

# Introduction to Probabilistic Graphical Models

## Directed Graphical Models (or Bayesian Networks)

Wilker Aziz

Self-study material for NTMI/NLP1 – 2023/24 edition

# Outline and goals

This self-study class is an introduction to probabilistic graphical models (PGMs), in particular, directed graphical models (also known as Bayesian networks).<sup>1</sup>

We will motivate the topic from an NLP point of view, then discuss the topic in general terms.

**ILOs**<sup>2</sup> After this class the student

- ▶ understands the idea behind factorisation of probabilities;
- ▶ recognises a factorisation expressed graphically;
- ▶ can re-express probability queries using chain rule, conditional probability, and marginalisation.

---

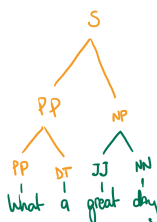
<sup>1</sup>To go far beyond of what we cover, check Koller and Friedman (2009).

<sup>2</sup>If, while working through this class, you find that you need a recap of probability theory, check, for example, my lecture notes <https://wilkeraziz.github.io/assets/pdfs/lecture-notes-appendix-B.pdf> or this Jupyter notebook <https://colab.research.google.com/github/probabll/ntmi-tutorials/blob/main/Discrete-Distributions.ipynb> or any undergraduate text on probability and statistics.

# The problem

In NLP we design probability models involving structured data such as documents, trees and graphs.

Typically, the sample spaces we care about are **too large** (e.g., the number of possible syntactic trees for a sentence grows exponentially with sentence length)



...

# The problem

In NLP we design probability models involving structured data such as documents, trees and graphs.

Typically, the sample spaces we care about are **infinite** (e.g., there is no obvious limit to how many different ways we could write about our sentiment)



# The problem

In NLP we design probability models involving structured data such as documents, trees and graphs.

Typically, the sample spaces we care about are

- ▶ too large (e.g., the number of possible syntactic trees for a sentence grows exponentially with sentence length)
- ▶ or infinite (e.g., there is no obvious limit to how many different ways we could write about our sentiment)

and, as a consequence, we cannot represent joint probability distributions over these spaces by simply storing one probability value for each possible outcome.

Not only we might not be able to store those, we probably cannot estimate all necessary values from finitely many observations.

# A solution

Instead, what we do is we use our knowledge of the application domain to motivate simplifying assumptions that make our distributions simpler/feasible to specify.

The key to our solution is that we will be computing those probabilities by manipulating a finite (and hopefully small) number of *elementary* probability factors.

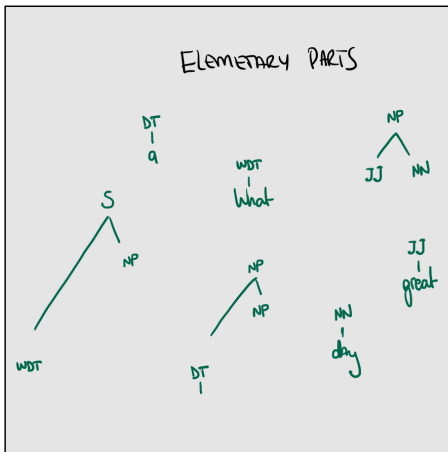
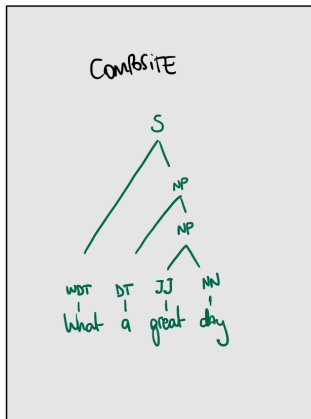
## Factorisation (intuition)

Remember when you learnt how to factorise natural numbers using prime numbers? For example, 24 is  $2 \times 2 \times 2 \times 3$ .

The idea is very similar (not identical, but similar enough to give you an intuition). Factorising into primes allows us to express all composite (i.e., non-prime) numbers by multiplying together some prime numbers. For any large but finite subset  $[1, M] \in \mathbb{N}$  we consider, there are far fewer prime numbers than composite numbers in it. When we see a number in this set, and it is not a prime number, we can think of it as if we were seeing a product-composition of primes, and *factorisation into primes* reveals that construct.

