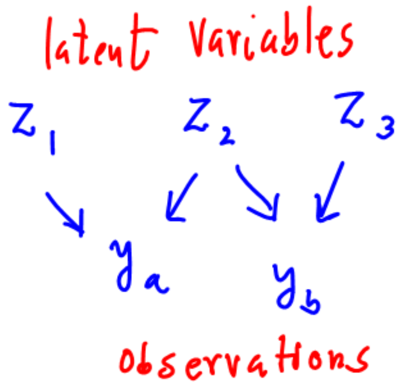
Notice that there are many redundant computations. We can reduce computation by reusing the redundant ones.

$$p(z_1|\mathbf{y}) \propto p(z_1) \underbrace{\sum_{z_2} p(y_a|z_1, z_2) \underbrace{p(z_2) \underbrace{\sum_{z_3} p(y_b|z_2, z_3) \underbrace{p(z_3)}_{m_{3\to b}(z_3)}}_{m_{b\to 2}(z_2)}}_{m_{2\to a}(z_2)}}_{m_{a\to 1}(z_1)}$$

(Summary)

Messages

Suppose, we have many latent variables z_i and many observations y_a. Also, assume that the graph between the observations and the variables is bi-partite. In this graph, neighbours of observations are variables, and vice-versa. Define $\mathbb{N}(i)$ to be the set of all neighbourhood of z_i and $\mathbb{N}(a)$ to be the set of all neighbours of observation y_a. Also, define the set $\mathbb{N}(i)\backslash a$ to be the set of "all neighbours of i except a".



Neighbors

$N(1) = \{a\}$ $\quad N(a) = \{1, 2\}$

$N(2) = \{a, b\}$ $\quad N(b) = \{2, 3\}$

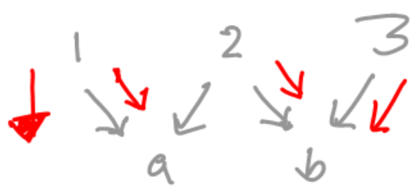
$N(3) = \{b\}$

$N(2)/a = \{b\}$

$N(1)/a = \{\ \}$

Define the messages from variables to observations as shown below:

$$m_{i\to a}(z_i) = p(z_i) \prod_{b\in\mathbb{N}(i)\backslash a} m_{b\to i}(z_i).$$



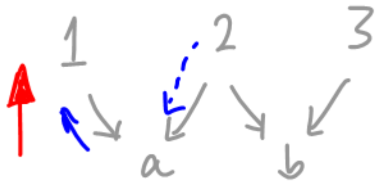
$m_{1\to a}(z_1) = p(z_1)$

$m_{3\to b}(z_3) = ?$

$m_{2\to a}(z_2) = p(z_2)\, m_{b\to 2}(z_2)$

$m_{2\to b}(z_2) = ?$

Therefore, in our example, messages from variables to observations are,



$$
\begin{aligned}
m_{1\to a}(z_1) &= p(z_1),\\
m_{2\to a}(z_2) &= p(z_2)m_{b\to 2}(z_2),\\
m_{2\to b}(z_2) &= p(z_2)m_{a\to 2}(z_2),\\
m_{3\to b}(z_3) &= p(z_3).
\end{aligned}
$$

$$m_{a\to 1}(z_1) = \sum_{z_2} p(y_a / z_1, z_2)\, m_{2\to a}(z_2)$$

$$m_{a\to 2}(z_2) = \;?$$

Similarly, the messages from observations to variables are shown below:

Sum over all values of all

$$m_{a\to i}(z_i) = \sum_{j \neq i} p(y_a|\mathrm{Pa}_a) \prod_{j\in\mathbb{N}(a)\setminus i} m_{j\to a}(z_j).$$

