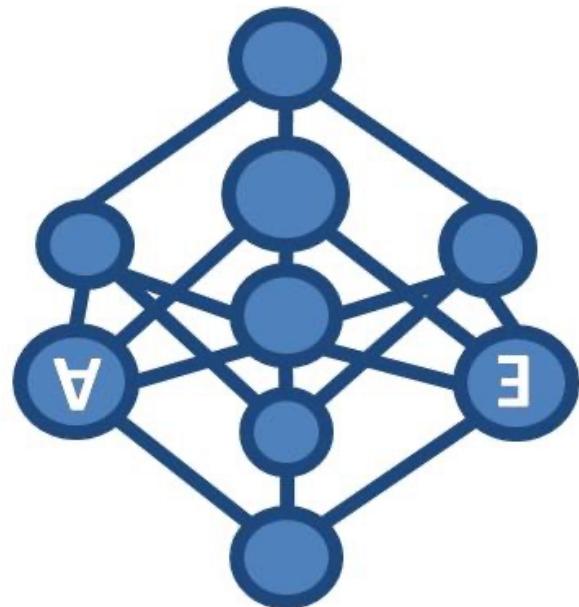


Probabilistic Graphical Models*

Bayesian Networks



TECHNISCHE
UNIVERSITÄT
DARMSTADT



*Thanks to Carlos Guestrin, Pedro Domingos and many others for making their slides publically available



What you need to know from last class

- Basic definitions of probabilities
- Independence
- Conditional independence
- The chain rule
- Bayes rule



Let's start on Bayesian Networks

- One of the most exciting recent advancements in statistical AI

Judea Pearl won the [ACM Turing Award 2012](#) for his fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning



- Compact representation for exponentially-large probability distributions
- Fast marginalization
- Exploit conditional independencies



Representing Distributions by Enumeration



- Consider $P(X_i)$
 - Assign a probability to each $x_i \in \text{Val}(X_i)$
 - Number of parameters assuming $|\text{Val}(X_i)| = k$?
 $k - 1$
- Now, consider $P(X_1, \dots, X_n)$
 - How many parameters assuming again $|\text{Val}(X_i)| = k$?
 - Same thing,
 $k^n - 1$
- **Bayesian networks will often need fewer parameters. What is the trick?**



Conditional Parameterization (2 nodes)

- Let us assume that

Grade is influenced by Intelligence

- Make use of **chain rule**

$$P(G, I) = P(G | I) \cdot P(I)$$

$$P(I = VH, G = B)$$

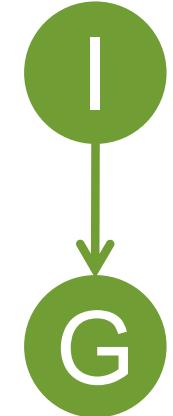
$$= P(I = VH) \cdot P(G = B | I = VH)$$

$$= 0.85 \cdot 0.1$$

$$= 0.085$$

	VH	H
P(I)	0.85	0.15
	VH	H

	I =	VH	H
P(G I)	A	0.9	0.5
	I =	VH	H
B	0.1	0.5	



Represent conditional distributions graphically

Same # of parameters as by enumeration

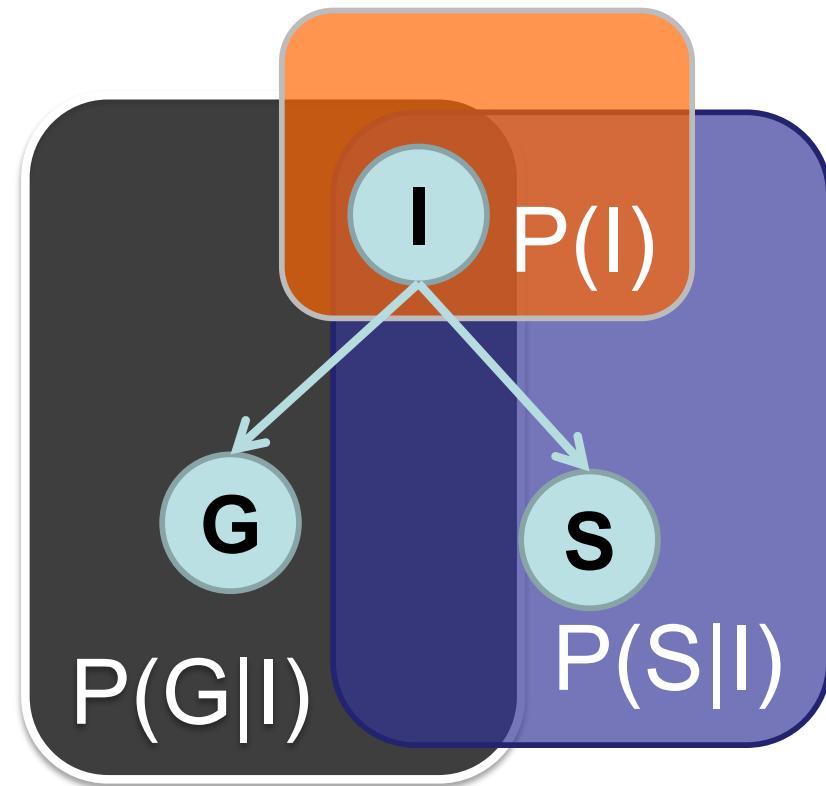


Conditional Parameterization (3 nodes) and what if variables are **independent**?

- Grade and SAT score are influenced by Intelligence
- but $(G \perp S | I)$, i.e., $P(G | S, I) = P(G | I)$

$$\begin{aligned}P(G, S, I) &= P(G, S | I) \cdot P(I) \\&= P(G | S, I) \cdot P(S | I) \cdot P(I) \\&= P(G | I) \cdot P(S | I) \cdot P(I)\end{aligned}$$

Independence can lead to smaller # of parameters as by enumeration



Can we even get a linear complexity?

- $(X_i \perp X_j), \forall i, j$ enough?



Can we even get a linear complexity?

- $(X_i \perp X_j), \forall i, j$ enough?

Let $X1$ and $X2$ be drawn from $Bernoulli(p=0.5)$ [value TRUE with prob p and value FALSE with q=1-p] and $X3 = X1 \text{ xor } X2$. Now, $P(Xi, Xj) = 1/4 = P(Xi)P(Xj)$ (use a table to show this, i.e., independence for all pairs).



Can we even get a linear complexity?

- $(X_i \perp X_j), \forall i, j$ **is not enough**
- We must assume that
$$(X \perp Y), \forall X, Y \text{ subsets of } \{X_1, \dots, X_n\}$$

Let $X1$ and $X2$ be drawn from $Bernoulli(p=0.5)$ [value TRUE with prob p and value FALSE with $q=1-p$] and $X3 = X1 \text{ xor } X2$. Now, $P(Xi, Xj) = 1/4 = P(Xi)P(Xj)$ (use a table to show this, i.e., independence for all pairs). Since $X3$ depends deterministically on $X1$ and $X2$ it cannot be the case that $X1, X2$ are independent of $X3$.

- Now, we can write $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i)$
- How many parameters?

$$n \cdot (k - 1) = O(n)$$





Let us now touch upon almost all topics — representation, inference, and learning— of the lecture once so that you get the big picture. For that use a simple Bayesian network for classification:

THE NAÏVE BAYES CLASSIFIER



The Naïve Bayes Model

- **Your first real Bayesian network !**
- **Class variable:** C
- **Evidence variables:** $\{X_1, \dots, X_n\}$
- **Assume** that $(X \perp Y | C), \forall X, Y$ subsets of $\{X_1, \dots, X_n\}$

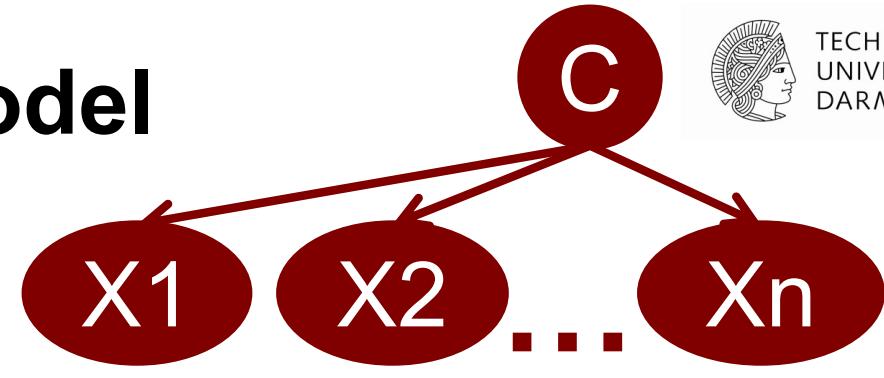
$$P(X_1, \dots, X_n, C) = P(C) \cdot \prod_{i=1}^n P(X_i | C)$$



The Naïve Bayes Model

$$P(X_1, \dots, X_n, C)$$

$$= P(C) \cdot P(X_1 | C) \cdot P(X_2 | X_1, C) \cdot \dots \cdot P(X_n | X_1, \dots, X_{n-1}, C)$$



So far, no assumptions. Just the chain rule.

