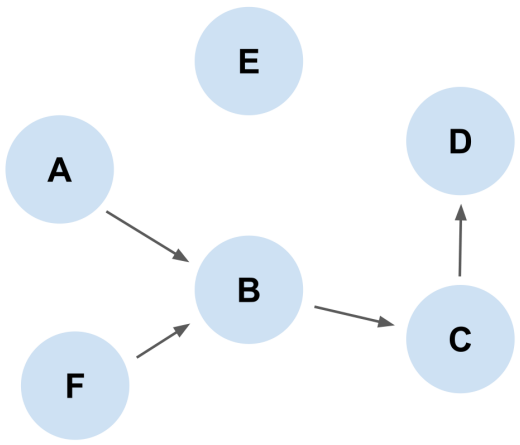
Homework 03

Exercise 1

1. Write the factorization of the joint distribution $P(A, B, C, D, E, F)$ corresponding to the Bayesian network:



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

a. A ⫫ B

b. A ⫫ F

c. A ⫫ C | {B,E}

d. F ⫫ D | B

e. B ⫫ D | C

Exercise 2

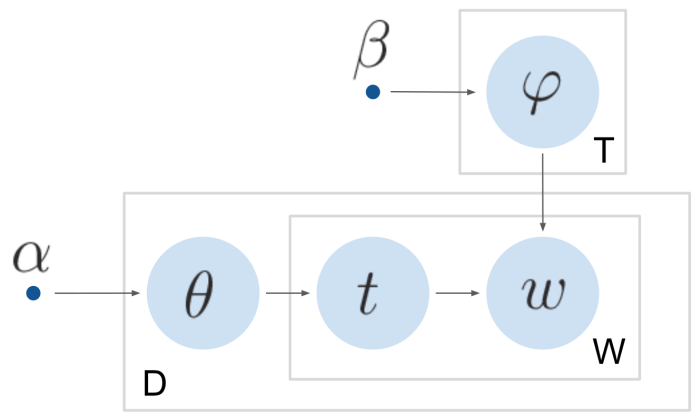
Topic models are statistical models that learn the distribution of the abstract topics occurring in a collection of documents. In this context, documents are collections of topics and topics are collections of words.

We consider a dictionary of N words, an ordered collection of D documents and an ordered collection of T possible topics appearing in each document. We build our model using the indexes representing each quantity in its collection (e.g. i is the i-th word in the dictionary). For simplicity, we assume that all documents contain the same number of words. Each document can contain multiple topics; specifically, we associate a topic to each word appearing in the document.

We make the following assumptions:

- D is the total number of documents
- T is the total number of topics
- $W < N$ is the number of words per document
- For the j-th word and the i-th document, we sample a topic $t_{i,j}$ from a Categorical distribution with parameters θ_i
- The distribution θ_i of topics in the i-th document is a Dirichlet distribution with concentration parameters α (the length of α is T)
- The j-th word in the i-th document, namely $w_{i,j}$, is sampled from a Categorical distribution with parameters $\varphi_{t_{i,j}}$
- The distribution φ_k of words per topic k is a Dirichlet distribution with concentration parameters β (the length of β is N)

1. Write the generative process of the following graphical model. Be aware of plate notation!



2. Set $T = 5$, $D = 10$, $W = 50$, $N = 100$ and use pyro to implement this graphical model as a function of the hyperparameters `model(alpha, beta)` , that outputs `theta, phi, t, w` .
3. Evaluate your function on the hyperparameters

```
alpha = torch.tensor([0.5, 0.3, 0.1, 0.4, 0.2])
beta = dist.Gamma(1./T, 1.).sample((N,))
```

and print the shape of the output tensors `theta, phi, t, w` .