# Factorisation (examples)

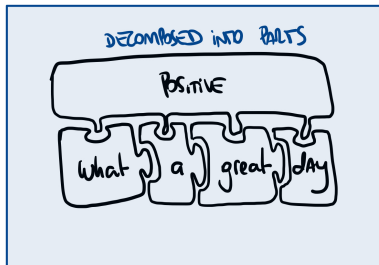
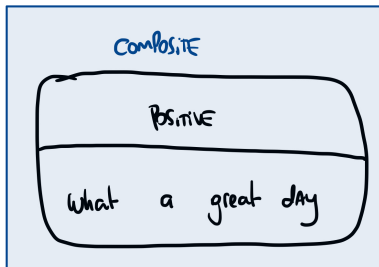
We decompose the probability of something which is 'complex' (or at least 'decomposable into parts', e.g., a syntactic tree) into a product of probabilities of simpler things (e.g., context-free tree fragments).





## Factorisation (examples)

We decompose the probability of something which is 'complex' (or at least 'decomposable into parts', e.g., a labelled document) into a product of probabilities of simpler things (e.g., labelled word pairs).



# Notation

We use uppercase letters to denote random variables (rvs for short), e.g.,  $X, Y$ . Lowercase letters (e.g.,  $x, y$ ) are then used for outcomes. The range of an rv  $X$  is denoted by a calligraphic letter  $\mathcal{X} \subseteq \mathbb{R}$ . The underlying sample space of  $X$  is denoted by  $\Omega_X$ . An assignment  $X = x$ , with  $x \in \mathcal{X}$  is an event (i.e., a subset of  $\Omega_X$ ), we pronounce this as “the rv  $X$  takes on the value  $x$  in its range”. For convenience, we do not really distinguish the sample space and the range, esp when dealing with discrete rvs, we just assume the reader can imagine some fixed mapping from outcomes in  $\Omega_X$  to a subset of  $\mathbb{R}$ , such as an enumeration.

We use  $P_X(x)$  to denote the probability  $P(X = x)$ , and, similarly with multiple variables:  $P_{AB|C}(a, b|c) = P_{BA|C}(b, a|c) = P(A = a, B = b|C = c) = P(B = b, A = a|C = c)$ . When we refer to the distribution of  $X$ , we often use  $P_X$ ; then  $P_{XY}$  is the distribution of the pair of rvs  $(X, Y)$ ; and  $P_{Y|X=x}$  is the conditional probability distribution (cpd) of the rv  $Y$  given the assignment  $X = x$ . The notation  $P_{Y|X}$  then refers to the collection of all cpds of the kind  $P_{Y|X=x}$  for any possible outcome  $x \in \mathcal{X}$  of  $X$ .

## Factorising via chain rule

To factorise is to break into parts, to factorise a joint probability like  $P_{AB}(a, b)$  we can use chain rule and decompose it like this

$$P_{AB}(a, b) = P_A(a)P_{B|A}(b|a)$$

or like that

$$P_{AB}(a, b) = P_B(b)P_{A|B}(a|b)$$

We may prefer to work with one version or the other, but they are *both* valid, and we are going to learn how to go from one to the other whenever needed.

## Factorising via chain rule: more variables

Chain rule works with more variables too.

$$P_{ABC}(a, b, c) = P_A(a)P_{B|A}(b|a)P_{C|AB}(c|a, b)$$

or

$$P_{ABC}(a, b, c) = P_B(b)P_{C|B}(c|b)P_{A|BC}(a|b, c)$$

or ... (you can guess other ways to factorise it).

You might have noticed one thing: our probability factors are growing in complexity, they start simple (like  $P_A(a)$  or  $P_B(b)$ ) but *eventually* involve **all** rvs (like  $P_{C|AB}(c|a, b)$ ).

## Conditional independence

To factorise a joint probability like  $P_{AB}(a, b)$  into parts that are 'simpler' (i.e., involve fewer rvs) such as  $P_A(a) \times P_B(b)$  we need to assume statistical independence of  $A$  and  $B$ .

When independence is 'licensed' in a specific context, we say it is a conditional independence. For example: if

$P_{AB|C}(a, b|c) = P_{A|C}(a|c)P_{B|C}(b|c)$  we have assumed that  $A$  is independent of  $B$  given  $C$ .<sup>3</sup>

Examples: i) to say that one's grade in this course is independent of the weather in November; ii) to say that the number of students and the identity of the teaching staff are independent of one another given the course syllabus; iii) to say that in an auto-complete application, the next word depends on the current word, but not on the one before that; and many other examples.

---

<sup>3</sup>The notation  $A \perp B \mid C$  is another way to express this conditional independence.

## Tractability in mind

Some (maybe most) conditional independences are not very realistic, rather they are needed for feasibility.

Regardless of which probability factors we represent directly (i.e., we store and estimate), together they induce a joint distribution over all variables, and probability calculus tells us how we can recombine the elementary probability factors we have to answer *any* probability query for those variables.