Now, let us make the assumption that $(\mathbf{X} \perp \mathbf{Y} | C), \forall \mathbf{X}, \mathbf{Y}$

Then we get the Naïve Bayes model

$$P(X_2 | X_1, C) = P(X_2 | C) \quad P(X_n | X_1, \dots, X_{n-1}, C) = P(X_n | C)$$

$$(X_1 \perp X_2 | C)$$

...

$$(X_n \perp X_1 X_2 \dots X_{n-1} | C)$$



Example application of Naïve Bayes: Spam Classification

From: "" <takworlld@hotmail.com>

Subject: real estate is the only way... gem oalvgkay

Anyone can buy real estate with no money down

Stop paying rent TODAY !

There is no need to spend hundreds or even thousands for similar courses

I am 22 years old and I have already purchased 6 properties using the methods outlined in this truly INCREDIBLE ebook.

Change your life NOW !

=====

Click Below to order:

<http://www.wholesaledaily.com/sales/nmd.htm>

=====



Is this
email
spam or
not?



Defintion of Classification

Given

- A description of an instance, $x \in X$, where X is the *instance language* or *feature space*.
 - **How to represent text documents?** E.g. Bag-Of-Words, i.e., we count how often each word of a fixed dictionary appears in the documents
- A fixed set of categories: $C = \{c_1, c_2, \dots, c_n\}$

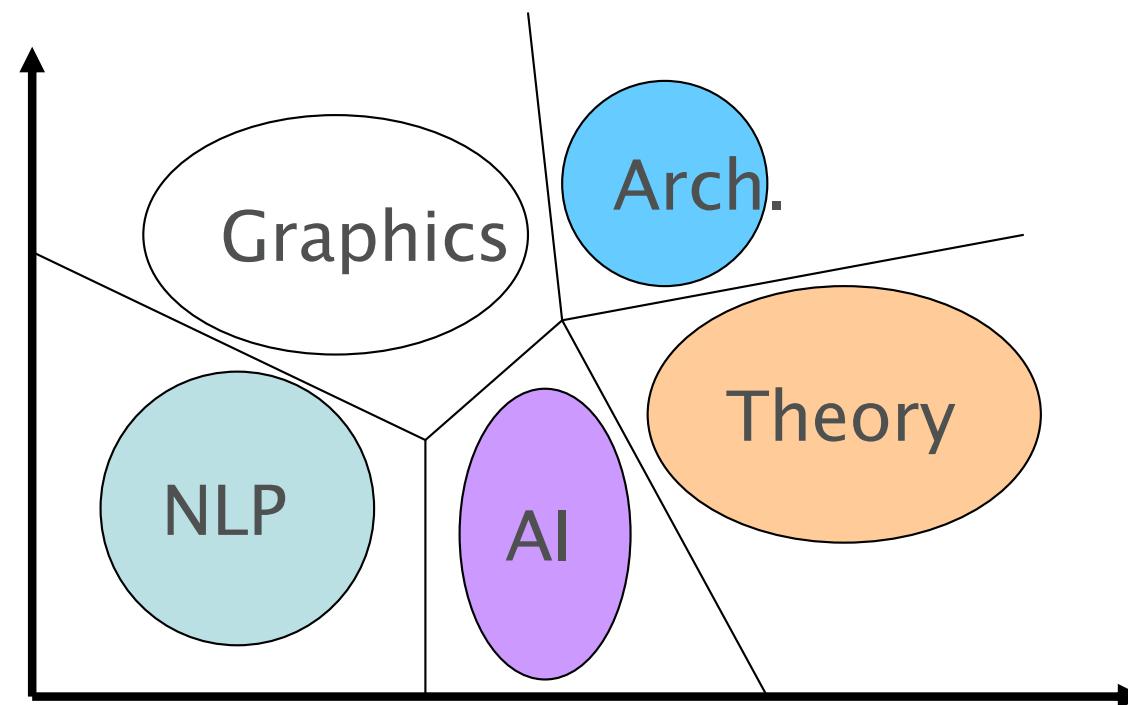
Determine

- The class of x : $c(x) \in C$, where $c(x)$ is a *classification function* whose domain is X and whose range is C .
 - **If x is spam then $c(x)=1$ otherwise $c(x)=0$**
 - **We want to know how to build classification functions (“classifiers”).**



A Graphical View of Text Classification

The feature space is partitioned into regions of different classes,
e.g. topics of research papers from computer science



Examples of Text Clasifications

- **CLASS LABELS=BINARY**
 - “spam” / “not spam”
- **CLASS LABELS=TOPICS**
 - “finance” / “sports” / “asia”
- **CLASS LABELS=OPINION**
 - “like” / “hate” / “neutral”
- **CLASS LABELS=AUTHOR**
 - “Shakespeare” / “Marlowe” / “Ben Jonson”



Indeed, Naïve Bayes is not the only approach one can images



1. Manual classification

- Often used by Yahoo!, Looksmart, about.com, ODP, Medline
- Very accurate when job is done by experts
- Consistent when the problem size and team is small
- Difficult and expensive to scale

2. Automatic document classification

- Hand-coded rule-based systems
 - Reuters, CIA, Verity, ...
 - E.g., assign category if document contains a given Boolean combination of words
 - Commercial systems have complex query languages (everything in IR query languages + accumulators)
 - Accuracy can be high if a rule has been carefully refined over time by a subject expert also expert in the query language
 - Building and maintaining these rules is expensive



3. Supervised learning

- All systems rely on machine learning (Autonomy, MSN, Verity, Enkata, Yahoo!, ...)
 - k-Nearest Neighbors (simple, powerful)
 - *Naive Bayes* (*simple method*)
 - **Support-vector machines** (more powerful)
 - Deep Learning (most powerful), using e.g. transformers for learning embedding
 - ... plus many other methods
- **No free lunch**: requires hand-classified training data
- But data can be built up (and refined) by amateurs





How to get a classifier from the joint distribution / Bayesian network?

Reminder: Bayes' Rule

$$P(C, D) = P(C | D)P(D) = P(D | C)P(C)$$

C = Class, D = Document

$$P(C | D) = \frac{P(D | C)P(C)}{P(D)}$$



Maximum a posteriori Hypothesis

$$c_{MAP} \equiv \operatorname{argmax}_{c \in C} P(c | D)$$

$$= \operatorname{argmax}_{c \in C} \frac{P(D | c)P(c)}{P(D)}$$

$$= \operatorname{argmax}_{c \in C} P(D | c)P(c)$$

**Since $P(D)$ is
constant**





Maximum Likelihood (ML) Hypothesis

If all hypotheses are a priori equally likely, we only need to consider the $P(D|c)$ term:

$$c_{ML} \equiv \operatorname{argmax}_{c \in C} P(D | c)$$



Naive Bayes Classifiers

Task: Classify a new instance D based on a tuple of attribute values $D = \langle x_1, x_2, \dots, x_n \rangle$ into one of the classes $c_j \in C$

$$c_{MAP} = \operatorname{argmax}_{c \in C} P(c | x_1, x_2, \dots, x_n)$$

$$= \operatorname{argmax}_{c \in C} \frac{P(x_1, x_2, \dots, x_n | c)P(c)}{P(x_1, x_2, \dots, x_n)}$$

$$= \operatorname{argmax}_{c \in C} P(x_1, x_2, \dots, x_n | c)P(c)$$

For instance, bag-of-word features when classifying text documents



Naïve Bayes (NB) Classifier: Assumption

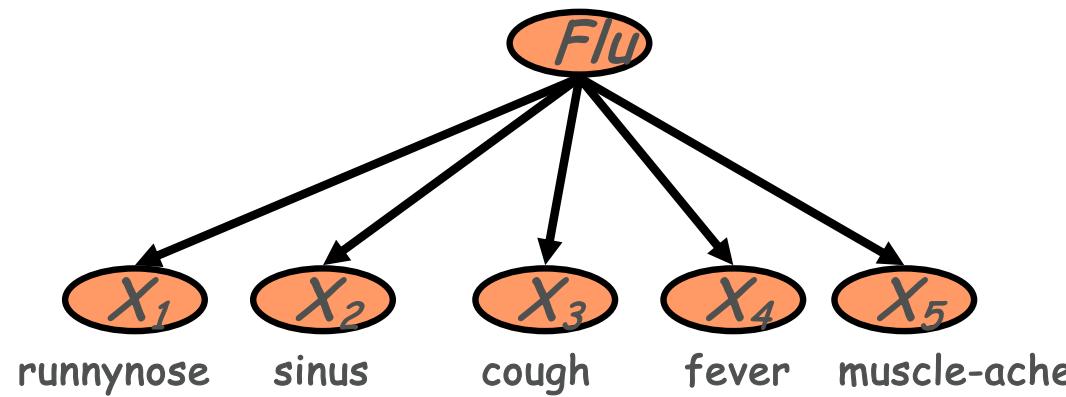
- $P(c_j)$
 - Can be estimated from the frequency of classes in the training examples.
- $P(x_1, x_2, \dots, x_n | c_j)$
 - $O(|X|^n \cdot |C|)$ parameters
 - Could only be estimated if a very, very large number of training examples was available.

