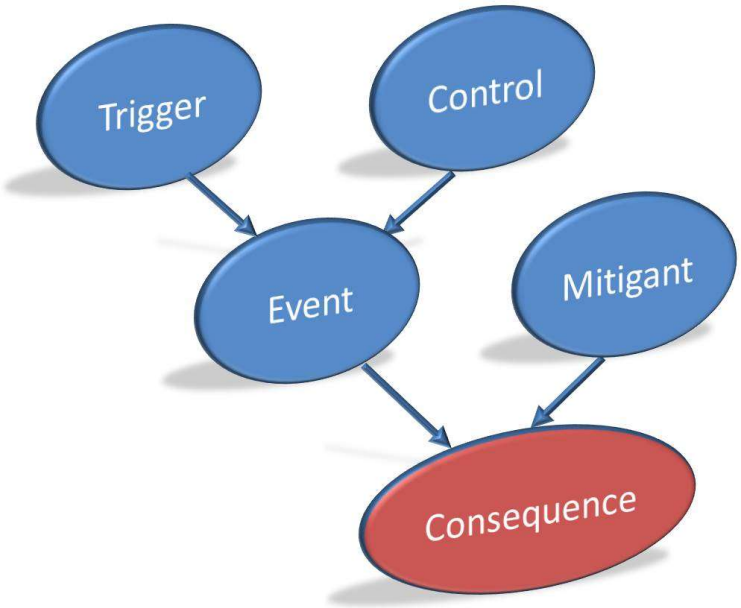


Risk and Decision Making for Data Science and AI

LESSON 9 Learning from Data: limitations and how to resolve them

Norman Fenton

@ProfNFenton

Performance of different algorithms on the Titanic test data

Method	Accuracy	AUC
Coin toss (i.e. randomly assign 50% survive)	0.500	0.500
Randomly assign 37% survive	0.533	0.500
All females survive all males don't	0.786	0.578
Simple classification tree	**0.806**	0.819
Over-fitted classification tree	**0.806**	0.810
Naïve Bayes	0.798	0.824
Bayesian network	0.802	0.836
Logistic regression	0.798	0.824
Random forest	0.799	**0.850**
Support vector machine	0.782	0.825
Neural network	0.794	0.828
Averaged neural network	0.794	0.837
K-nearest neighbour	0.774	0.812

Very simple methods

Serious methods

Little to choose between

But it is not just about prediction….

All models produce a similar prediction of the outcome for any passenger once we know their attributes (Sex, Class, Age, etc.)

What happens if some of the attribute values are missing?

- **Most of these models cannot produce a prediction at all, and those that do will be wrong.**

These models cannot answer simple questions like:

- **If I know that a passenger survived - and that the passenger was male - what is the probability the passenger was in 1st Class?**

Causal models can answer these questions.

- **But the causal models must reflect reality.**

Pearl's example: data for potential outcomes

Employee (u)	Experience (u)	Education(u)*	$Salary_0(u)$	$Salary_1(u)$	$Salary_2(u)$
Alice	6	0	$81,000	?	?
Bert	9	1	?	$92,000	?
Caroline	9	2	?	?	$97,000
David	8	1	?	$91,000	?
Ernest	12	1	?	$100,000	?
Frances	13	0	$97,000		?
etc	…	…	…	…	…

* 0 = high school
  1 = college degree
  2 = graduate degree

$Salary_i(u)$ represents salary if *u* had education level *i*

Each "?" represents a counterfactual question: what would *u*'s salary have been if *u* had education level *i*?

Table is devoid of causal information: e.g. does education affect salary or the other way round? Does education affect experience or the other way round? Table does not allow us to represent such information

Statisticians regard "?"s not as potential outcomes but 'missing values' to be 'imputed'.
There are many interpolation techniques to do this 'imputation'.

Pearl's example: data for potential outcomes

Employee (u)	Experience (u)	Education(u)*	$Salary_0(u)$	$Salary_1(u)$	$Salary_2(u)$
Alice	6	0	$81,000	?	?
Bert	9	1	?	$92,000	? $97,000
Caroline	9	2	?	? $92,000	$97,000
David	8	1	?	$91,000	?
Ernest	12	1	?	$100,000	?
Frances	13	0	$97,000		?
etc	…	…	…	…	…

* 0 = high school
  1 = college degree
  2 = graduate degree

$Salary_i(u)$ represents salary if *u* had education level *i*

Typical imputation technique is "Matching": Look for pairs of records that match closely on all but the 'missing' values.
e.g. Bert and Caroline have same years of experience
So we could conclude that Bert's salary would have been $97,000 if, like Caroline he had a grad degree
And Caroline's salary would have been $92,000 if, like Bert she had only a college degree
But what about Alice for whom there is no close match?
There are many statistical approaches to do it, but the 'correct' answer depends critically on whether education affects experience or the other way round and this cannot be learnt' from the data

Pearl's example: data for potential outcomes

Employee (u)	Experience (u)	Education(u)*	$Salary_0(u)$	$Salary_1(u)$	$Salary_2(u)$
Alice	6	0	$81,000	? $85,000	? $90,000
Bert	9	1	?	$92,000	? $97,000
Caroline	9	2	?	? $92,000	$97,000
David	8	1	?	$91,000	?
Ernest	12	1	?	$100,000	?
Frances	13	0	$97,000		?
etc	…	…	…	…	…

* 0 = high school
1 = college degree
2 = graduate degree

$Salary_i(u)$ represents salary if *u* had education level *i*

Could use a regression based method to arrive at an equation like:

$$\text{Salary} = 65000 + 2500 \times \text{Experience} + 5000 \times \text{Education}$$

According to this we predict

$$\text{Salary}_1(\text{Alice}) = 65000 + 2500 \times 6 + 5000 \times 1 = 85000$$

$$\text{Salary}_2(\text{Alice}) = 65000 + 2500 \times 6 + 5000 \times 2 = 90000$$

Pearl's example: data for potential outcomes

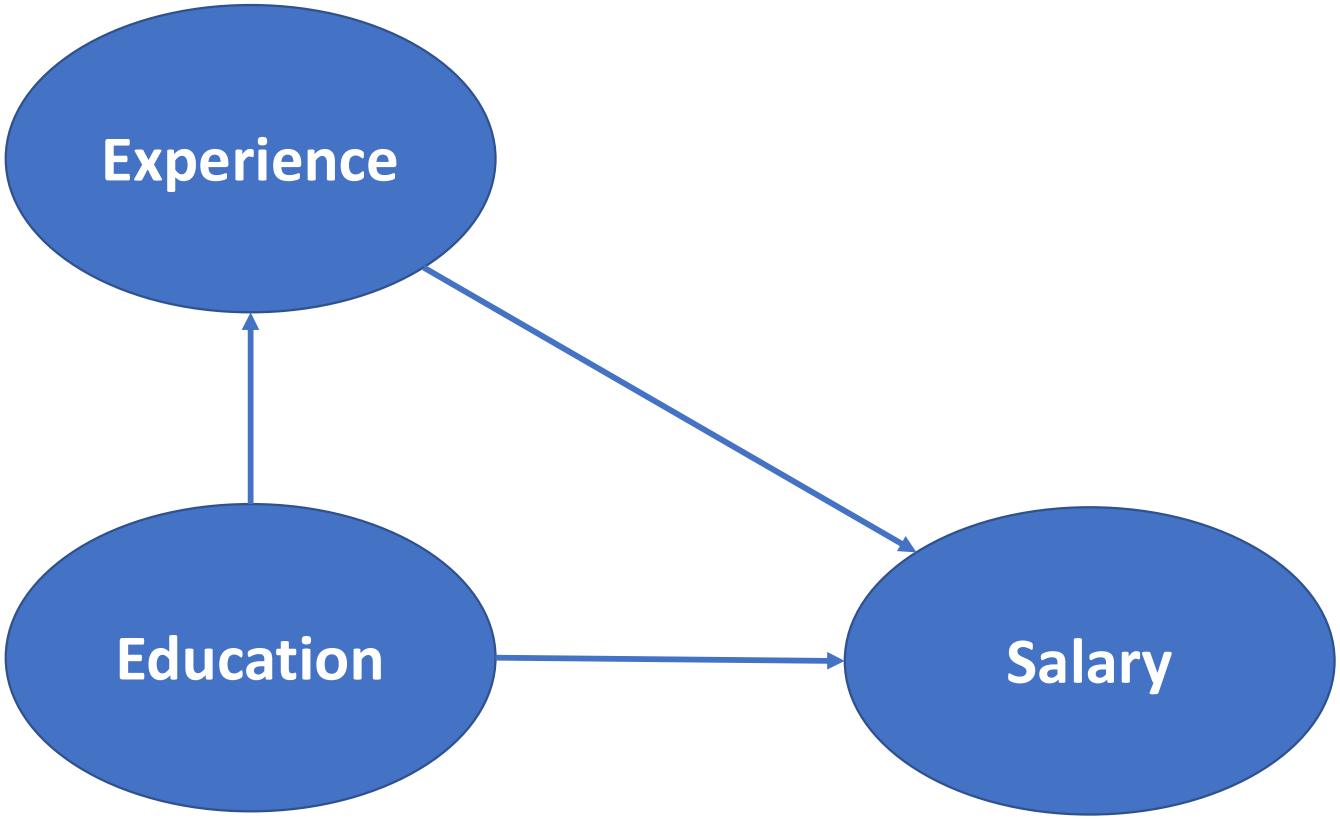
Employee (u)	Experience (u)	Education(u)*	$Salary_0(u)$	$Salary_1(u)$	$Salary_2(u)$
Alice	6	0	$81,000	? $85,000	? $90,000
Bert	9	1	?	$92,000	? $97,000
Caroline	9	2	?	? **>$97,000**	$97,000
David	8	1	?	$91,000	?
Ernest	12	1	?	$100,000	?
Frances	13	0	$97,000		?
etc	…	…	…	…	…