$z_j \neq z_i \longrightarrow$

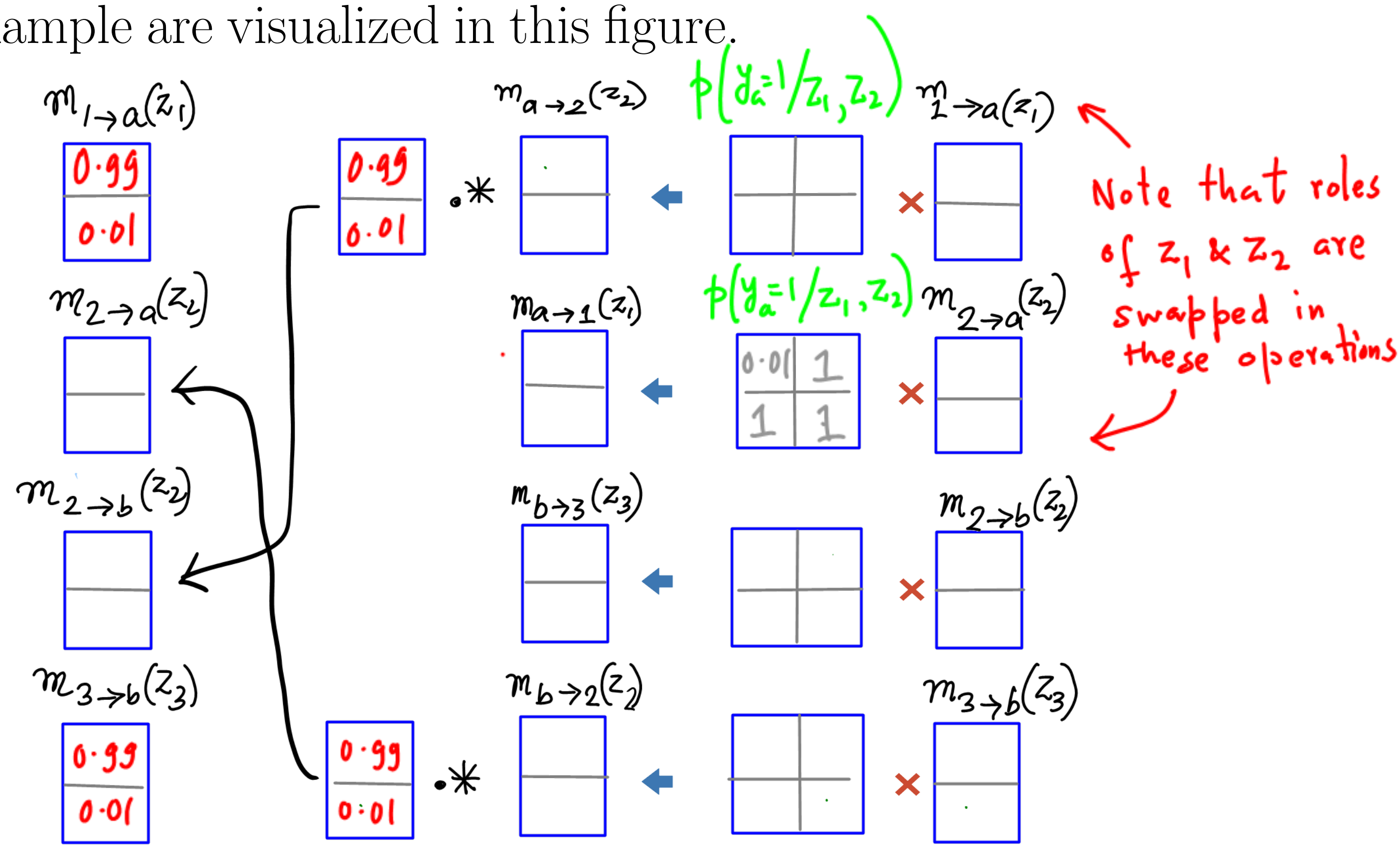
In our example, these can be written as shown below:

$$
\begin{aligned}
m_{a\to 1}(z_1) &= \sum_{z_2} p(y_a|z_1, z_2)m_{2\to a}(z_2),\\
m_{a\to 2}(z_2) &= \sum_{z_1} p(y_a|z_1, z_2)m_{1\to a}(z_1),\\
m_{b\to 2}(z_2) &= \sum_{z_3} p(y_b|z_2, z_3)m_{3\to b}(z_3),\\
m_{b\to 3}(z_3) &= \sum_{z_2} p(y_b|z_2, z_3)m_{2\to b}(z_2).
\end{aligned}
$$

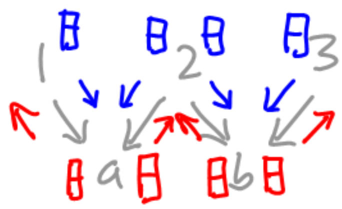
The first set of messages perform a product operation, while the second set performs the sum.

