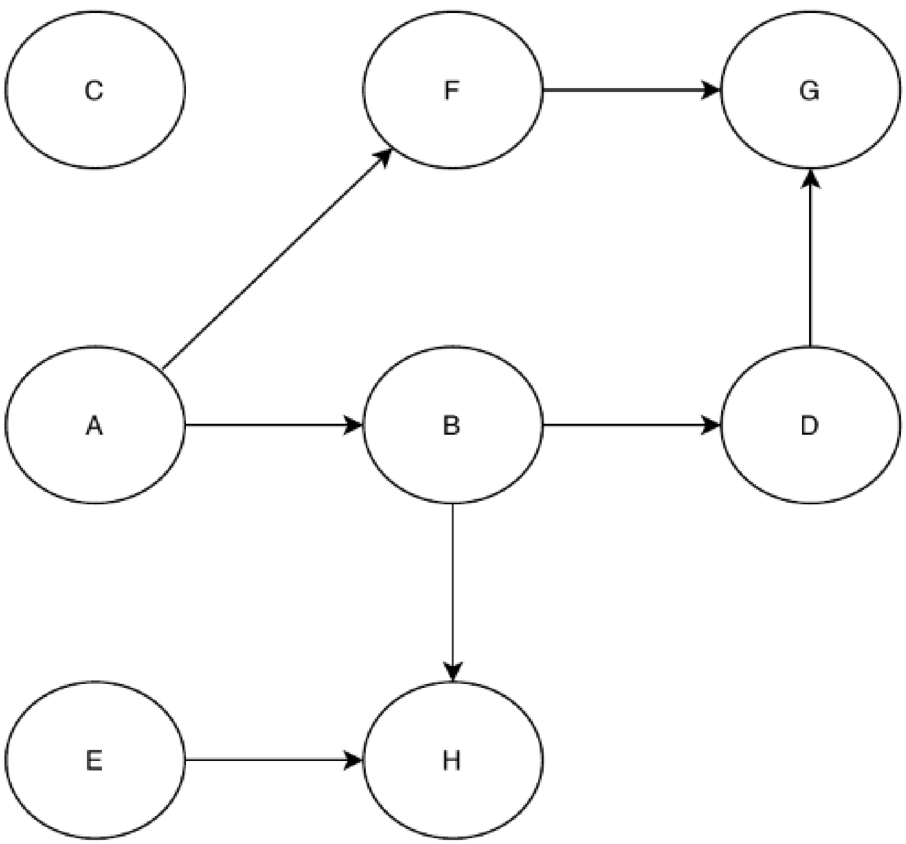
homework_03

April 10, 2020

0.0.1 Exercise 1

1. Draw the Bayesian Network representing the joint distribution

$$P(A, B, C, D, E, F, G, H) = P(A)P(B|A)P(C)P(D|B)P(E)P(F|A)P(G|D, F)P(H|E, B).$$



By considering, two nodes A and B in a directed graph are **conditionally independent** given a node C if and only if

$$p(A, B|C) = p(A|C)p(B|C).$$

2. Indicate whether the following statements on conditional independence are True or False and motivate your answer.

a. $A \perp\!\!\!\perp B$

- False, it is obvious that A and B are not conditionally independent and $P(A, B) = P(A).P(B|A)$

b. $A \perp\!\!\!\perp C$

- True, A and C has no relation and $P(A, C) = P(A)P(C)$

c. $A \perp\!\!\!\perp D|\{B, H\}$

- True, because $P(A, B, C, D) = P(A)P(D|B)P(B|A)P(H)$ imply that $P(A|\{B, H\})P(D|\{B, H\})$

d. $A \perp\!\!\!\perp E|F$

- True, because $P(A, E, F) = P(A)P(E)P(F|A)$ implies that $P(A, E|F) = P(A|F)P(E|F)$