* 0 = high school
  1 = college degree
  2 = graduate degree

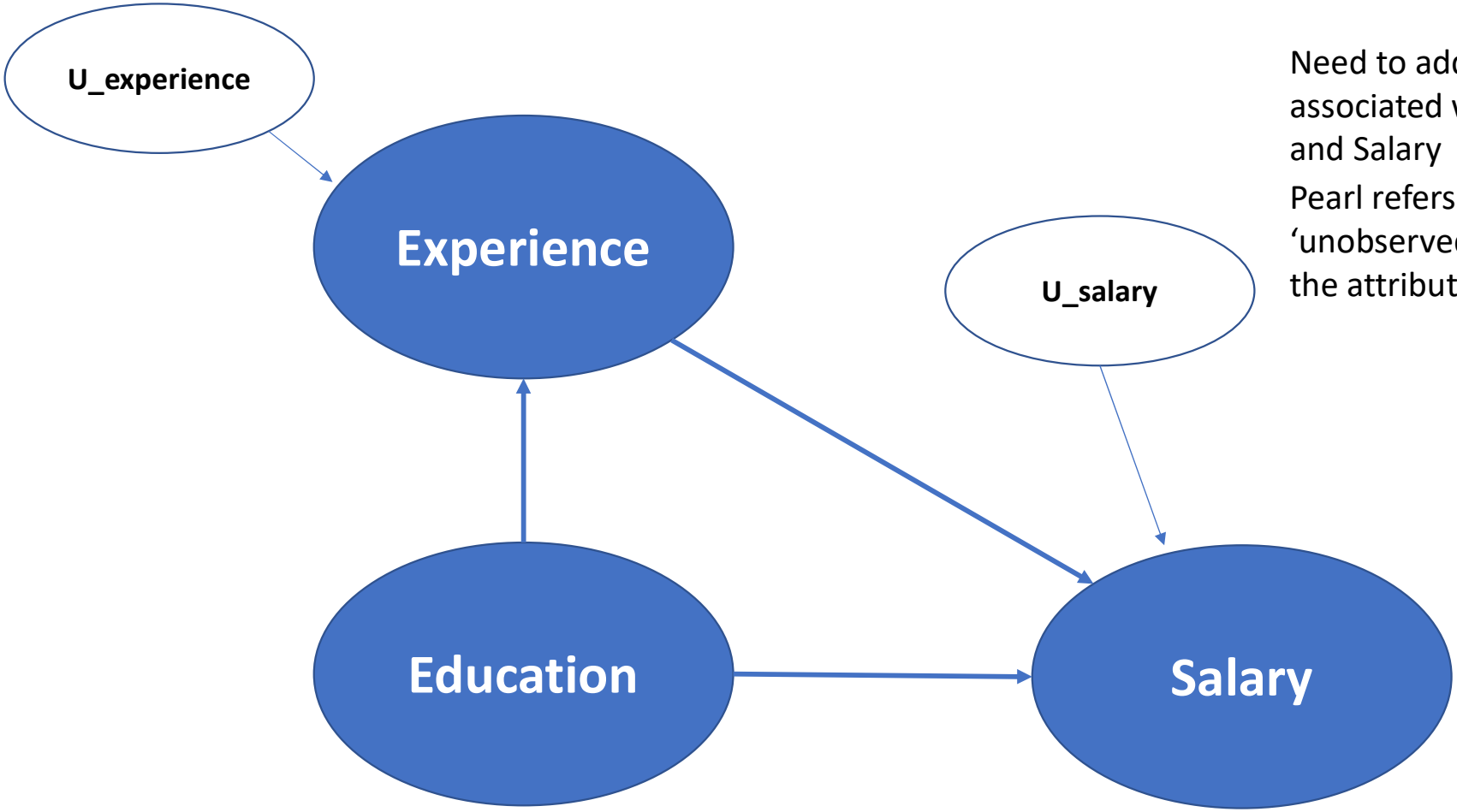
$Salary_i(u)$ represents salary if *u* had education level *i*

BUT: These methods fail because we KNOW experience depends on education.
(In the USA) each degree takes 4 years.
If Caroline had used the 4 years she spent on a graduate degree to work instead, she would have had 13 years of experience rather than 9.
That would have given her the same education as Bert but 4 more years experience.
So we should conclude $Salary_1$(Caroline)>$97,000 since Frances earned $97,000 with 13 years of experience and no college degree

Pearl's example: data for potential outcomes



Pearl's example: data for potential outcomes



Need to add 'exogenous' variables associated with each Experience and Salary

Pearl refers to these as 'unobserved variables that affect the attribute'.

Pearl's example: data for potential outcomes

The structural equation model solution

$$\text{Equation 1:} \quad \text{Salary} = 65000 + 2500 \times \text{Experience} + 5000 \times \text{Education} + U_{\text{Salary}}$$

$$\text{Equation 2:} \quad \text{Experience} = 10 - 4 \times \text{Education} + U_{\text{Experience}}$$

Step 1 (Abduction) Use data about Alice to estimate her special features (the U values):

We know Salary(Alice) = 81000, Experience(Alice)=6 and Education(Alice)=0

So from Equation 1 we know U_{Salary}(Alice)= 1000

So from Equation 2 we know $U_{Experience}$(Alice)= - 4

This represents everything 'unique' about Alice and, whatever it is, it adds $1000 to her predicted salary

Step 2 (Action) Change model to reflect counterfactual assumption (intervention)