So, while we will be storing/representing only the elementary factors, we will be able to—on demand—spend some computation to obtain any other probability we may ever need.

# Applications

The ability to factorise and answer probability queries has major applications in NLP and all of machine learning.

For example, we will soon be looking into answering conditional probability queries about a class  $Y = y$  given a large document  $X = x$ . As we shall see, it is not viable to store a table of probabilities for every pair  $(x, y)$ , but, by exploiting a certain factorisation, we will be able to re-express probability queries about  $P_{Y|X=x}$ , for practically any one  $y$  given  $X = x$ , in terms of probabilities stored in relatively small tables. For that we will need knowledge of conditional independence and some probability calculus.

The *framework* of choice to express these factorisations is that of *probabilistic graphical models* (PGMs).

# Probabilistic Graphical Models

PGMs are a way to specify probability distributions over complex sample spaces.

In this class we will concentrate on **discrete** random variables.

Sometimes the joint sample space of all variables we care about grows combinatorially or is infinite or is simply too large for us to represent a joint distribution without simplifications.

PGMs give us a language to precisely encode these simplifications.

Let's start by seeing the *tabular representation* of a joint distribution without any independence assumptions.



## Tabular representation

Suppose  $A$ ,  $B$ , and  $C$  are binary random variables (rvs). How do we represent a joint distribution  $P_{A,B,C}$  without making independence assumptions?

## Tabular representation

Suppose  $A$ ,  $B$ , and  $C$  are binary random variables (rvs). How do we represent a joint distribution  $P_{A,B,C}$  without making independence assumptions?

We create a table that lists all joint assignments of the rvs and their probability values. Probabilities are constrained to be between 0 and 1, and the sum of all of these must be 1:

Joint assignments			Probability values
$a$	$b$	$c$	$P_{ABC}(a, b, c)$
0	0	0	$P_{ABC}(0, 0, 0)$
0	0	1	$P_{ABC}(0, 0, 1)$
0	1	0	$P_{ABC}(0, 1, 0)$
1	0	0	$P_{ABC}(1, 0, 0)$
0	1	1	$P_{ABC}(0, 1, 1)$
1	1	0	$P_{ABC}(1, 1, 0)$
1	0	1	$P_{ABC}(1, 0, 1)$
1	1	1	$P_{ABC}(1, 1, 1)$

**Table:** Tabular joint distribution over 3 binary rvs

## Watch out!

A different question is **where do the probability values come from?**

But that is not a question of how to 'represent the distribution' it is a question of how to obtain numerical values for the probabilities in the tabular representation.

The challenges in representing the distribution in this format are inherent to the sample space being so big, no matter whether these numerical values are fixed by hand or estimated by a computer programme from data.

## Exercise

How many probability values does it take to represent a joint distribution over  $n$  random variables, where each one can take on 1 of  $K$  values, using a table as we did before while making no independence assumptions?

## Exercise

How many probability values does it take to represent a joint distribution over  $n$  random variables, where each one can take on 1 of  $K$  values, using a table as we did before while making no independence assumptions?

It takes  $K^n$  probability values

## Exercise

How many probability values does it take to represent a joint distribution over  $n$  random variables, where each one can take on 1 of  $K$  values, using a table as we did before while making no independence assumptions?

It takes  $K^n$  probability values

Suppose  $n$  is the length of a document (say some 30 words) and  $K$  is the number of words in English (say 100000). How bad is this?

## Exercise

How many probability values does it take to represent a joint distribution over  $n$  random variables, where each one can take on 1 of  $K$  values, using a table as we did before while making no independence assumptions?

It takes  $K^n$  probability values

Suppose  $n$  is the length of a document (say some 30 words) and  $K$  is the number of words in English (say 100000). How bad is this?

Bad!  $(10^5)^{30} = 10^{150}$

# Graphs

Now that we agree that a tabular representation of a joint distribution can get ridiculously large, we are going to talk about a way to implicitly express these large objects using smaller ones. For that we will need a bit of graphs and a bit of probability calculus.



# Directed graphical models or Bayesian networks (BNs)

A BN is a directed acyclic graph (DAG):

- ▶ nodes represent rvs  
(content of node is the assignment)
- ▶ edges represent direct dependence
- ▶ there are no **directed cycles**

DAGs encode a set of conditional independence statements: an rv is conditionally independent of its **non-descendants** given its **parents**.<sup>a</sup>

In the example DAG, descendants of  $C$  are  $\{D, F\}$ , non-descendants of  $C$  are  $\{A, B, E\}$ , parents of  $C$  are  $\{B, E\}$ .