NB Conditional Independence Assumption to the rescue!

- **Assume that the probability of observing the attributes is equal to the product of the individual probabilities $P(x_i | c_j)$.**



The Naïve Bayes Classifier



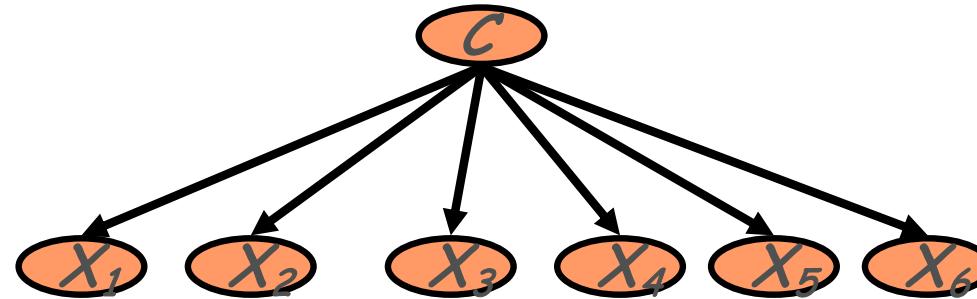
- **Conditional Independence Assumption:** features are independent of each other given the class:

$$P(X_1, \dots, X_5 | C) = P(X_1 | C) \bullet P(X_2 | C) \bullet \dots \bullet P(X_5 | C)$$



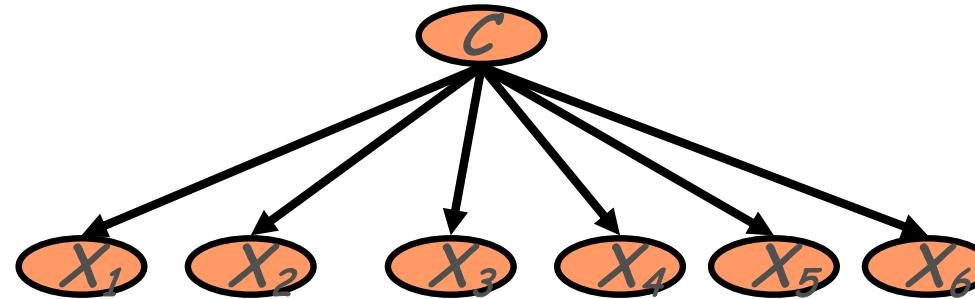
Learning the Model

Where do the parameters come from?



Learning the Model

Where do the parameters come from?



- First approach: maximum likelihood estimates
 - simply use the frequencies in the data

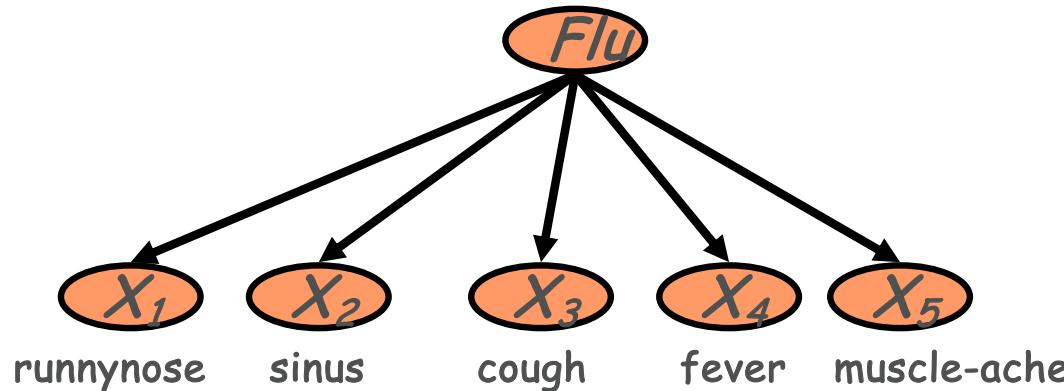
We will derive the
equations
later in class

$$\hat{P}(c_j) = \frac{N(C = c_j)}{N}$$

$$\hat{P}(x_i | c_j) = \frac{N(X_i = x_i, C = c_j)}{N(C = c_j)}$$



Problem with Max Likelihood



- What if we have seen no training cases where patient had flu and muscle aches?

$$P(X_1, \dots, X_5 | C) = P(X_1 | C) \cdot P(X_2 | C) \cdot \dots \cdot P(X_5 | C)$$

$$\hat{P}(X_5 = t | C = \text{flu}) = \frac{N(X_5 = t, C = \text{flu})}{N(C = \text{flu})} = 0$$

- Zero probabilities cannot be conditioned away, no matter the other evidence!

$$\ell = \arg \max_c \hat{P}(c) \prod_i \hat{P}(x_i | c) = 0$$





Smoothing to Avoid Overfitting

$$\hat{P}(x_i | c_j) = \frac{N(X_i = x_i, C = c_j) + 1}{N(C = c_j) + k}$$

We will derive the
equations
later in class

of values of X_i

- Somewhat more subtle version

overall fraction in data
where $X_i = x_{i,k}$

$$\hat{P}(x_{i,k} | c_j) = \frac{N(X_i = x_{i,k}, C = c_j) + mp_{i,k}}{N(C = c_j) + m}$$

extent of “smoothing”



Wrap-up: (1) Naïve Bayes Learning

- From training corpus, extract *Vocabulary*
- Calculate required $P(c_j)$ and $P(x_k | c_j)$ terms
 - **For each c_j in C do**

$docs_j \leftarrow$ subset of documents for which the target class is c_j

$$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$

- **For each word x_k in *Vocabulary***

$n_k \leftarrow$ number of occurrences of x_k in all $docs_j$

$$P(x_k | c_j) \leftarrow \frac{n_k + 1}{|docs_j| + |\text{Vocabulary}|}$$



Wrap-up: (2) Naïve Bayes Classifying



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- Return c_{NB} , where

$$c_{NB} = \operatorname{argmax}_{c_j \in C} P(c_j) \prod_{i \in \text{vocabulary}} P(x_i | c_j)$$

- To avoid small numbers, use the sum of logs





Naïve Bayes works

- Uses a Naïve Bayes classifier on SPAMCOP
- EMail M is spam if $P(\text{Spam}|M) > P(\text{NonSpam}|M)$
- Method
 - Tokenize message using Porter Stemmer
 - Estimate $P(W|C)$ using m-estimate (a form of smoothing)
 - Remove words that do not satisfy certain conditions
 - **Train: 160 spams, 466 non-spams**
 - **Test: 277 spams, 346 non-spams**
- Results: ERROR RATE of 4.33%
 - Worse results using trigrams





Take-Away Message:

Naive Bayes is Not So Naive

- Naïve Bayes: First and Second place in KDD-CUP 1997 competition, among 16 (then) state of the art algorithms
 - Goal: Financial services industry direct mail response prediction model: Predict if the recipient of mail will actually respond to the advertisement – 750,000 records.
- A good dependable baseline for text classification
 - But not the best!
- Optimal if the Independence Assumptions hold:
 - If assumed independence is correct, then it is the Bayes Optimal Classifier for problem
- Very Fast:
 - Learning with one pass over the data;
 - Testing linear in the number of attributes, and document collection size
- Low Storage requirements