Assume $y_a=1,\ y_b=0$.

Calculation of these messages for our example are visualized in this figure.



Variables → Observations



$$m_{1\to a}(z_1) = p(z_1),$$
$$m_{2\to a}(z_2) = p(z_2)m_{b\to 2}(z_2),$$
$$m_{2\to b}(z_2) = p(z_2)m_{a\to 2}(z_2),$$
$$\rightarrow m_{3\to b}(z_3) = p(z_3).$$

$$m_{a\to 1}(z_1) = \sum_{z_2} p(y_a|z_1,z_2)m_{2\to a}(z_2),$$
$$m_{a\to 2}(z_2) = \sum_{z_1} p(\underline{y_a|z_1,}\, z_2)m_{1\to a}(z_1),$$
$$m_{b\to 2}(z_2) = \sum_{z_3} p(y_b|z_2,z_3)m_{3\to b}(z_3),$$
$$m_{b\to 3}(z_3) = \sum_{z_2} p(y_b|z_2,z_3)m_{2\to b}(z_2).$$

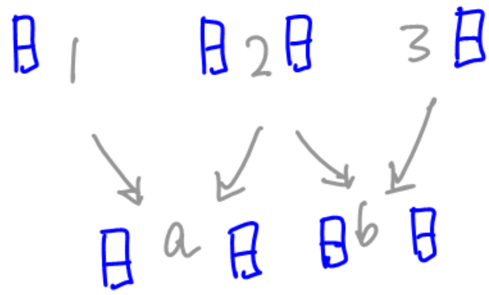
Belief Propagation

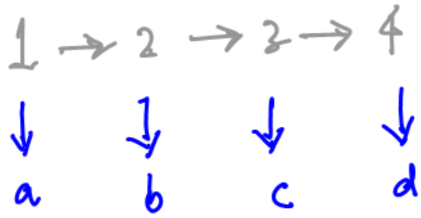
Here is a Belief propagation algorithm for our example.

1. Initialize messages of variables i to $p(z_i)$ and iterate until messages do not change:
   (a) Send messages from variables to observations.
   (b) Send messages from observations to variables.