---

<sup>a</sup>Descendants of  $X$ : nodes reachable by paths that begin at  $X$ .  
Non-descendants: all nodes except  $X$  and its descendants. Parents: non-descendants directly connected to  $X$ .

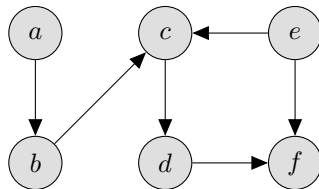


Figure: DAG

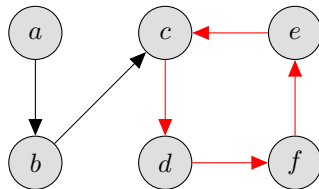


Figure: Not a DAG

## Exercise: complete the table of relationships

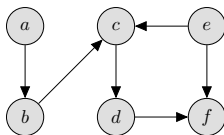


Figure: DAG

Node	Descendants	Non-descendants	Parents
<i>A</i>			
<i>B</i>			
<i>C</i>			
<i>D</i>			
<i>E</i>			
<i>F</i>			

Table: Relationships

## Exercise: complete the table of relationships

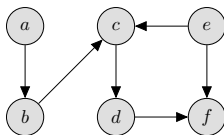


Figure: DAG

Node	Descendants	Non-descendants	Parents
<i>A</i>	{ <i>B, C, D, F</i> }	{ <i>E</i> }	$\emptyset$
<i>B</i>	{ <i>C, D, F</i> }	{ <i>A, E</i> }	{ <i>A</i> }
<i>C</i>	{ <i>D, F</i> }	{ <i>A, B, E</i> }	{ <i>B, E</i> }
<i>D</i>	{ <i>F</i> }	{ <i>A, B, C, E</i> }	{ <i>C</i> }
<i>E</i>	{ <i>C, D, F</i> }	{ <i>A, B</i> }	$\emptyset$
<i>F</i>	$\emptyset$	{ <i>A, B, C, D, E</i> }	{ <i>D, E</i> }

Table: Relationships

## Examples of BNs

Assumptions: these are variables that matter, they depend on one another as shown in the DAG.

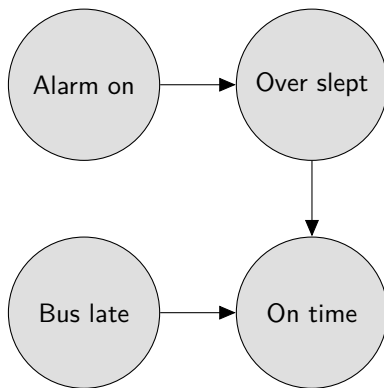


Figure: Students being on time

## Examples of BNs

Assumptions: these are variables that matter, they depend on one another as shown in the DAG.

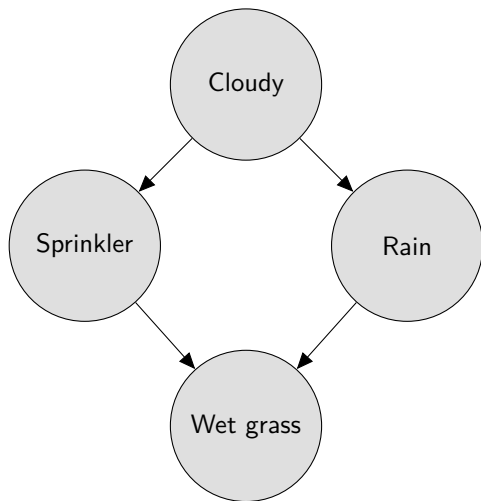


Figure: Wet grass on campus

## Conditional independence in BNs

Consider  $A$ ,  $B$ , and  $C$ , due to chain rule we can write

$$P_{A,B,C}(a, b, c) = P_A(a)P_{B|A}(b|a)P_{C|AB}(c|a, b) \quad (1)$$

and recall, we can use any other order.