Though deep networks are dominating today



So far



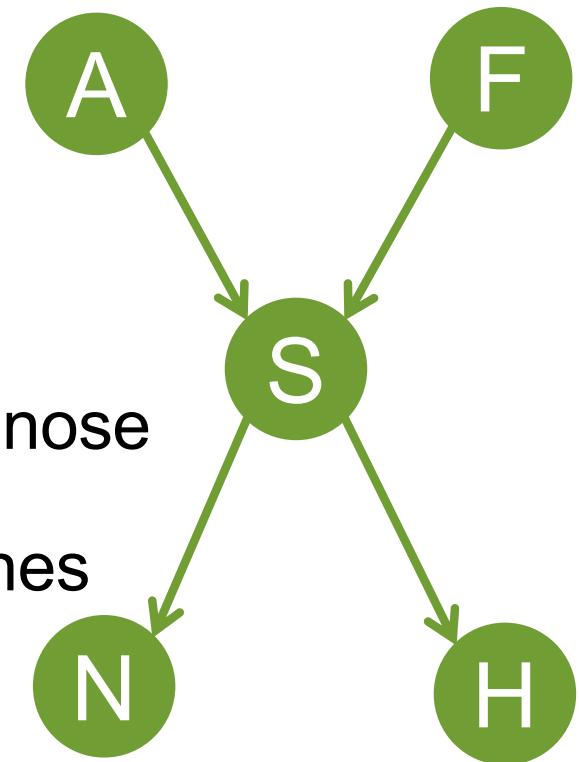
- We've heard of Bayes nets
- We've played with Bayes nets
- We've even used them for classifying spam

- Now, we'll learn
 - the definition and semantics of BNs and
 - **relate them to independence assumptions encoded by the graph**



„Causal“ Structure

- Suppose we know the following:
 - The flu causes sinus inflammation
 - Allergies cause sinus inflammation
 - Sinus inflammation causes a runny nose
 - Sinus inflammation causes headaches
- How are these connected?



Possible Queries

- **Inference**

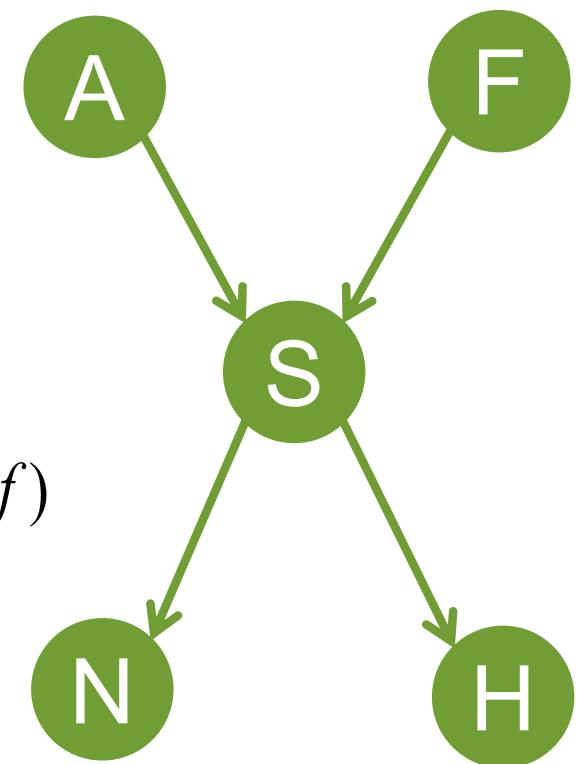
- Assume $H=t$, $N=f$. What is the probability $P(A=t|H=t, N=f)$ of an allergic reaction?

- **Most probable explanation**

- $\max_{f,a,s} P(F = f, A = a, S = s | H = t, N = f)$

- **Active data collection**

- What is next best test, i.e., variable to observe

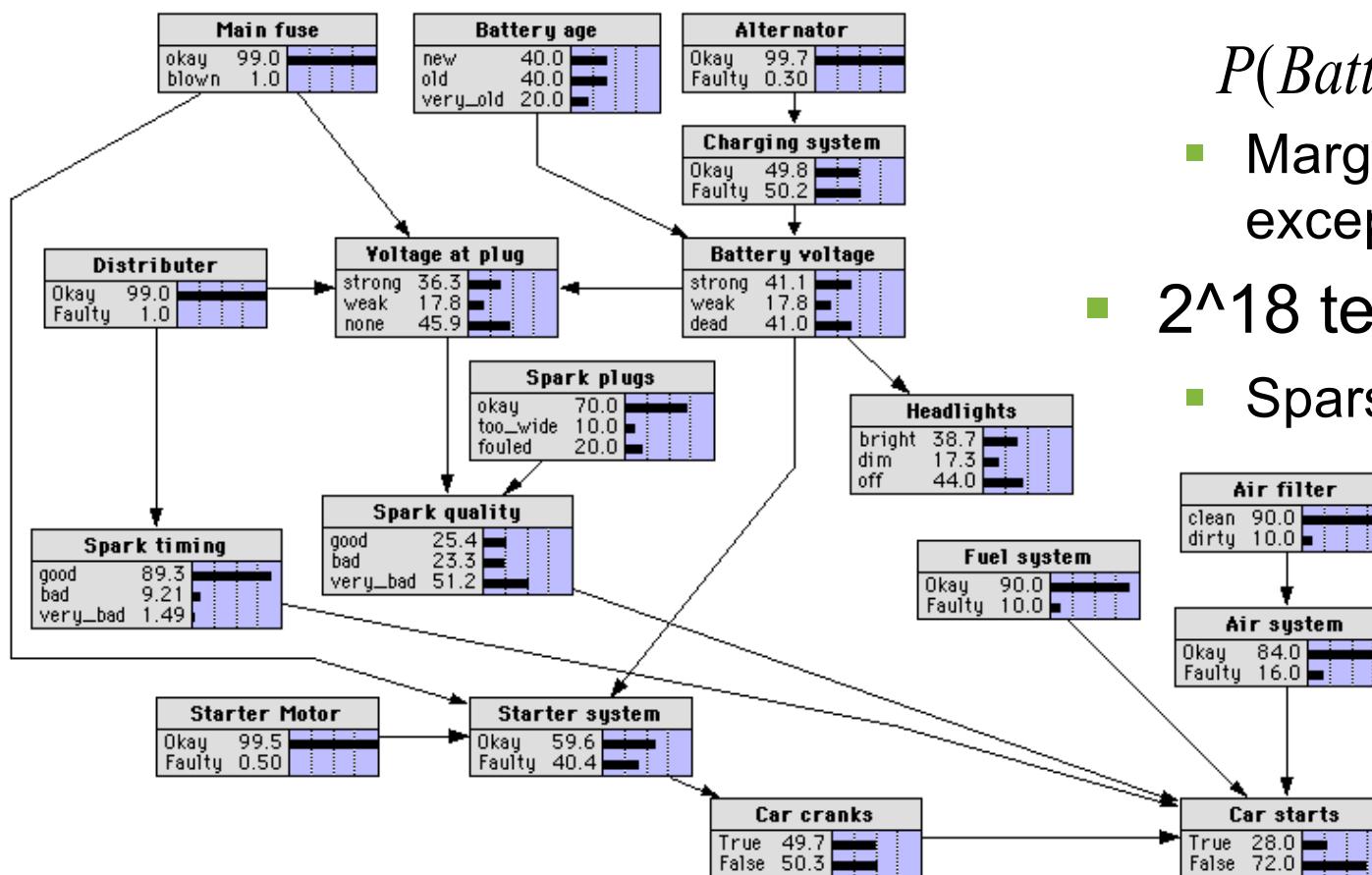


„Car Diagnosis“ Bayesian Network

- 18 binary attributes
- Inference

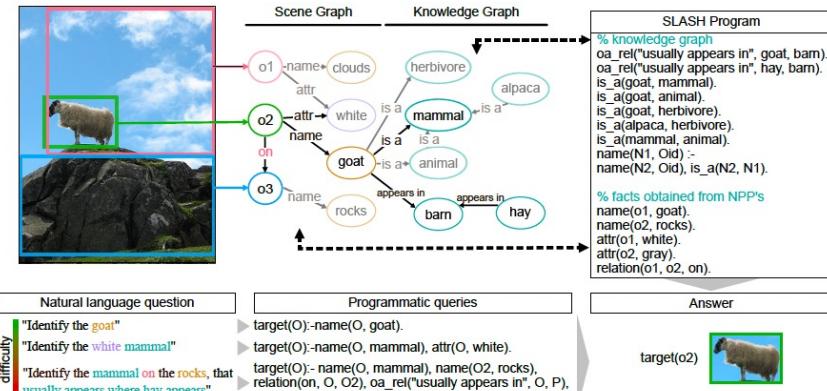
$$P(\text{BatteryAge} \mid \text{Starts} = f)$$

- Marginalize all variables except BA and S
- 2^{18} terms, why so fast?
 - Sparse structure !!!