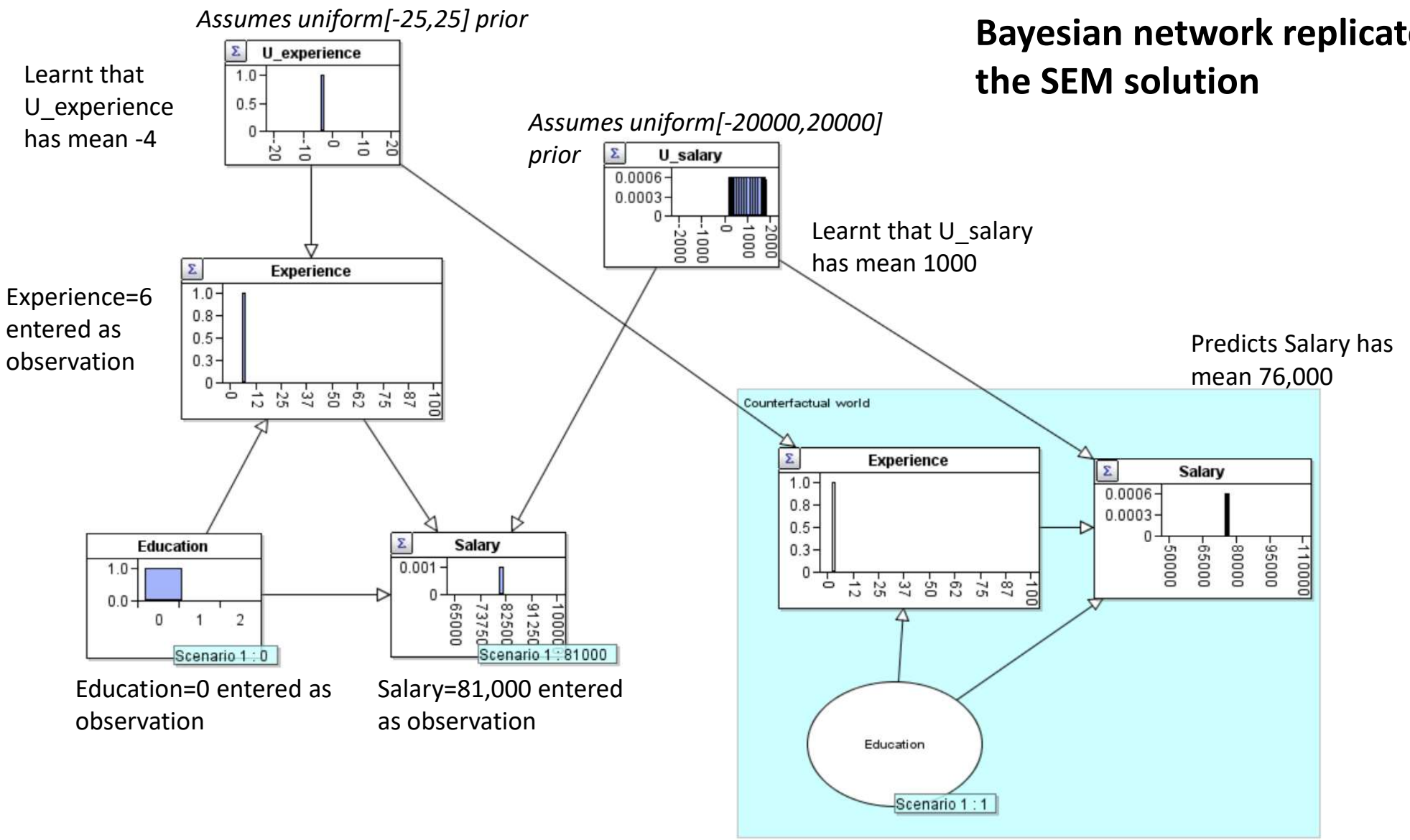
In this case it simply means assigning Education = 1

Step 3 (Prediction) Calculate Alice's new salary based on the equations with the 'learnt' U values and step 2 values. So, in the counterfactual world

From Equation 2: Experience(Alice) = 10 – 4 – 2 = 2

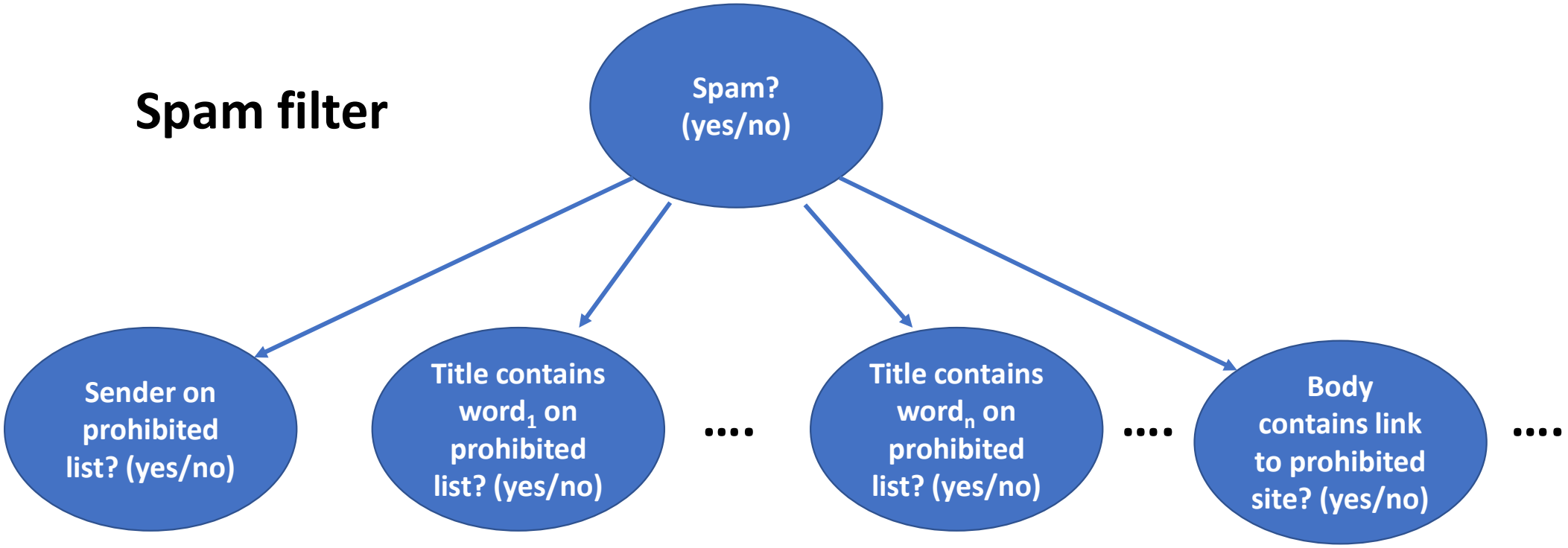
Hence, from Equation 1: Salary(Alice) = 65000 + 5000 + 5000 + 1000 = 76000

Bayesian network replicates the SEM solution

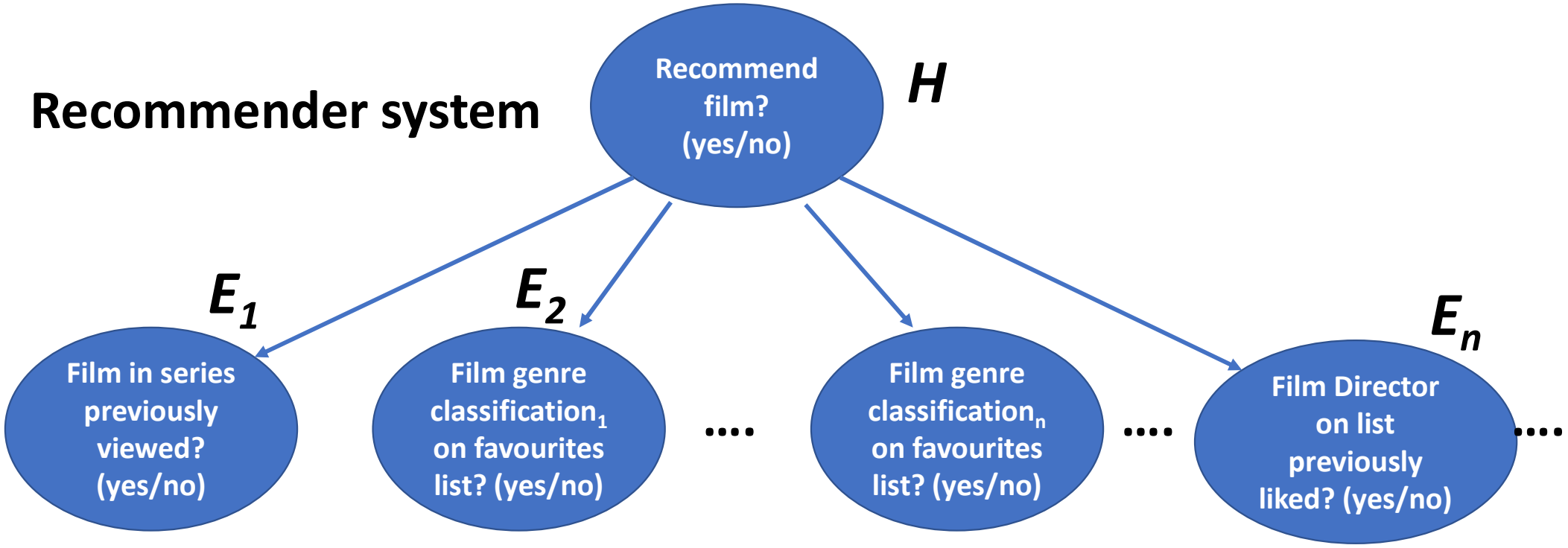


Naïve Bayes versus Causal Model

Spam filter



Naïve Bayes versus Causal Model



You could do all the necessary calculations without a BN tool. You just need these Bayes formulas:

$$P(H|E_1, E_2, \dots, E_n) = \frac{P(E_1, E_2, \dots, E_n|H) \times P(H)}{P(E_1, E_2, \dots, E_n)} = \frac{P(E_1|H) \times P(E_2|H) \dots \times P(E_n|H) \times P(H)}{P(E_1, E_2, \dots, E_n)}$$

$$= \frac{P(E_1|H) \times P(E_2|H) \dots \times P(E_n|H) \times P(H)}{P(E_1|H) \times P(E_2|H) \dots \times P(E_n|H) \times P(H) + P(E_1|not\ H) \times P(E_2|not\ H) \dots \times P(E_n|not\ H) \times P(not\ H)}$$