2. Compute marginals by multiplying all the message received at node i.

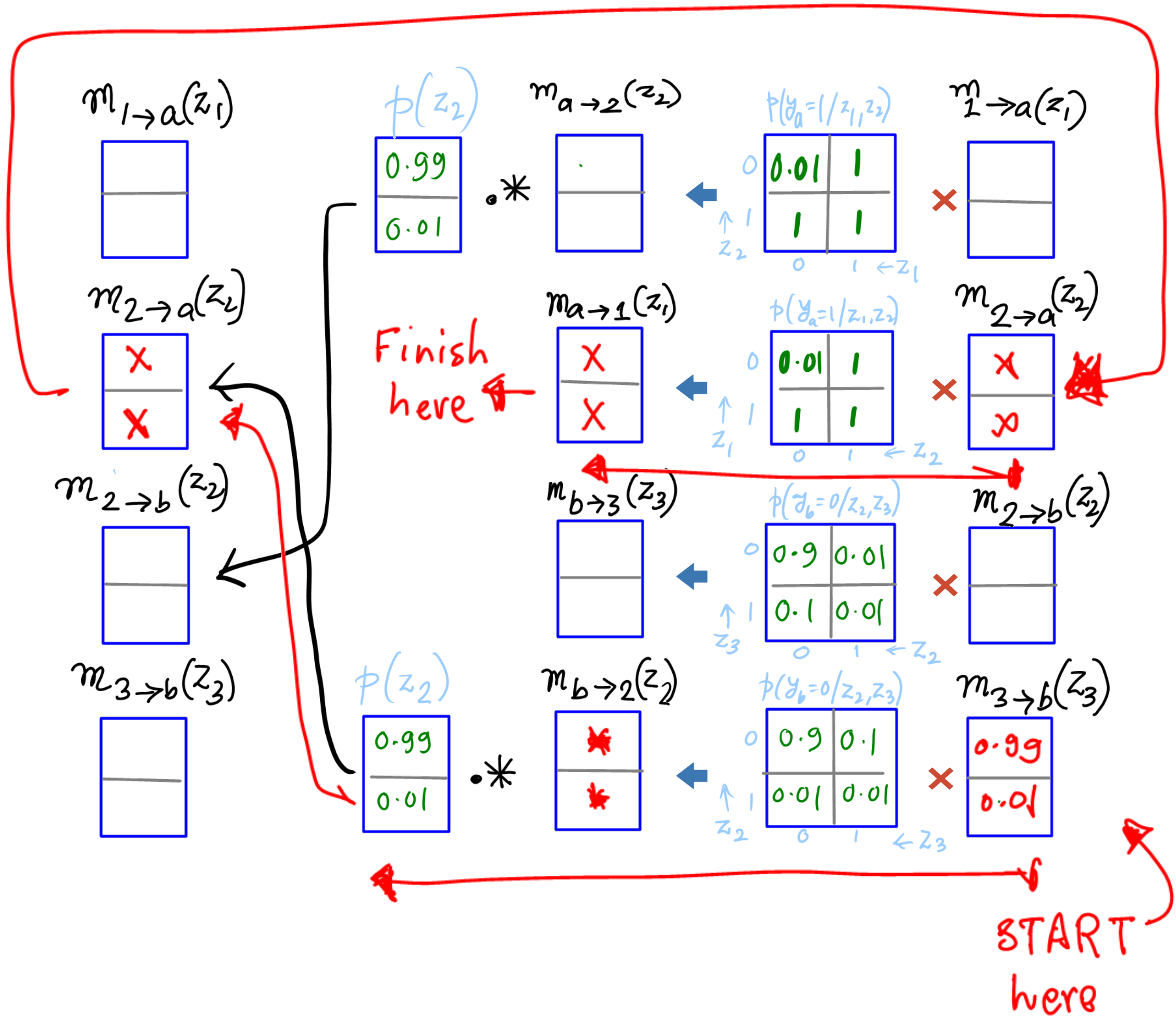
$$p(z_i|\mathbf{y}) = p(z_j) \prod_{j \in \mathbb{N}(a)} m_{j \to a}(z_j)$$





Why will this work? See the path $1 \leftarrow a \leftarrow 2 \leftarrow b \leftarrow 3$.