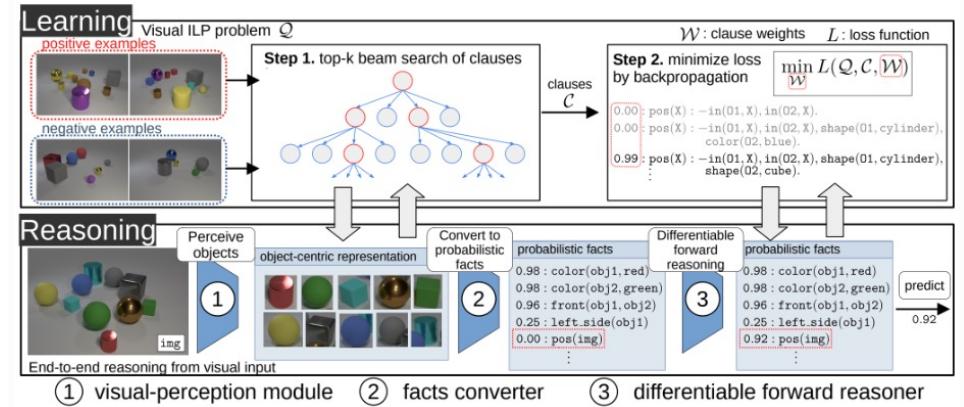


Not impressed?

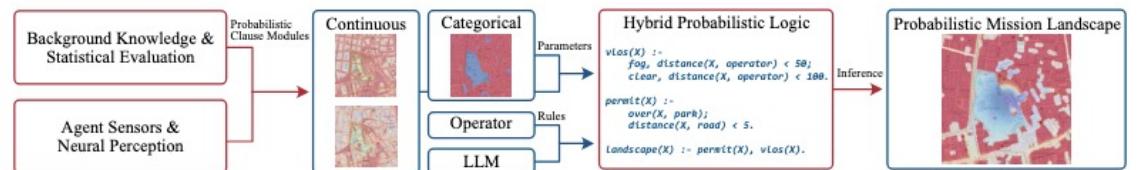
- We are currently dealing with (differentiable) models that have far more than 300 variables (nodes)
- $> 2^{300} \sim 10^{90}$ terms, btw, the age of the earth: $\sim 4.54 \times 10^9$ years $\sim 10^{20}$ seconds



Skryagin, Ochs, Singh Dhami, Kersting (2023): **Scalable Neural-Probabilistic Answer Set Programming**. Journal of Artificial Intelligence Research (JAIR).



Shindo, Pfanschilling, Singh Dhami, Kersting (2023): **alphalLP: Thinking Visual Scenes as Differentiable Logic Programs**. Machine Learning Journal (MLJ)



Simon Kohaut, Benedict Flade, Daniel Ochs, Devendra Singh Dhami, Julian Eggert, Kristian Kersting (2025): **Probabilistic Mission Design for Neuro-Symbolic Unmanned Aircraft Systems**. IEEE Transactions on Intelligent Transportation Systems:1-10.

Interested in joining?



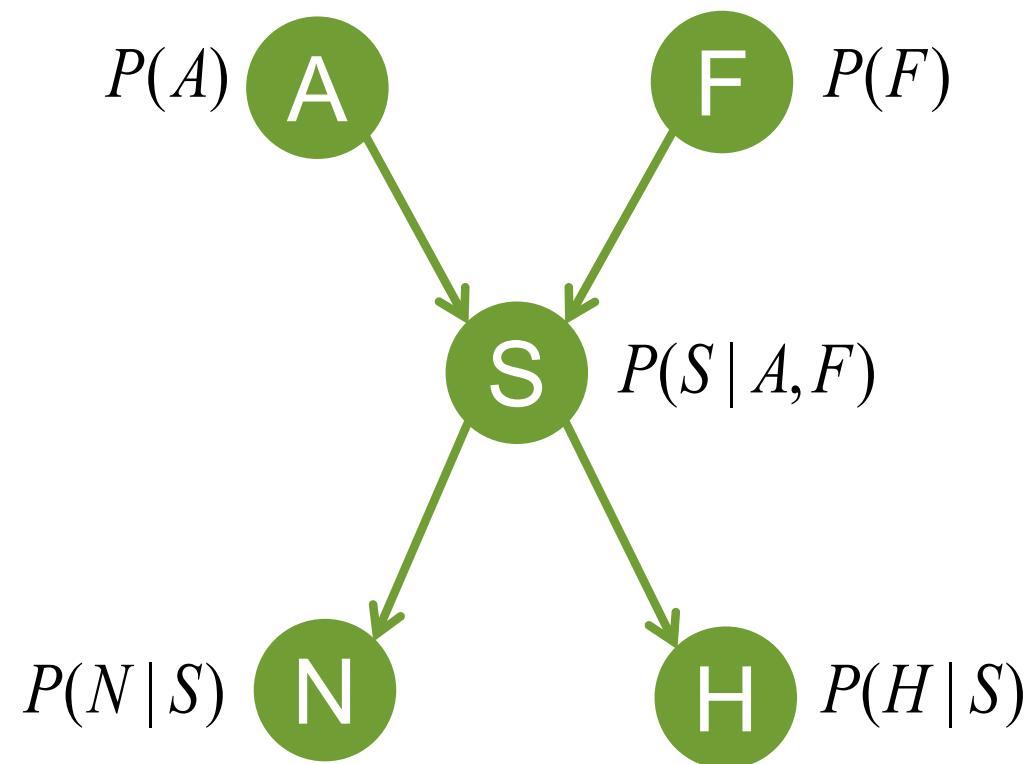
Factored Joint Distributions

Preview



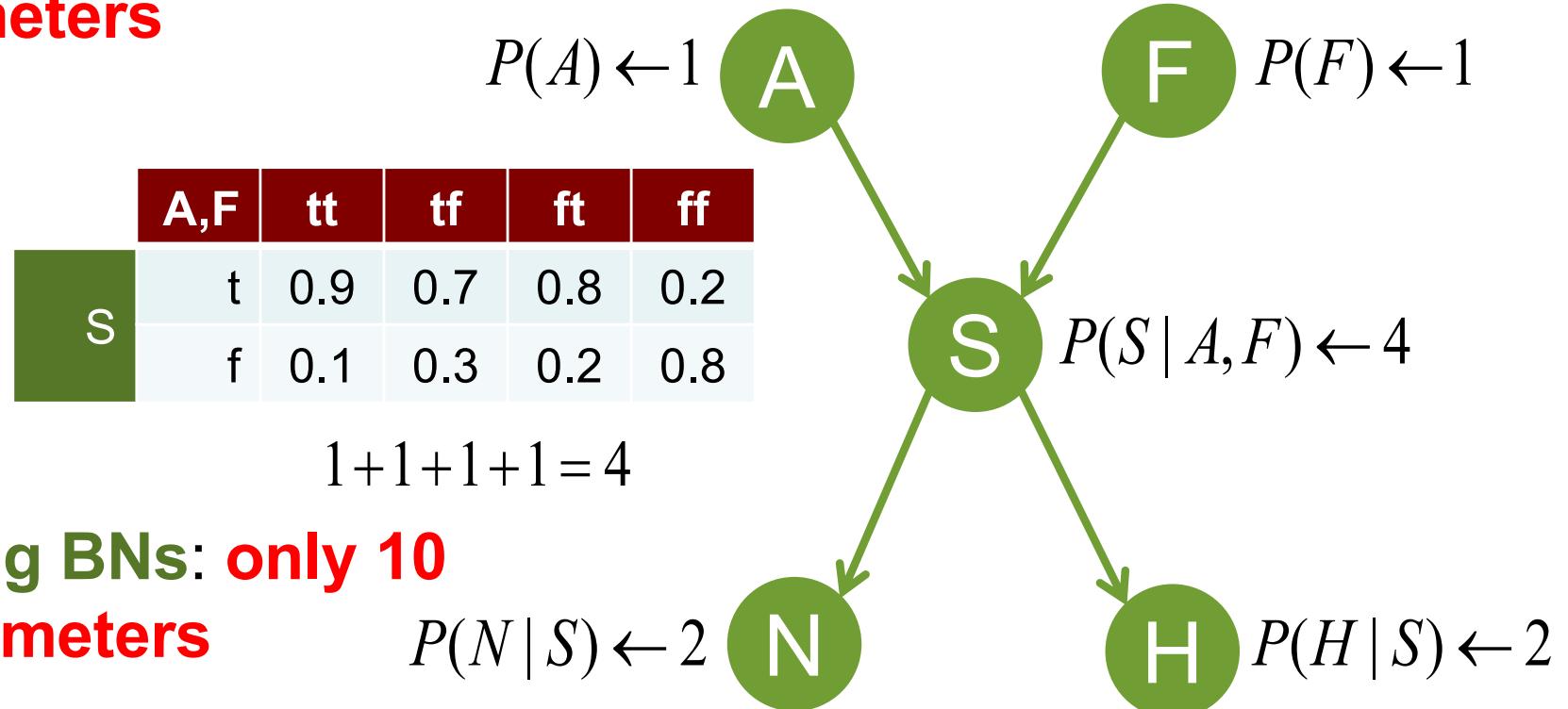
- Intuition: **Nodes = Variables**, **Edges = Influences**

$$\begin{aligned} P(A, F, S, H, N) \\ = P(A) \\ \cdot P(F) \\ \cdot P(S | A, F) \\ \cdot P(N | S) \\ \cdot P(H | S) \end{aligned}$$