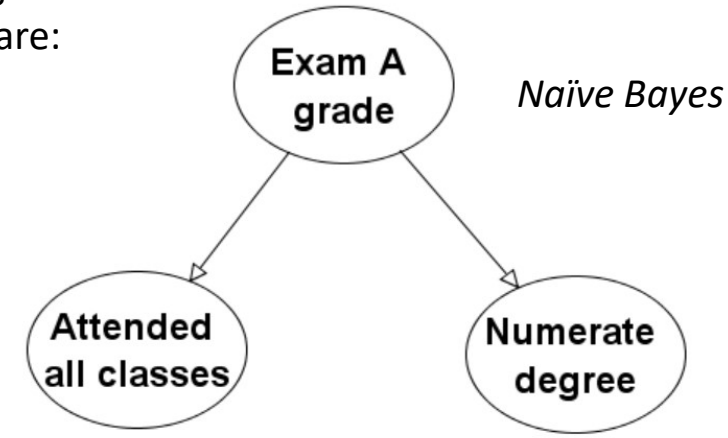
Naïve Bayes versus Causal Model

Example: Data reveals that the two most discriminating attributes for predicting whether or not a student on a particular MSc module achieves an A grade in their exam are:

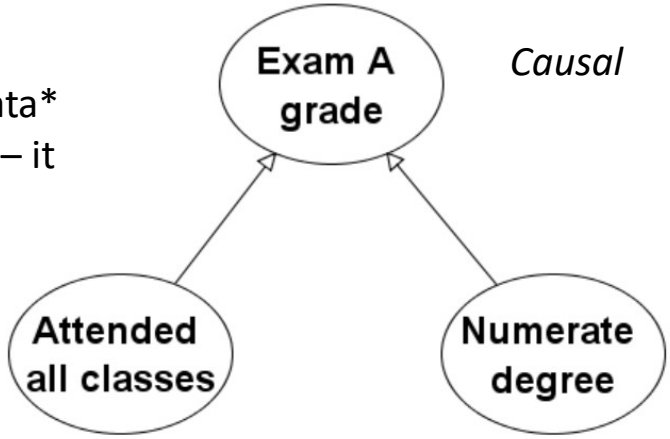
- Having a numerate undergrad degree
- Attending all classes on the module

We have the data for past students:

Attended	Numerate	exam A
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	FALSE
TRUE	FALSE	FALSE
FALSE	FALSE	FALSE
TRUE	FALSE	TRUE
TRUE	TRUE	TRUE
TRUE	TRUE	FALSE
…	…	…



Naïve Bayes



Causal

We could use a naïve Bayes model and learn the probability tables from the data* (note: in this context naïve Bayes is essentially like all the other ML algorithms – it produces similar results)

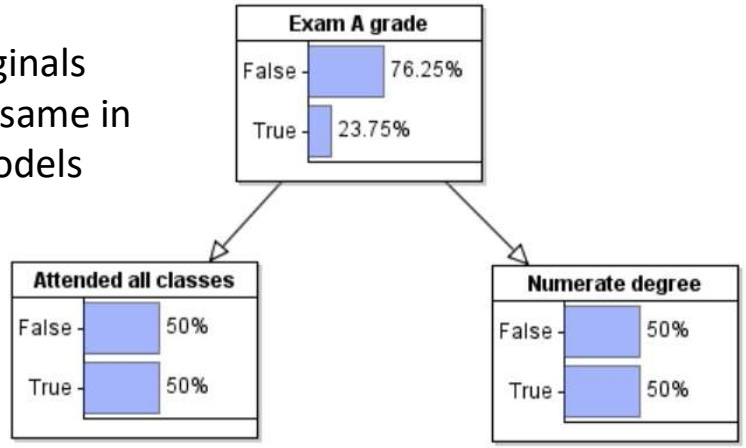
But this model is causally 'incorrect' in several ways. Instead we could use a 'correct' causal model and learn the probability tables from the data

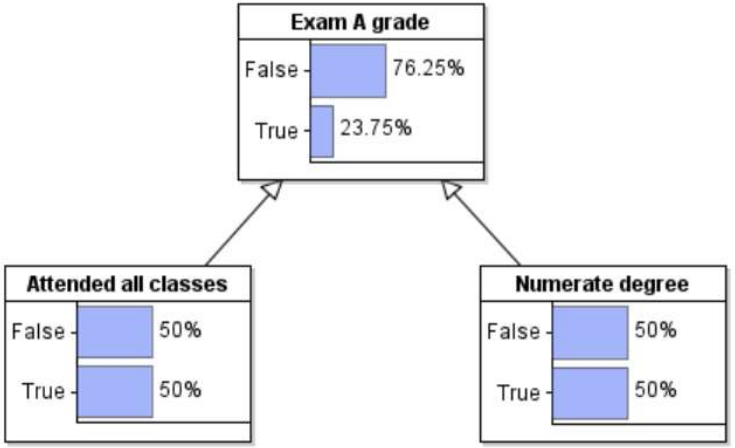
*DATA file: "exam pass naïve Bayes." – contains 80 records

Naïve Bayes versus Causal Model

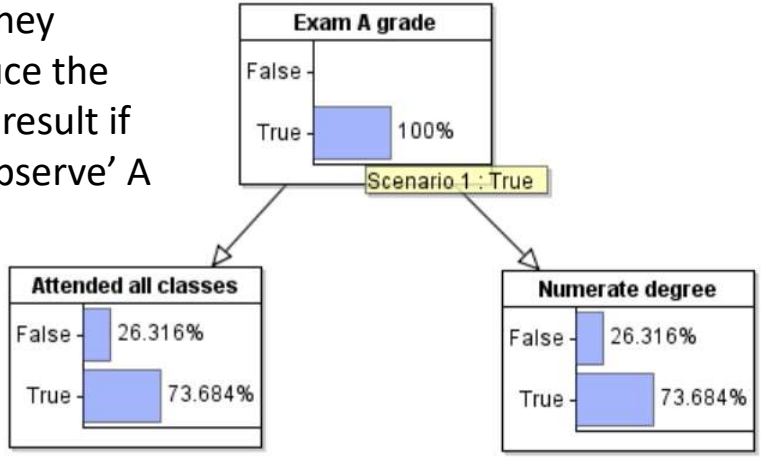
For both models use the table learning on the dataset

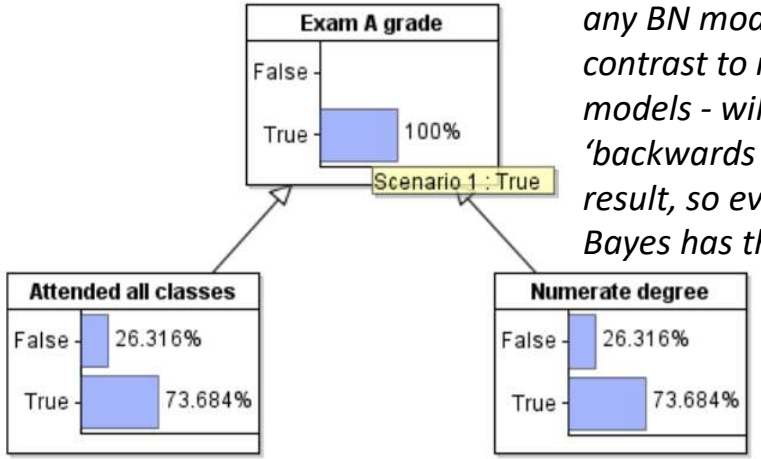
All marginals are the same in both models