But if we are told the assumptions in this DAG hold

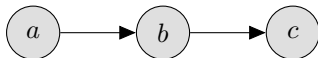


Figure: BN  $\mathcal{G}$

then we can simplify it

$$P_{A,B,C}(a, b, c) = P_A(a)P_{B|A}(b|a)P_{C|AB}(c|a, b) \quad (2)$$

$$\stackrel{\mathcal{G}}{=} P_A(a)P_{B|A}(b|a)P_{C|B}(c|b) \quad (3)$$

The last equality holds under the assumptions expressed in  $\mathcal{G}$ , namely, that  $C$  is independent of non-descendants  $\{A\}$  given its parents  $\{B\}$

# Chain rule for Bayesian networks

Chain rule (in general)

$$P_{X_1, \dots, X_m}(x_1, \dots, x_m) = \prod_{i=1}^m P_{X_i | X_{<i}}(x_i | x_{<i}) \quad (4)$$


Chain rule for Bayesian networks

$$P_{X_1, \dots, X_m}(x_1, \dots, x_m) = \prod_{i=1}^m P_{X_i | \text{Pa}_{X_i}}(x_i | \text{pa}_{x_i}) \quad (5)$$

where

- ▶  $X_{<i}$  is the sequence of rvs up until but not including  $X_i$ , and  $x_{<i}$  is its assignment
- ▶  $\text{Pa}_X$  set of rvs parents of  $X$
- ▶  $\text{pa}_x$  assignments of parents of  $X = x$

## Representing BNs


For each random variable, conditioned on assignments of its parents in the BN, we need a conditional probability distribution (CPD), which we represent in tabular form. Thus for binary rvs  , we have

- ▶ 1 cpd over  $A$ :  $P_A$  (it 'conditions' on nothing)
- ▶ 2 cpds over  $B \mid A$ :  $P_{B|A=0}$  and  $P_{B|A=1}$
- ▶ 2 cpds over  $C \mid B$ :  $P_{C|B=0}$  and  $P_{C|B=1}$

Exercise: for this BN, list all CPDs in tabular form:



## Representing BNs

For each random variable, conditioned on assignments of its parents in the BN, we need a conditional probability distribution (CPD), which we represent in tabular form. Thus for binary rvs , we have


- ▶ 1 cpd over  $A$ :  $P_A$  (it 'conditions' on nothing)
- ▶ 2 cpds over  $B \mid A$ :  $P_{B|A=0}$  and  $P_{B|A=1}$
- ▶ 2 cpds over  $C \mid B$ :  $P_{C|B=0}$  and  $P_{C|B=1}$

Exercise: for this BN, list all CPDs in tabular form:

$A$	$P_A$	$A$	$B$	$P_{B A}$	$B$	$C$	$P_{C B}$
0	$P_A(0)$	0	0	$P_{B A}(0 0)$	0	0	$P_{C B}(0 0)$
1	$P_A(1)$	0	1	$P_{B A}(1 0)$	0	1	$P_{C B}(1 0)$
		1	0	$P_{B A}(0 1)$	1	0	$P_{C B}(0 1)$
		1	1	$P_{B A}(1 1)$	1	1	$P_{C B}(1 1)$


## Exercise

$A$  is binary,  $B$  is 3-valued, and  $C$  is 4-valued.

This is the BN . List the cpds:

## Exercise

$A$  is binary,  $B$  is 3-valued, and  $C$  is 4-valued.

This is the BN . List the cpds:

					$B$	$C$	$P_{C B}$
		$A$	$B$	$P_{B A}$	0	0	$P_{C B}(0 0)$
		0	0	$P_{B A}(0 0)$	0	1	$P_{C B}(1 0)$
		0	1	$P_{B A}(1 0)$	0	2	$P_{C B}(2 0)$
		0	2	$P_{B A}(2 0)$	0	3	$P_{C B}(3 0)$
		1	0	$P_{B A}(0 1)$	1	0	$P_{C B}(0 1)$
		1	1	$P_{B A}(1 1)$	1	1	$P_{C B}(1 1)$
		1	2	$P_{B A}(2 1)$	1	2	$P_{C B}(2 1)$
					1	3	$P_{C B}(3 1)$
					2	0	$P_{C B}(0 2)$
					2	1	$P_{C B}(1 2)$
					2	2	$P_{C B}(2 2)$
					2	3	$P_{C B}(3 2)$

## Exercise: cost of representation

Consider a joint distribution over 6 binary random variables.

Without making conditional independence assumptions, what is the size of a tabular representation of such a joint distribution?

## Exercise: cost of representation

Consider a joint distribution over 6 binary random variables.

Without making conditional independence assumptions, what is the size of a tabular representation of such a joint distribution?

We have 6 variables, each binary, with no conditional independences, we have to specify a probability value for each and every outcome in the joint sample space directly. This leads to the need for a table with  $2^6$  probability values in it.

## Exercise: cost of representation

Consider a joint distribution over 6 binary random variables.

Without making conditional independence assumptions, what is the size of a tabular representation of such a joint distribution?

We have 6 variables, each binary, with no conditional independences, we have to specify a probability value for each and every outcome in the joint sample space directly. This leads to the need for a table with  $2^6$  probability values in it.