e. $G \perp\!\!\!\perp E|B$

- True, because $P(G, E, B) = P(G)P(B)P(E)$

f. $F \perp\!\!\!\perp C|D$

- True, because $P(F, C, D) = P(F)P(C)P(D)$

g. $E \perp\!\!\!\perp D|B$

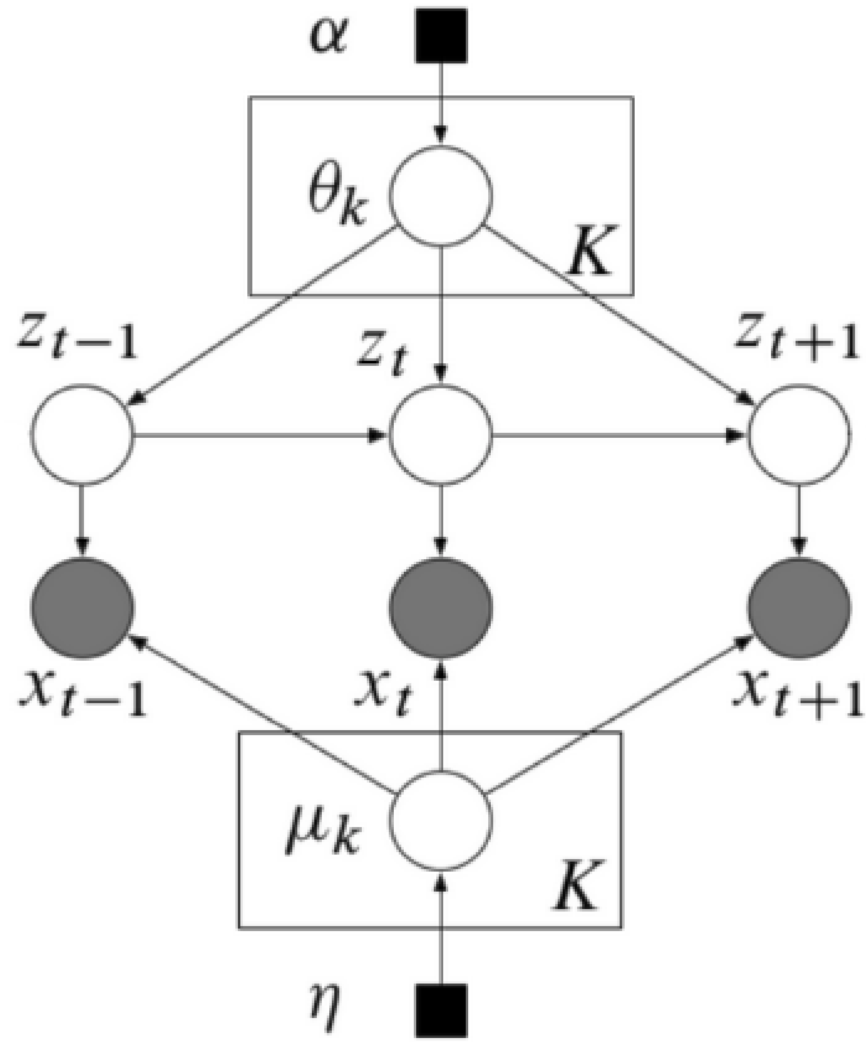
- True, becasue $P(E, D, B) = P(E)P(D|B)P(B)$ implies that $P(E, D|B) = P(E|B)P(D|B)$

h. $C \perp\!\!\!\perp H|G$

- True, because $P(C, H, G) = P(C)P(H)P(G)$

0.0.2 Exercise 2

- Build the generative model corresponding to the directed graph



Generative model

- $\theta \sim Dirichlet(\alpha)$
- $\mu_k \sim \mathcal{N}(0, \eta^2)$ for the mixture components
- for each data point t:
- $z_0|\theta \sim Categorical(\theta)$
- $z_t|\theta, z_{t-1} \sim Categorical(\theta, z_{t-1})$
- $x_t|z_t, \mu_{k_t} \sim \mathcal{N}(\mu_{k_t}, 1)$

where θ, μ_k, z_t are the **hidden variables**, x_i the **observables** and α, η the fixed **hyperparameters**.

The joint distribution factorizes as:

$$p(\theta, \mu, z, x|\eta, \alpha) = \prod_{k=1}^{K} p(\theta_t|\alpha)p(\mu_k|\eta) \prod_{i=1}^{N} [p(z_t|\theta, z_{t-1})p(x_t|z_t, \mu_k)].$$

where N is the number of observation and $z_0 \sim Categorical(\theta)$, (θ chosen uniformly)

From this we can define the posterior distribution as:

$$p(\theta, \mu, z|x, \eta, \alpha) = \frac{p(\theta, \mu, z, x|\eta, \alpha)}{p(x|\eta, \alpha)}.$$

- Using Dirichlet, Categorical and Normal distributions and supposing that $K = 2$. Then, write a `pyro` implementation of the resulting model.

[9]:
```
import pyro
import torch
import pyro.distributions as dist
import random as rnd

# Number of components
K = 2

#Hyperparameters

alpha = 0.7
eta = 5
idx = rnd.randint(0,1) # random index that will help to sample first z (z_f) to␣
↪choose theta parameter uniformly

def model(data):
    N = len(data)

    with pyro.plate('hidden_variable', K):
        theta = pyro.sample('theta', dist.Dirichlet(alpha * torch.ones(K)))

    with pyro.plate('components', K):
        mu = pyro.sample("mu", dist.Normal(0., eta))

    # list that will be used for storing z values
    z = list()
    for i in pyro.plate("data", N):
        if i == 0:
            # first z, so theta parameter is chosen uniformly
            z_f = pyro.sample("z",dist.Categorical(probs = theta[idx]))
            z.append(z_f)
        else:
            # z_r which depends on previous z values (z_f)
            z_r = pyro.sample('z', dist.Categorical(probs = theta[z_f]))
            z.append(z_r)

            # z_f are updated to make z dependent to previous ones
            z_f = z_r
```

```python
        # sampling x, dependent to z and mu
        x = pyro.sample("x", dist.Normal(mu[z_f],1),obs= data)
    # bringing all z's to one place
    z = torch.stack(tuple(z),0)

    print("theta =",theta,"\nmu =",mu,"\nz =", z,"\nx =",  x)


model(data = [5.3,2.4,3.5,6.1,1.2,2.6])
```

```
theta = tensor([[0.0121, 0.9879],
        [0.2809, 0.7191]])
mu = tensor([ 7.3445, -0.3565])
z = tensor([0, 1, 1, 1, 1, 1])
x = [5.3, 2.4, 3.5, 6.1, 1.2, 2.6]
```