Number of Parameters

- **Sparse structure**: compact representation for exponentially-large probability distributions
- Enumerative representation of binary variables: $2^5 - 1 = 31$ parameters



Key: Independence Assumptions

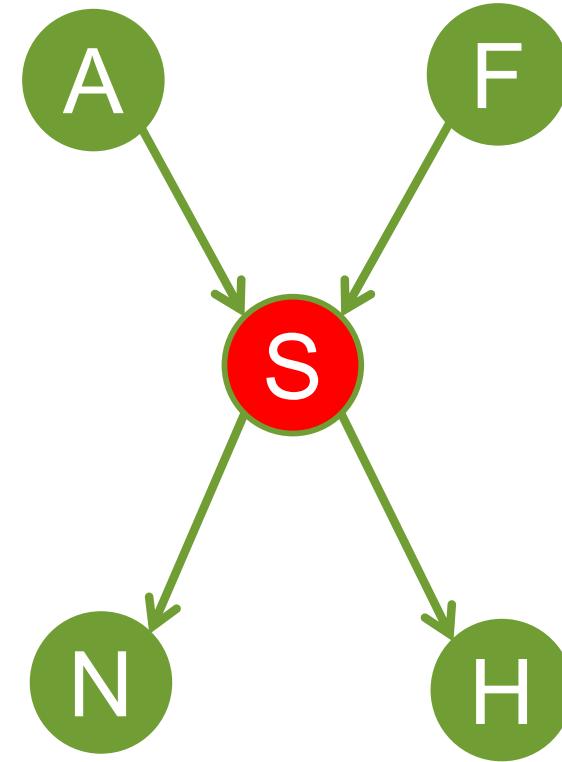
(intuitively clear)

$$\neg(F \perp H)$$

$$(F \perp H | S)$$

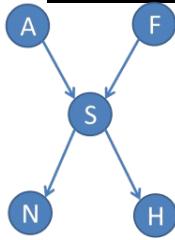
$$\neg(N \perp A)$$

$$(N \perp A | S)$$



Knowing sinus separates
symptoms from causes





Recap: (Marginal) Independence



- Flu and Allergy are marginally independent

$$(A \perp F)$$

$$P(A, F) = P(A) \cdot P(F)$$

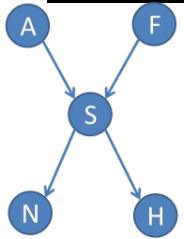
A	t	f	F	t	f	A,F	t	f
t	0.3	0.7	t	0.1	0.9	t	0.3*0.1=0.03	0.3*0.9=0.27
f			f			f	0.7*0.1=0.07	0.7*0.9=0.63

\forall subsets $X, Y \subseteq \{X_1, \dots, X_n\} : (X \perp Y)$

- General case:

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i)$$





Recap: (Conditional) Independence

- Flu and Headache are (not) marginally independent
- Flu and headache are independent given Sinus infection

$$\neg(F \perp H) \quad P(F, H) \neq P(F) \cdot P(H)$$

$$(F \perp H | S) \quad P(F, H | S) = P(F | S) \cdot P(H | S)$$

$$P(F | H, S) = P(F | S)$$

- Generally: $(X_1 \perp X_2 \dots X_n | C)$ iff

$$P(X_1, X_2, \dots, X_n | C) = P(X_1 | C) \cdot P(X_2, \dots, X_n | C)$$

$$P(X_1 | X_2, \dots, X_n, C) = P(X_1 | C)$$

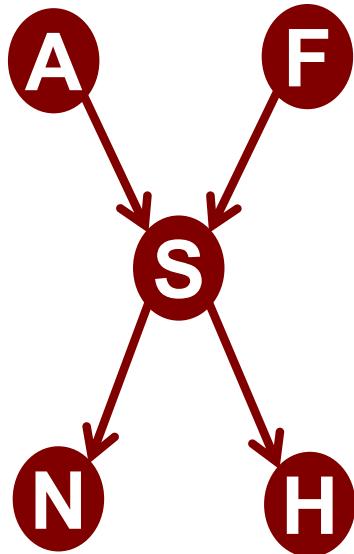


Local Markov Assumption

(Second most important slide!)



TECHNISCHE
UNIVERSITÄT
DARMSTADT



A variable X is independent of its non-descendants given its parents and only its parents ($X_i \perp \text{NonDescendants}_{X_i} | \text{Pa}_{X_i}$)

$$\text{Pa}_F = \emptyset$$

$$\text{NonDescendants}_F = \{A\}$$

$$(F \perp A)$$

$$\text{Pa}_S = \{F, A\}$$

$$\text{NonDescendants}_S = \emptyset$$

$$(S \perp ?? | F, A)$$

NO ASSUMPTIONS

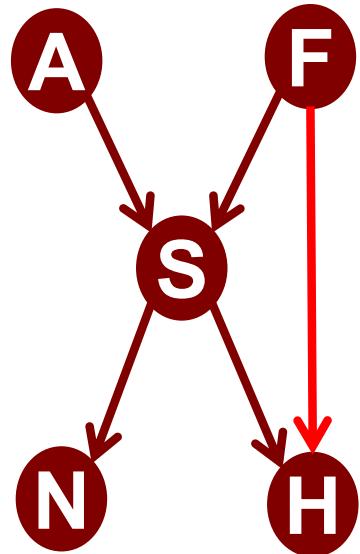
$$\text{Pa}_N = \{S\}$$

$$\text{NonDescendants}_N = \{F, A, H\}$$

$$(N \perp \{F, A, H\} | S)$$



Local Markov Assumption



before edge included

A variable X is independent of its non-descendants given its parents and only its parents ($X_i \perp \text{NonDescendants}_{X_i} | \text{Pa}_{X_i}$)

$$\text{Pa}_H = \{S\}$$

$$\text{NonDescendants}_H = \{A, F, N\}$$

$$(H \perp \{A, F, N\} | S)$$

after edge included

$$\text{Pa}_H = \{F, S\}$$

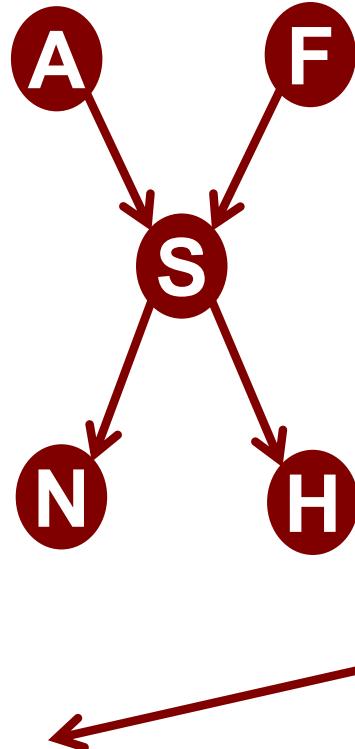
$$\text{NonDescendants}_S = \{A, N\}$$

$$(H \perp \{A, N\} | F, S)$$



Two independent events become conditionally dependent (negatively dependent) given that at least one of them occurs

Explaining Away / Berkson's Paradox



A variable X is independent of its non-descendants given its parents and only its parents ($X_i \perp \text{NonDescendants}_{X_i} \mid \text{Pa}_{X_i}$)

$$F \perp A$$

S is not a parent but a descendant

$F \perp A \mid S$ is not implied by local Markov assumption

- Two causes „compete“ to „explain“ the observed data
 - **Having a flu makes you less likely to have an allergy!!**
- $$P(A=t) \leq P(A=t|S=t, F=T) \leq P(A=t|S=t)$$



Explaining Away / Berkson's Paradox

Selection bias can induce a *false negative correlation* between independent traits.



Example:

A school admits students who are *either* academically or athletically excellent.

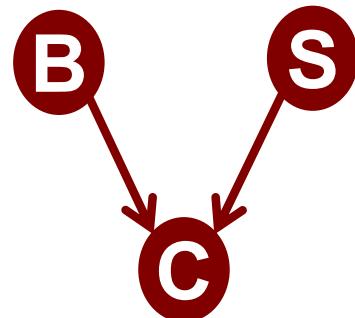
In the overall population, the two abilities are independent.



But among admitted students, they appear **negatively correlated** — strong athletes tend to be weaker academically, and vice versa — purely due to the **selection rule**.