Now we decide to make the conditional independences stated in the following BN, what is the cost of a tabular representation of the joint distribution?

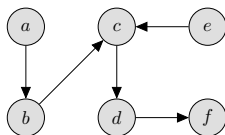


Figure: DAG

## Exercise: cost of representation

Consider a joint distribution over 6 binary random variables.

Without making conditional independence assumptions, what is the size of a tabular representation of such a joint distribution?

We have 6 variables, each binary, with no conditional independences, we have to specify a probability value for each and every outcome in the joint sample space directly. This leads to the need for a table with  $2^6$  probability values in it.

Now we decide to make the conditional independences stated in the following BN, what is the cost of a tabular representation of the joint distribution?

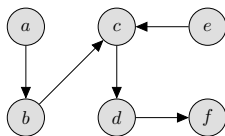
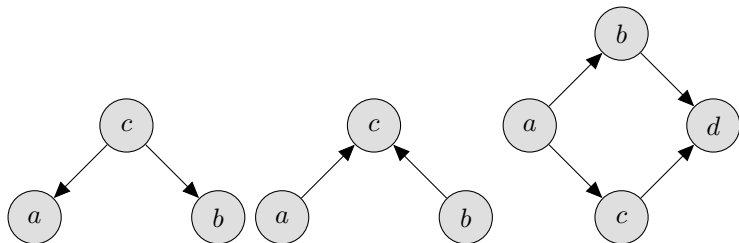


Figure: DAG

The assumptions in the BN make some variables independent of one another, this means that we can store smaller CPDs and compose them together via probability calculus whenever we need the probability of an arbitrary outcome in the joint sample space.  $P_A$  has cost 2 because it has no parents,  $P_{B|A}$  has cost  $2 \times 2 = 4$ ,  $P_{C|BE}$  has cost  $2^3 = 8$ ,  $P_{D|C}$  has cost 4,  $P_E$  has cost 2 and  $P_{F|D}$  has cost 4. A grand total of 24 (instead of  $2^6 = 64$ )

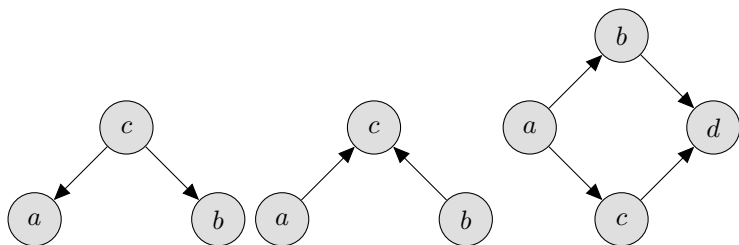
## Exercises



**Figure:** Write down the *minimal* factorisation (i.e., factorisation in terms of elementary factors).



## Exercises



**Figure:** Write down the *minimal* factorisation (i.e., factorisation in terms of elementary factors).

1.  $P_{ABC}(a, b, c) = P_C(c)P_{A|C}(a|c)P_{B|C}(b|c)$
2.  $P_{ABC}(a, b, c) = P_A(a)P_B(b)P_{C|AB}(c|a, b)$
3.  $P_{ABCD}(a, b, c, d) = P_A(a)P_{B|A}(b|a)P_{C|A}(c|a)P_{D|BC}(d|b, c)$

## Inferences

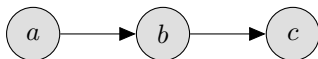


Figure: Example of BN

The BN tells us the CPDs we have to represent explicitly (i.e., store probabilities for). For this example:  $P_A$ ,  $P_{B|A}$  and  $P_{C|B}$ .

What if we want to reason about something that's not  $P_A$  or  $P_{B|A}$  or  $P_{C|B}$ ? Such as

- ▶  $P_B$  or  $P_C$
- ▶ or  $P_{B|C}$  or  $P_{A|B}$  or  $P_{A|C}$  or  $P_{C|A}$
- ▶ or  $P_{BC|A}$  or  $P_{AB|C}$  or  $P_{AC|B}$
- ▶ or  $P_{ABC}$

# Inferences

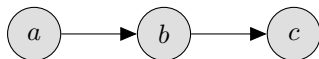


Figure: Example of BN

The BN tells us the CPDs we have to represent explicitly (i.e., store probabilities for). For this example:  $P_A$ ,  $P_{B|A}$  and  $P_{C|B}$ .