And they produce the same result if we 'observe' A grade



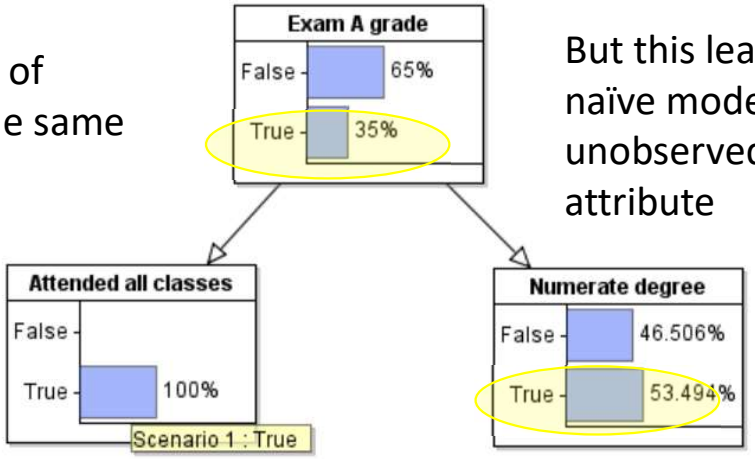


(As an aside: note that any BN model – in contrast to regression models - will provide a 'backwards inference' result, so even the naïve Bayes has this advantage)

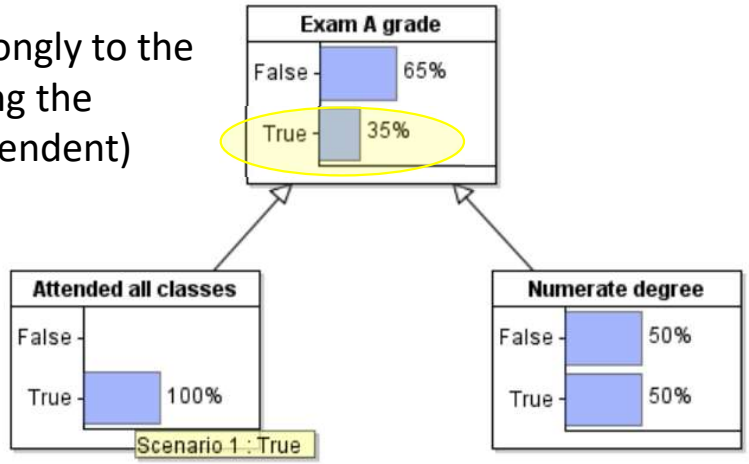
Naïve Bayes versus Causal Model

But when we use the model for prediction things start to change

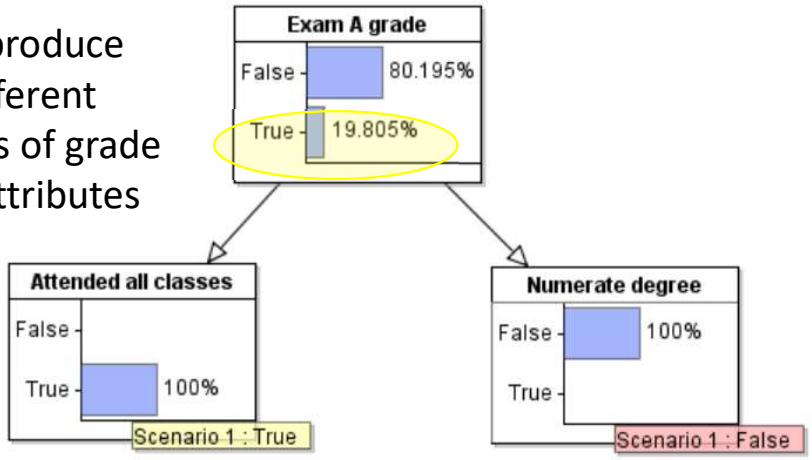
Prediction of grade is the same

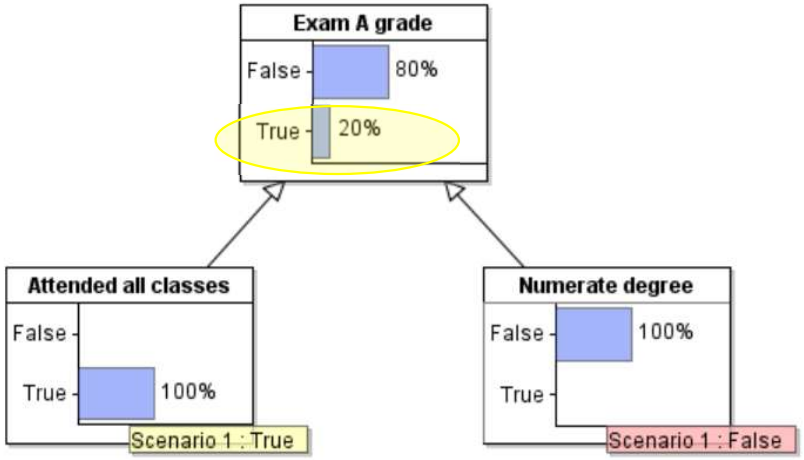


But this leads – wrongly to the naïve model revising the unobserved (independent) attribute



And they produce slightly different predictions of grade when all attributes observed

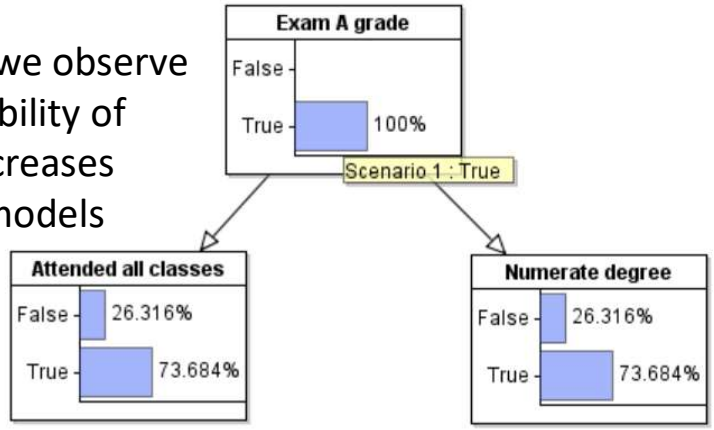


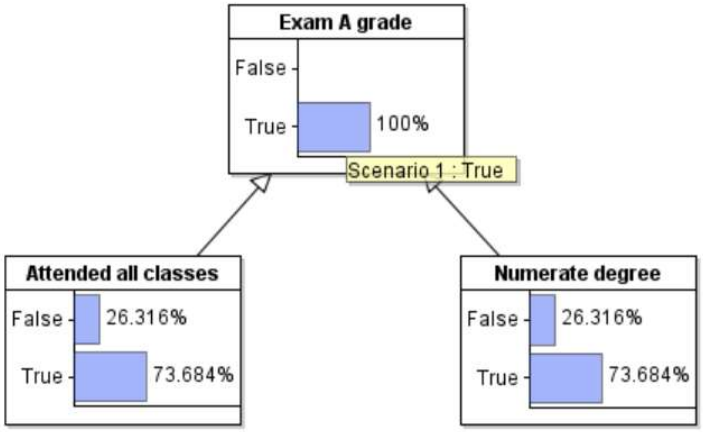


Naïve Bayes versus Causal Model

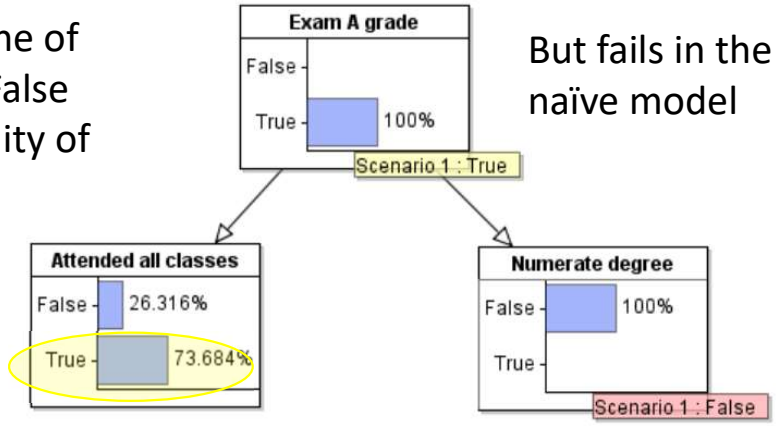
Maybe those ‘issues’ do not sound too serious but there are more serious problems with the naïve model