Explaining Away / Berkson's Paradox

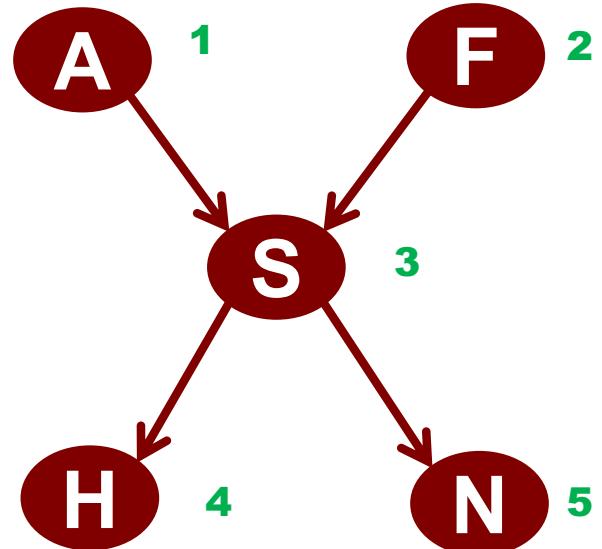


- A college which admits students who are either brainy or sporty (or both!)
 - Two causes „compete“ to „explain“ the observed data
-
- Look at a population of college students
 - **Being a brainy makes you less likely to be sportive and vice versa** because either property alone is sufficient to explain the evidence on C

$$P(S=t|C=t, B=T) \leq P(S=t|C=t)$$



How to come up with the Joint Distribution



Consider **topological orders**

Now, to interpret a BN

1. Choose particular **chain rule order**
2. Apply independence assumption

$$P(A, F, S, H, N) =$$

$$P(F)P(A)P(S | F, A)P(H | S)P(N | S)$$

$$P(A, F, S, H, N) = P(F)P(A | F)P(S | F, A)P(H | S, F, A)P(N | S, F, A, H)$$

$$\begin{array}{llll} A \perp F & P(S | F, A) & H \perp \{F, A, N\} | S & N \perp \{F, A, H\} | S \\ P(A) & & H \perp \{F, A\} | S & P(N | S) \\ & & P(H | S) & \end{array}$$

We can decompose due to the local Markov assumption



Definition: Bayesian Network

(Most important slide!)

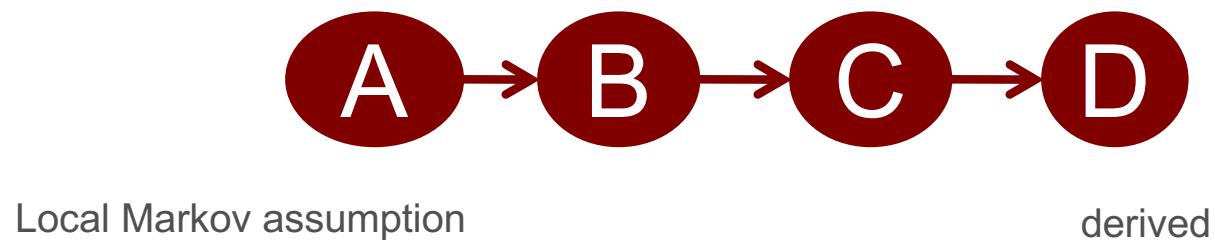
- Set of random variables $\{X_1, \dots, X_n\}$
- Directed acyclic graph
 - loops are ok but **no directed cycles**
- CPT with each X_i : $P(X_i | \text{Pa}(X_i))$
- Joint distribution $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}(X_i))$
- **Local Makov assumption**

A variable X is independent of its non-descendants given its parents and only ist parents: $(X_i \perp \text{NonDescendants}_{X_i} | \text{Pa}_{X_i})$



Awesome, now we know exactly what Bayesian networks are! But there are still some representational questions left

- What distributions can be represented by a BN?
 - What BNs can represent a distribution?
 - What are the independence assumptions encoded, next to the ones due to the local Markov assumption



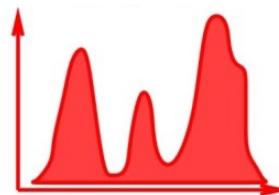
$$\{A, B\} \perp D | C$$

$$A \perp D | C$$



Independencies in Real Problems

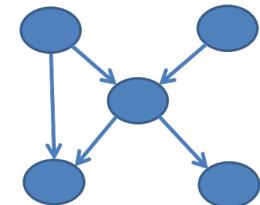
World, Data, Reality



True distribution P contains
independence assertions

$$I(P)$$

Model, BN



Graph G encodes local
independence assumptions

$$I_l(G)$$

Key representational assumption: $I_l(G) \subseteq I(P)$



The Representation Theorem



If conditional independencies in BN are a subset of conditional independencies in P, i.e., $I_l(G) \subseteq I(P)$

obtain

Then the joint probability distribution factorizes according to BN

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}(X_i))$$

Important: Every P has at least one BN structure G

If joint probability distribution factorizes according to BN

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}(X_i))$$

obtain

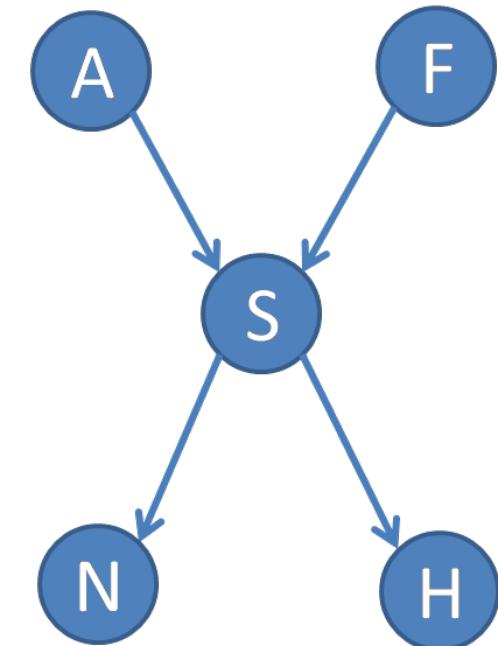
Important: Read independencies of P from BN structure G

Then conditional independencies in BN are a subset of conditional independencies in P, i.e., $I_l(G) \subseteq I(P)$



Local Markov Assumption & I-maps

A variable X is independent of its non-descendants given its parents and only its parents ($X_i \perp \text{NonDescendants}_{X_i} \mid \text{Pa}_{X_i}$)



- Local independence assumption in BN structure G : $I_l(G)$
- Independence assertion of P : $I(P)$
- **BN structure G is an I-map (independence map) if:**

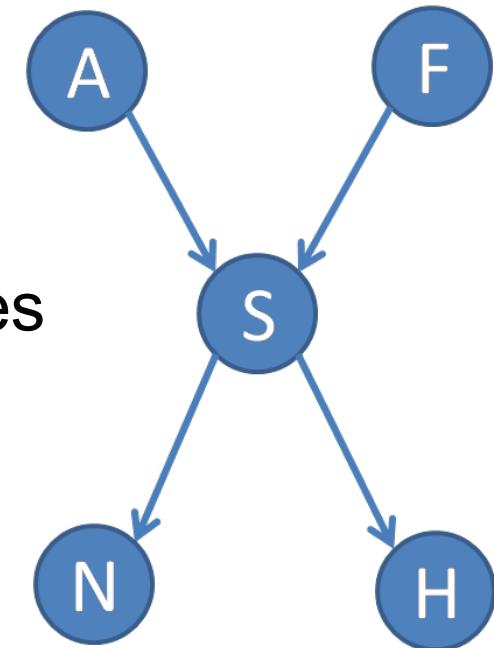
$$I_l(G) \subseteq I(P)$$



Factorized Distribution

- Given
 - Random variables $\{X_1, \dots, X_n\}$
 - P distribution over the same variables
 - BN structure G over the same variables
- P factorizes according to G if

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}(X_i))$$



The Representation Theorem



G is I-map of P



P factorizes according to G



BN Rep. Theorem: I-Map to Factorization

G is I-map of P obtain P factorizes according to G

- Start with a topological ordering, wlog X_1, \dots, X_n

- Apply chain rule

$$P(X_1, \dots, X_n) = P(X_1)P(X_2 | X_1) \dots P(X_n | X_1, \dots, X_{n-1})$$

- Consider $P(X_i | X_1, \dots, X_{i-1})$

- We know that $\text{Pa}(X_i) \subseteq \{X_1, \dots, X_{i-1}\}$, i.e., there are no descendants of X_i in X_1, \dots, X_{i-1}