What if we want to reason about something that's not  $P_A$  or  $P_{B|A}$  or  $P_{C|B}$ ? Such as

- ▶  $P_B$  or  $P_C$
- ▶ or  $P_{B|C}$  or  $P_{A|B}$  or  $P_{A|C}$  or  $P_{C|A}$
- ▶ or  $P_{BC|A}$  or  $P_{AB|C}$  or  $P_{AC|B}$
- ▶ or  $P_{ABC}$

For whatever combination of variables, we use rules of probability!

## It all starts with the joint distribution

Remember that the BN is coding a set of assumptions that gives us a joint distribution, for the example this distribution assigns probability

$$P_{ABC}(a, b, c) = P_A(a)P_{B|A}(b|a)P_{C|B}(c|b) \quad (6)$$

to an outcome  $(a, b, c)$  in the joint sample space.

If we have to answer a query about any such joint outcome, we could directly look those probabilities up in their corresponding cpds and multiply them together.

Now let's see how we use probability calculus to get to the probability of **each and every** outcome involving any subset of these three rvs.

## Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

## Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

- ▶ start from the definition of conditional probability

$$P_{B|C}(b|c) = \frac{P_{BC}(b,c)}{P_C(c)}$$

# Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

- ▶ start from the definition of conditional probability

$$P_{B|C}(b|c) = \frac{P_{BC}(b,c)}{P_C(c)}$$

- ▶ marginalise  $A$  in the numerator

$$P_{B|C}(b|c) = \frac{\sum_a P_{ABC}(a,b,c)}{P_C(c)}$$

# Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

- ▶ start from the definition of conditional probability

$$P_{B|C}(b|c) = \frac{P_{BC}(b,c)}{P_C(c)}$$

- ▶ marginalise  $A$  in the numerator

$$P_{B|C}(b|c) = \frac{\sum_a P_{ABC}(a,b,c)}{P_C(c)}$$

- ▶ factorise the joint distribution to introduce the cpds we have

$$P_{B|C}(b|c) = \frac{\sum_a P_A(a)P_{B|A}(b|a)P_{C|B}(c|b)}{P_C(c)}$$



# Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

- ▶ start from the definition of conditional probability

$$P_{B|C}(b|c) = \frac{P_{BC}(b,c)}{P_C(c)}$$

- ▶ marginalise  $A$  in the numerator

$$P_{B|C}(b|c) = \frac{\sum_a P_{ABC}(a,b,c)}{P_C(c)}$$

- ▶ factorise the joint distribution to introduce the cpds we have

$$P_{B|C}(b|c) = \frac{\sum_a P_A(a)P_{B|A}(b|a)P_{C|B}(c|b)}{P_C(c)}$$

- ▶ rearrange the terms for convenience

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a)P_{B|A}(b|a)}{P_C(c)}$$

# Conditional probability and marginalisation

If we have representations for  $P_A$ ,  $P_{B|A}$ , and  $P_{C|B}$ . Infer  $P_{B|C}$ :

- ▶ start from the definition of conditional probability

$$P_{B|C}(b|c) = \frac{P_{BC}(b,c)}{P_C(c)}$$

- ▶ marginalise  $A$  in the numerator

$$P_{B|C}(b|c) = \frac{\sum_a P_{ABC}(a,b,c)}{P_C(c)}$$

- ▶ factorise the joint distribution to introduce the cpds we have

$$P_{B|C}(b|c) = \frac{\sum_a P_A(a)P_{B|A}(b|a)P_{C|B}(c|b)}{P_C(c)}$$

- ▶ rearrange the terms for convenience

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a)P_{B|A}(b|a)}{P_C(c)}$$

- ▶ we would be able to compute every term that appears in the numerator by looking up cells of our elementary cpds

## Continuation

- ▶ we are here, where the denominator requires a probability that's not in any of the elementary cpds

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a) P_{B|A}(b|a)}{P_C(c)}$$

## Continuation

- ▶ we are here, where the denominator requires a probability that's not in any of the elementary cpds

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a) P_{B|A}(b|a)}{P_C(c)}$$

- ▶ now let's re-express the marginal probability in the denominator

$$\begin{aligned} P_C(c) &= \sum_a \sum_b P_{ABC}(a, b, c) \\ &= \sum_a \sum_b P_A(a) P_{B|A}(b|a) P_{C|B}(c|b) \\ &= \sum_a P_A(a) \sum_b P_{B|A}(b|a) P_{C|B}(c|b) \end{aligned}$$

## Continuation

- ▶ we are here, where the denominator requires a probability that's not in any of the elementary cpds