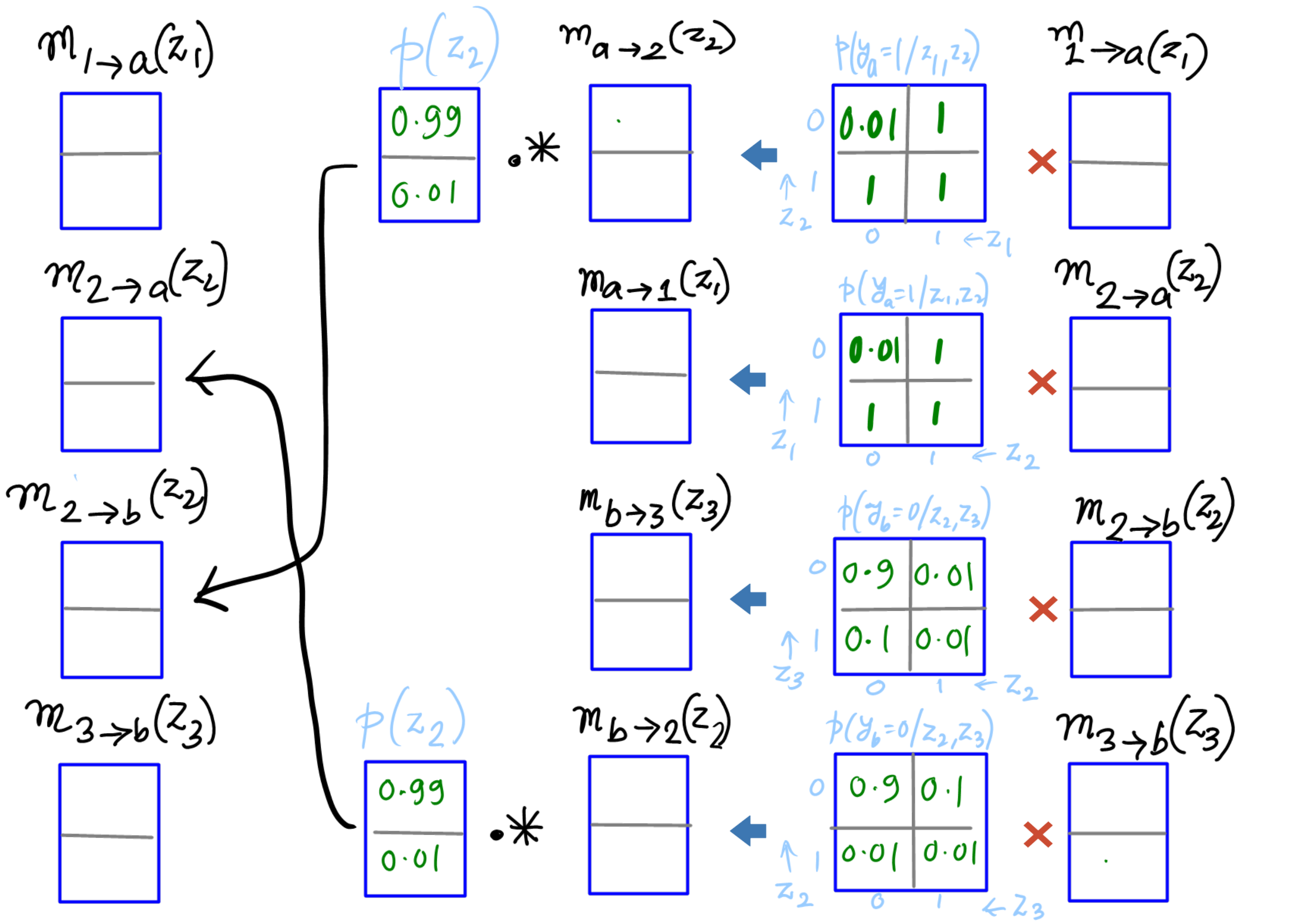
$$p(z_1|\mathbf{y}) \propto p(z_1) \underbrace{\sum_{z_2} p(y_a|z_1,z_2) \underbrace{p(z_2) \underbrace{\sum_{z_3} p(y_b|z_2,z_3) \underbrace{p(z_3)}_{m_{3\to b}(z_3)}}_{m_{b\to 2}(z_2)}}_{m_{2\to a}(z_2)}}_{m_{a\to 1}(z_1)}$$