- Hence, due to local Markov assumption

$$P(X_i | X_1, \dots, X_{i-1}) = P(X_i | \text{Pa}(X_i))$$



BN Rep. Theorem: I-Map to Factorization

P factorizes according to G → obtain G is I-map of P

- Most likely done in the exercise session, ☺
- We have to show that the local Markov independence assumptions that hold in G also hold in P:

$$P(X_i | \text{NonDes}(X_i), \text{Pa}(X_i)) = P(X_i | \text{Pa}(X_i))$$

- Then apply definition of cond. prob. and simplify expressions



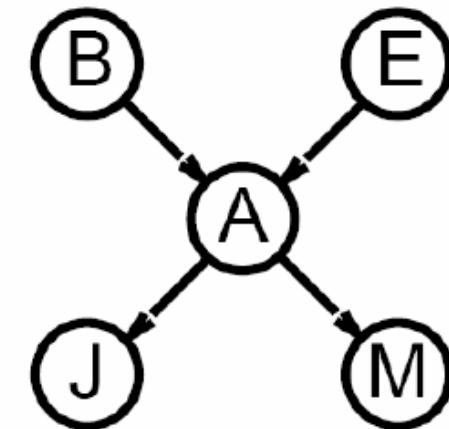
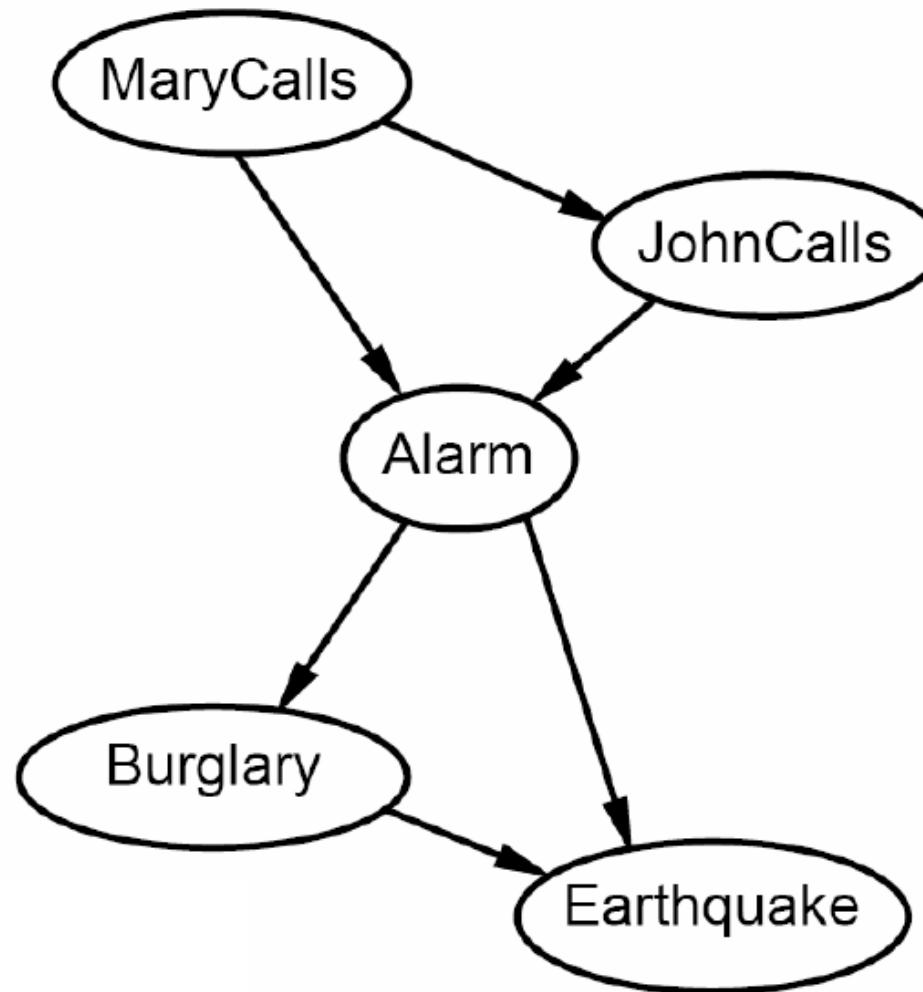


How do we come up with a Bayesian Network?

- **Given** a set of variables $\{X_1, \dots, X_n\}$ and conditional independence assertions of P (estimated from data)
- **Choose** an ordering on variables, i.e., X_1, \dots, X_n
- **For $i = 1$ to n**
 - Add X_i to the network
 - Define parents $\text{Pa}(X_i)$ of X_i in graph as the minimal subset of $\{X_1, \dots, X_{i-1}\}$ such that local Markov assumption holds
 - Define/learn CPT: $P(X_i | \text{Pa}(X_i))$



Be careful! There are typically many Bayesian Networks for a joint distributions



This all depends
on the ordering
you are choising

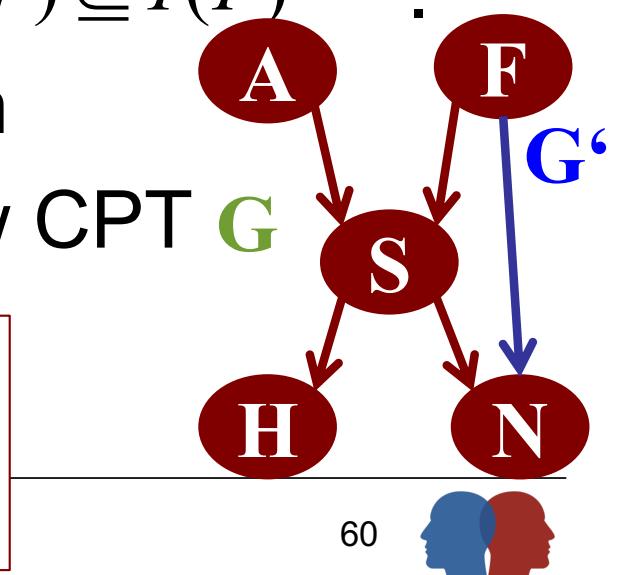


Adding edges does not hurt

(Third most important slide!)

- **Theorem:** Let G be an I-map for P , any DAG G' that includes the same directed edges as G is also an I-map for P .
 - **Corollary 1:** G' is strictly more expressive than G
 - **Corollary 2:** If G is an I-map for P , then adding edges still results in an I-map
- **Proof idea** for if $I_l(G) \subseteq I(P)$ then $I_l(G') \subseteq I(P)$.
We show $I_l(G) \supseteq I_l(G')$ by induction
Note that we are free to set the new CPT

$$\begin{aligned} P(X_i | Pa(X_i), X' = t) &= P(X_i | Pa(X_i), X' = f) \\ &= P(X_i | Pa(X_i)) \end{aligned}$$



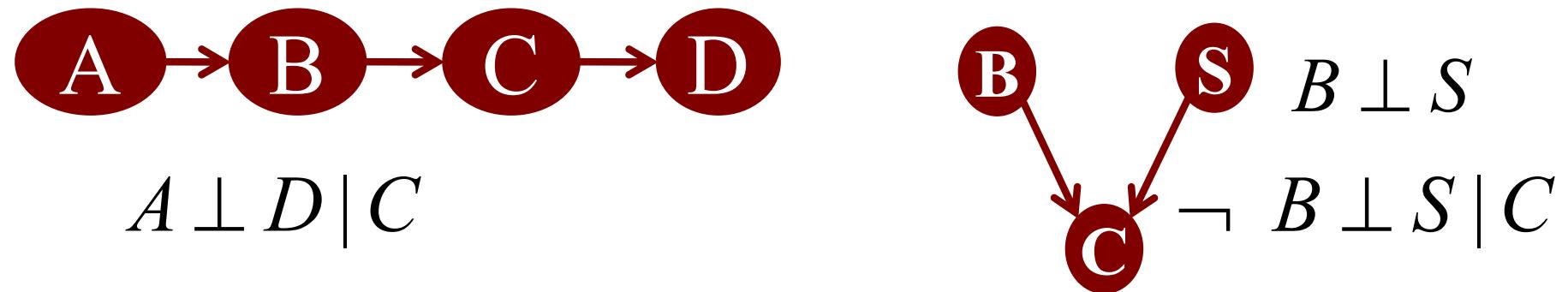
What you need to know thus far

- Independence and conditional independence
- Definition of a Bayesian network
- Local Markov assumption
- The representation theorem
 - G is I-map for P iff P factorizes according to G
 - Interpretation



Independencies encoded in BN

- We said: all you need is the local Markov assumption
($X_i \perp \text{NonDescendants}_{X_i} \mid \text{Pa}_{X_i}$)
- But then we talked about other (in)dependencies such as explaining away



- So, what are the independencies encoded by a BN?
 - Only assumption is local Markov but many other can be derived using the algebra of conditional independencies!