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a) P_{B|A}(b|a)}{P_C(c)}$$

- ▶ now let's re-express the marginal probability in the denominator

$$\begin{aligned} P_C(c) &= \sum_a \sum_b P_{ABC}(a, b, c) \\ &= \sum_a \sum_b P_A(a) P_{B|A}(b|a) P_{C|B}(c|b) \\ &= \sum_a P_A(a) \sum_b P_{B|A}(b|a) P_{C|B}(c|b) \end{aligned}$$

- ▶ substitute it back in the conditional to see that every term in it is now something we can look up in a table

$$P_{B|C}(b|c) = \frac{P_{C|B}(c|b) \sum_a P_A(a) P_{B|A}(b|a)}{\sum_a P_A(a) \sum_b P_{B|A}(b|a) P_{C|B}(c|b)}$$

## Exercise: probability calculus

Using the same BN as before, give expressions for the probability of an outcome in any of the following marginals and conditionals:

- ▶  $P_B$  or  $P_C$
- ▶ or  $P_{B|C}$  or  $P_{A|B}$  or  $P_{A|C}$  or  $P_{C|A}$
- ▶ or  $P_{BC|A}$  or  $P_{AB|C}$  or  $P_{AC|B}$

Once you've managed to represent a quantity in terms of probabilities in elementary cpds, it's okay to reuse it without expanding its expression (for example, once you have an expression for  $P_B(b)$  it's okay to reuse  $P_B(b)$  in other expressions such as  $P_{A|B}(a|b) = \frac{P_{AB}(a,b)}{P_B(b)}$ , but in this case you would still need to find an expression for  $P_{AB}(a,b)$ ).

## What Next?

We are going to apply this knowledge to design text classifiers, language models, taggers and more.

NLP1 students: the next slide is an exercise on naive Bayes classifiers, you've already seen the NBC, but now you will give it a PGM treatment.

NTMI students: you can ignore the next slide/exercise as this model and application will be covered very carefully in class (and in our lecture notes<sup>5</sup>).

---

<sup>5</sup><https://wilkeraziz.github.io/assets/pdfs/generative.pdf>

## Exercise - NBC

A probabilistic text classifier is a system built upon a conditional distribution  $P_{Y|X=w_{1:N}}$  where  $X = \langle W_1 = w_1, \dots, W_N = w_N \rangle$  is an observed document (expressed as a sequence of  $N$  random words drawn from a finite vocabulary containing  $V$  symbols) and  $Y$  is a random variable taking on one of  $K$  classes (e.g., positive, neutral, negative). A *naive Bayes classifier* (NBC) is a form of generative classifier built upon the joint distribution  $P_{XY}$  over the cross-product space of all documents (all finite-length word sequences) and classes. The NBC makes a key conditional independence assumption, namely, that *given* the class  $Y = y$ , the words in a document are independent of one another. That is,  $W_i \perp W_j \mid Y$  for  $j \neq i$ .<sup>6</sup>

1) Make a diagram of the conditional independences in NBC. 2) How many cpds are necessary to represent this model in tabular form, and how many probability values in total? 3) Express the joint probability  $P_{XY}(w_{1:N}, y)$  of a given document  $w_{1:N}$  and its class  $y$  in terms of the elementary factors of the model. 4) Express the probability of the class  $Y = y$  given the document  $X = w_{1:N}$ , again in terms of elementary factors of the model.<sup>7</sup>

<sup>6</sup>Most people don't need this information, but, if you are wondering, words are also assumed to be independent of their own position in the sequence.

<sup>7</sup>For this exercise, you can pretend  $N = 3$  if that makes it easier for you.



# Solution - NBC



**Figure:** 1) NBC for 3 words (left) and generalisation to  $N$  words (right; the plate can be thought of as a loop where  $n$  varies from 1 to  $N$ ).

2) We need 1 cpd for  $P_Y$  and  $K$  cpds for  $W|Y$  (i.e.,  $P_{W|Y=1}, \dots, P_{W|Y=K}$ ), assuming the vocabulary of known words has size  $V$  we have  $K + K \times V$  probabilities.

3)  $P_{YX}(y, w_{1:N}) = P_Y(y) \prod_{n=1}^N P_{W|Y}(w_n|y)$ , where  $W$  is a random variable taking on values in the vocabulary of known words.

4) By conditional prob:  $P_{Y|X}(y|w_{1:N}) = \frac{P_{YX}(y, w_{1:N})}{P_X(w_{1:N})}$ . The numerator is the expression in (3). The denominator is a marginal of that expression:

$$P_X(w_{1:N}) = \sum_{k=1}^K P_{YX}(k, w_{1:N}) = \sum_{k=1}^K P_Y(k) \prod_{n=1}^N P_{W|Y}(w_n|k).$$

# References I

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.