Homework

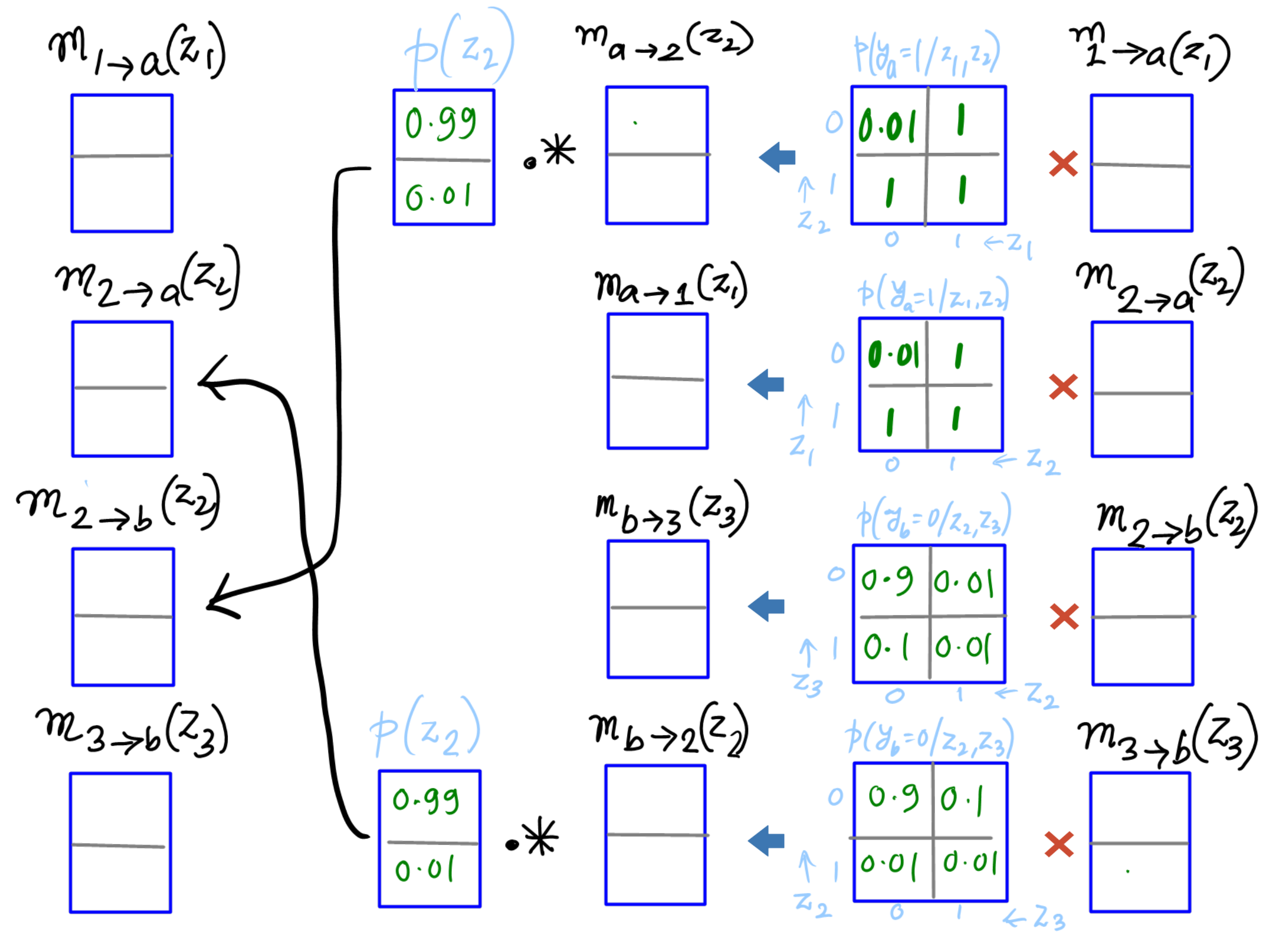
See the path $1 \to a \to 2 \leftarrow b \leftarrow 3$.

$$p(z_2|\mathbf{y}) \propto p(z_2) \underbrace{\sum_{z_1} p(y_a|z_1, z_2) \underbrace{p(z_1)}_{m_{1\to a}(z_1)}}_{m_{a\to 2}(z_2)} \underbrace{\sum_{z_3} p(y_b|z_2, z_3) \underbrace{p(z_3)}_{m_{3\to b}(z_3)}}_{m_{b\to 2}(z_2)}$$



Homework:

Exercise: Work out the example to compute $p(z_3)$.



Homework! Set $y_a = y_b = 1$, Compute all marginals.

Discussion

A general graphical model

Probabilistic graphical models also contain other models such undirected graphical models and factor graphs. Belief propagation can be applied to these models as well. The main advantage of this method is that it can be implemented using distributed computation and is very efficient for large ~~scale~~ models.