As we saw: when we observe grade A the probability of both attributes increases (equally) in both models

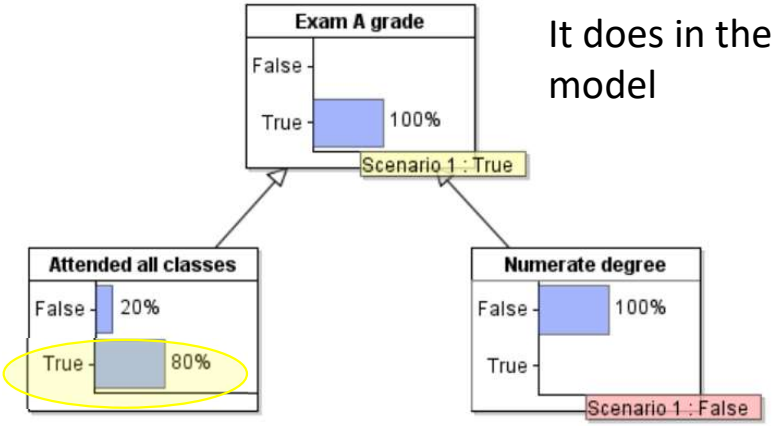




But if we know one of the attributes is False then the probability of the other should increase



But fails in the naïve model

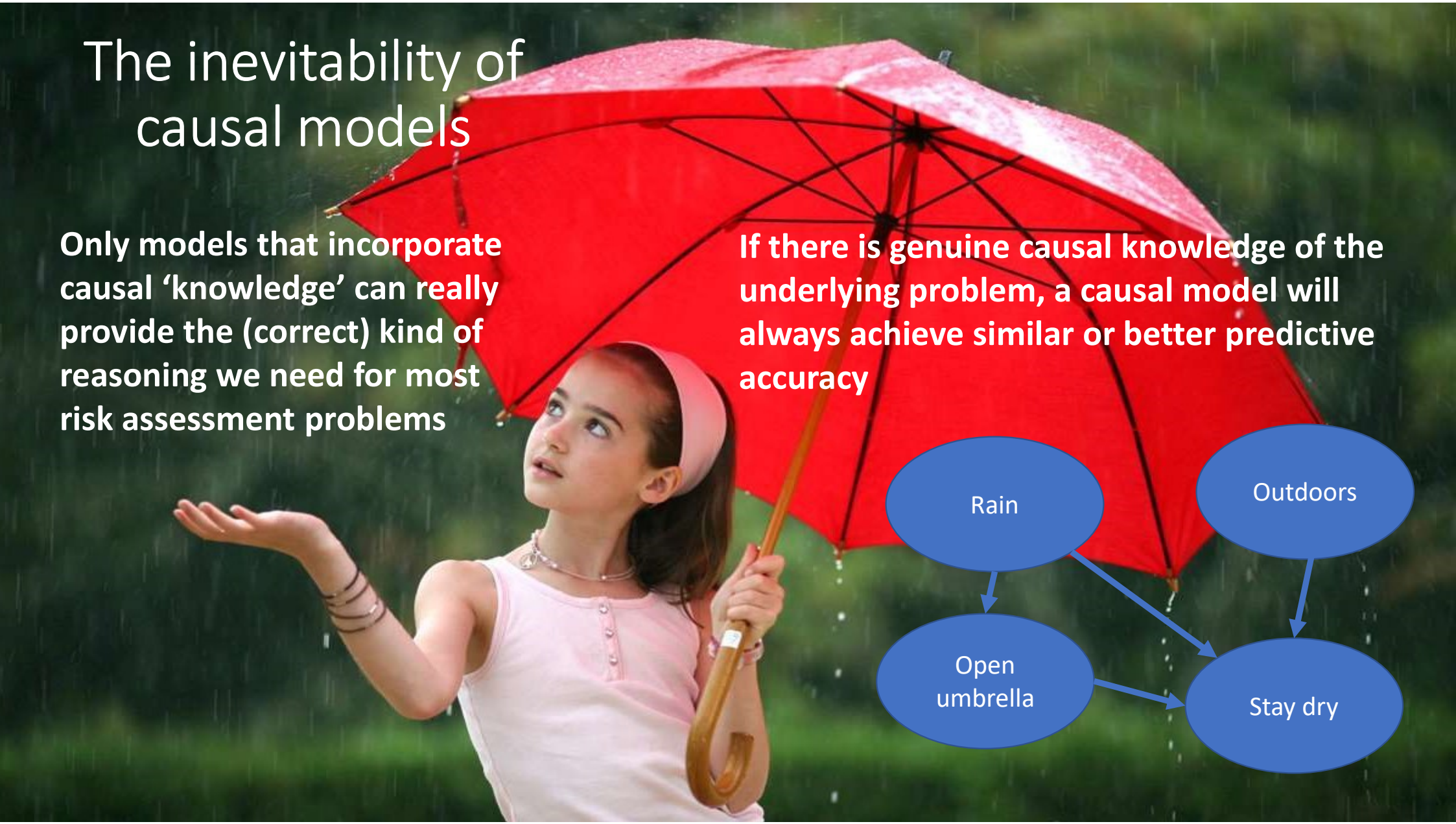


It does in the causal model

The inevitability of causal models

Only models that incorporate causal 'knowledge' can really provide the (correct) kind of reasoning we need for most risk assessment problems

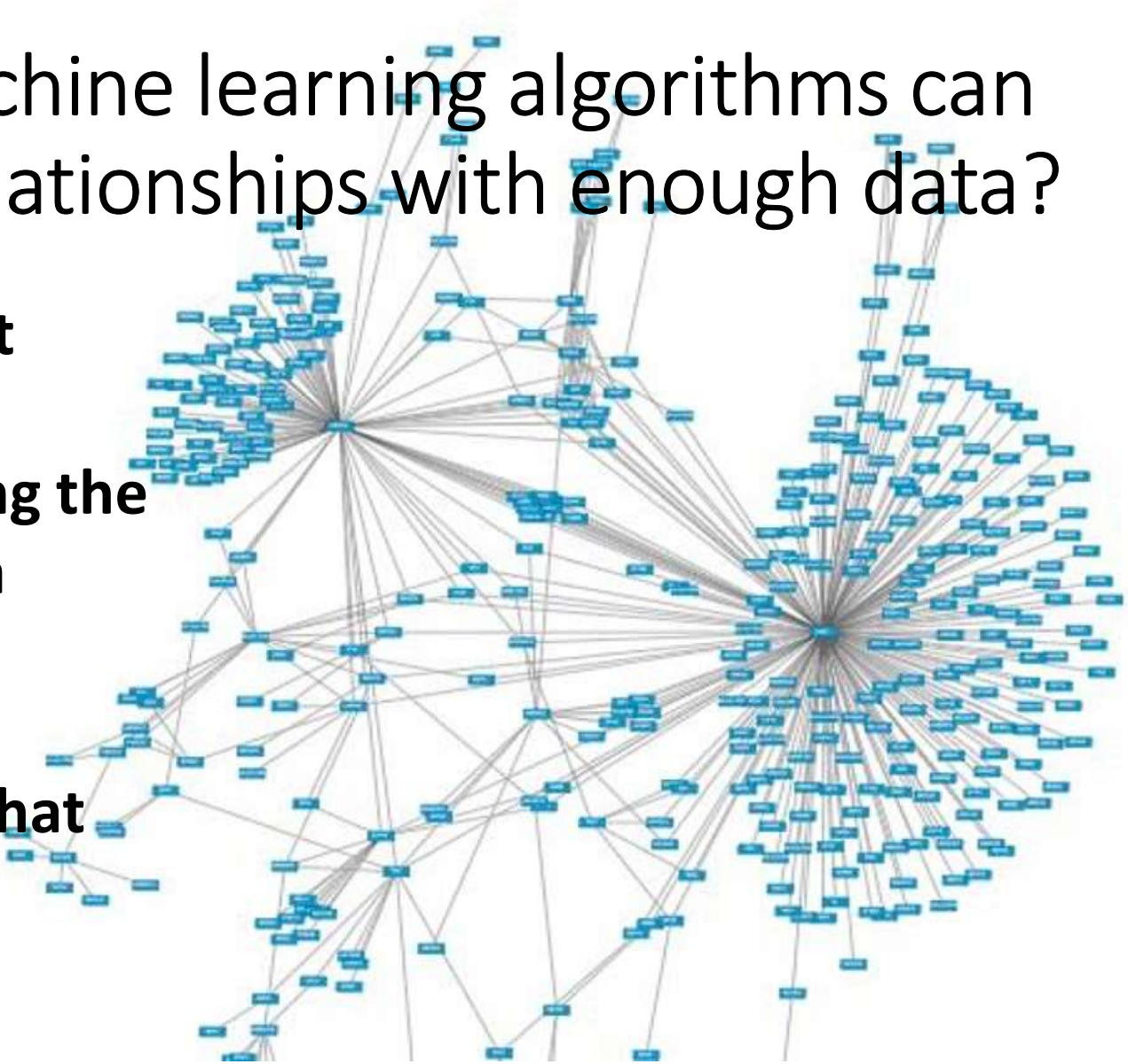
If there is genuine causal knowledge of the underlying problem, a causal model will always achieve similar or better predictive accuracy



But surely machine learning algorithms can learn causal relationships with enough data?

There is a lot of current research on 'causal discovery' – i.e. learning the BN structure from data

But there are some fundamental barriers that CANNOT be overcome

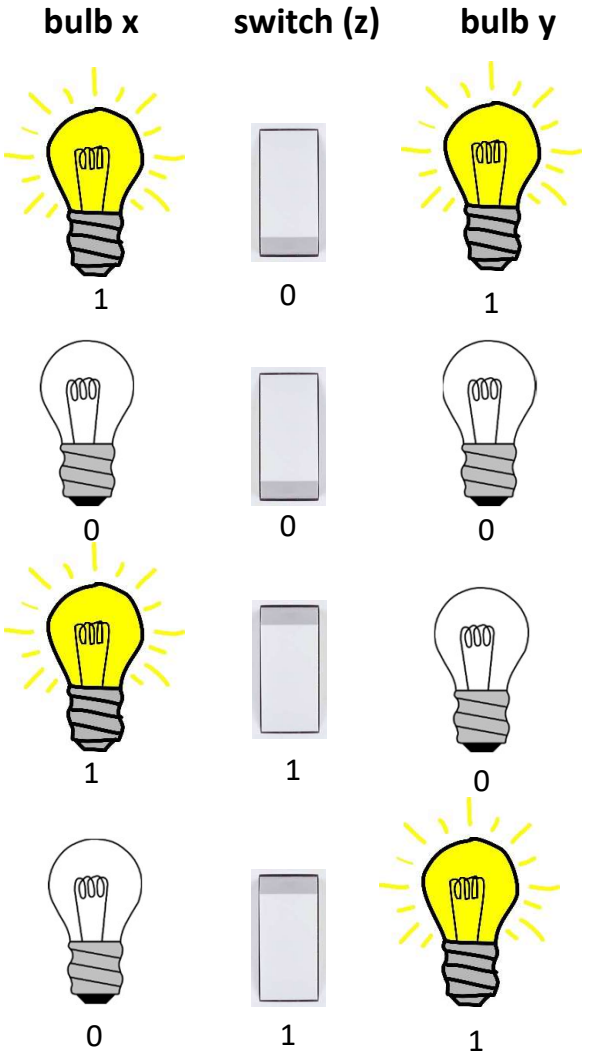


Is it possible to have correlation without causation?

Is it possible to have causation without correlation?

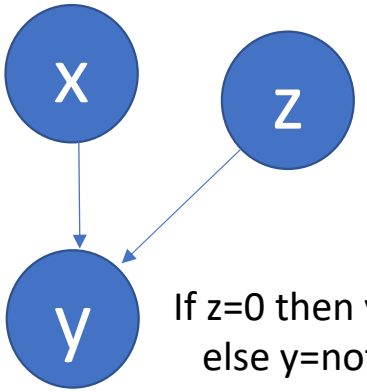
x	Outcome y
1	1
0	0
1	0
0	1
1	0
0	0
1	1
1	0
0	1
0	0
0	1
1	0
1	1
0	1
1	0
1	1
0	0
0	1

And there is a switch z ....



The 'hidden' switch provides a completely (deterministically) ***causal relationship*** between x and y

Once we know the switch value (on/off), the value of y is completely determined by the value of x

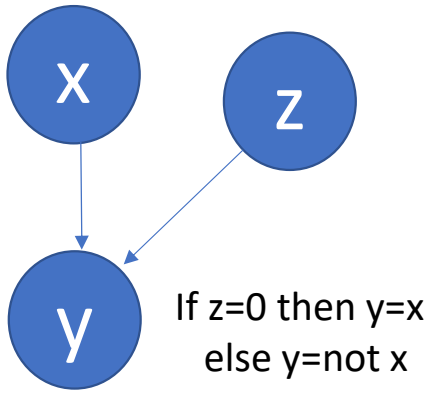


If z=0 then y=x
else y=not x

x	Outcome y
1	1
0	0
1	0
0	1
1	0
0	0
1	1
1	0
0	1
0	0
0	1
1	0
1	1
0	1
1	0
1	1
0	0
0	1

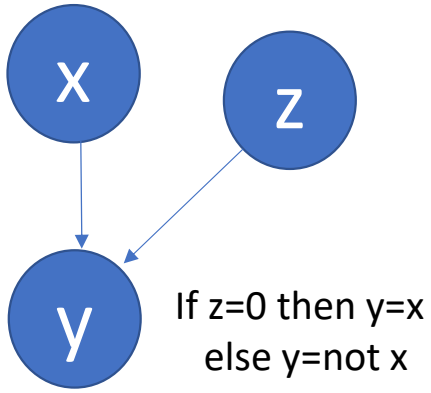
Machine Learning from data can never 'learn' this causal relationship

'Knowledge' with data can



x	Outcome y
1	1
0	0
1	0
0	1
1	0
0	0
1	1
1	0
0	1
0	0
0	1
1	0
1	1
0	1
1	0
1	1
0	0
0	1

…and this example also demonstrates how we can have *causation without correlation*



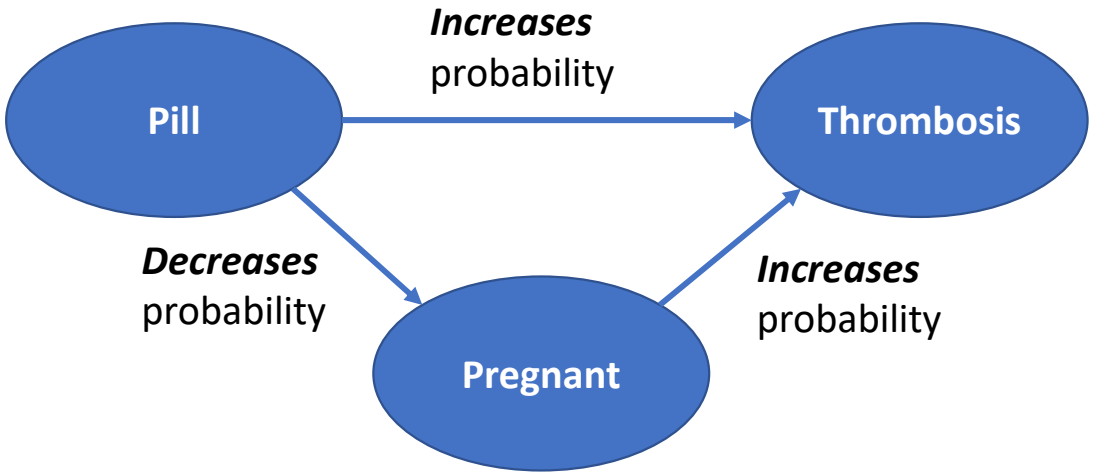
A more serious example of when an important causal relationship may remain completely 'hidden' from data-driven 'learning'

The following are known medical facts:

Taking the contraceptive pill **reduces** the probability of pregnancy

Pregnancy can **cause** thrombosis

Taking the contraceptive pill can **cause** thrombosis

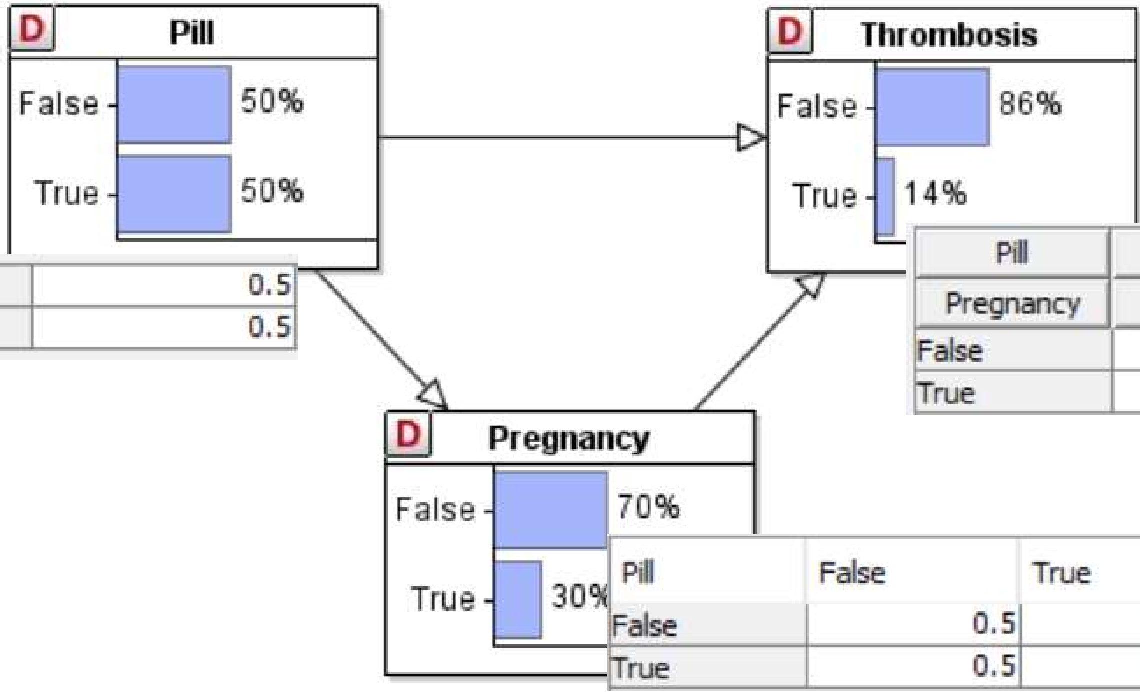


Question: If we are looking ONLY at the effect of the Pill on thrombosis what conclusions could we draw?

Knowing the causal structure we can use data to learn the parameter values

The data file “thrombosis.csv” contains 200 records of women of child-bearing age with the relevant data.

We load this data file to learn the probability tables for the associated Bayesian network model



False	0.5
True	0.5

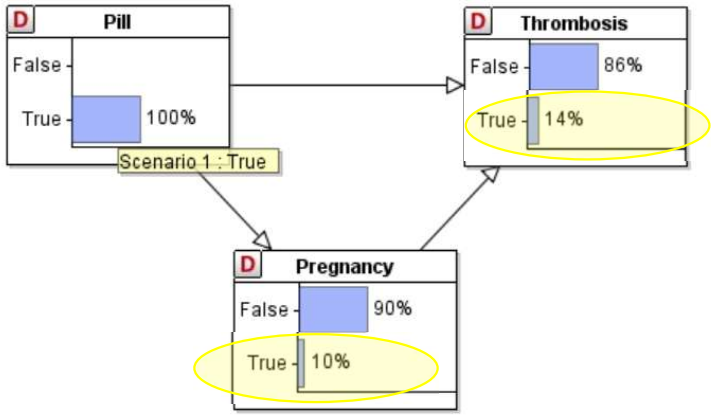
Pill	False		True	
Pregnancy	False	True	False	True
False	0.92	0.8	0.9	0.5
True	0.08	0.2	0.1	0.5

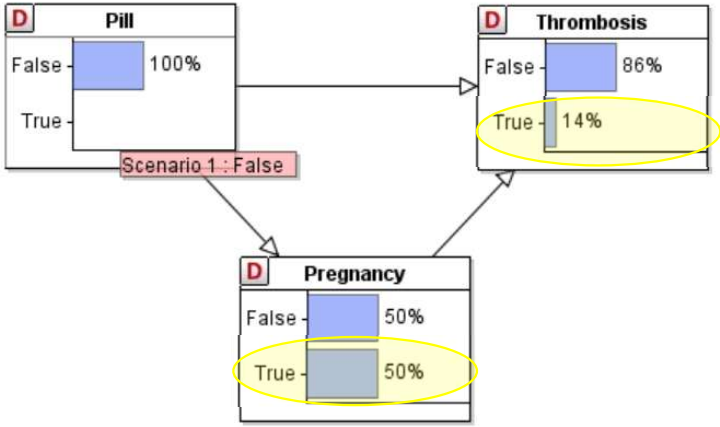
Pill	False	True
False	0.5	0.9
True	0.5	0.1

Pill	Pregnancy	Thrombosis
TRUE	TRUE	TRUE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	TRUE	FALSE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	TRUE
TRUE	FALSE	FALSE
TRUE	FALSE	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
[illegible]	[illegible]	FALSE
TRUE	FALSE	FALSE
TRUE	FALSE	FALSE
TRUE	FALSE	FALSE
TRUE	FALSE	FALSE
TRUE	FALSE	FALSE
FALSE	TRUE	TRUE
FALSE	TRUE	TRUE
FALSE	TRUE	TRUE

....

The model is “unfaithful” to the effect of Pill





Look ONLY at the effect of the Pill on Thrombosis

Looking at people who take the pill

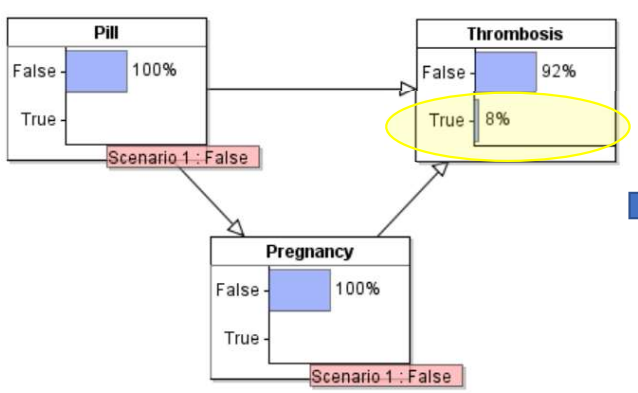
And those that don’t

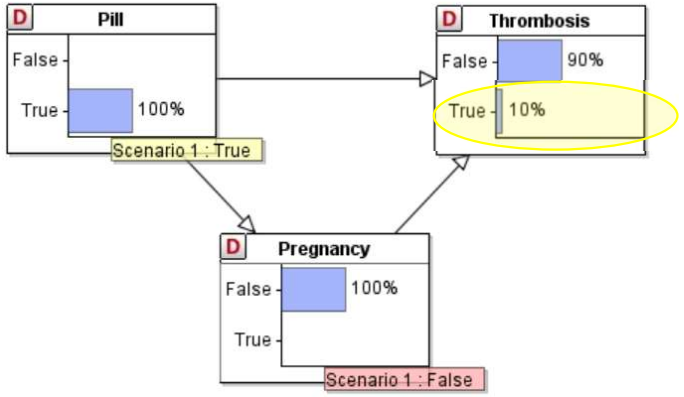
There seems to be no effect at all

This is because the pill's positive effect on thrombosis is cancelled out by its negative effective on pregnancy

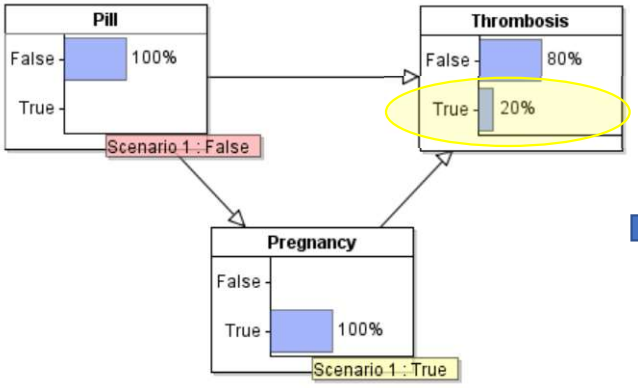
Formally, the model becomes unfaithful to its effect since $p(Thrombosis)=p(Thrombosis|Pill)$, even though there is a direct arc between them.

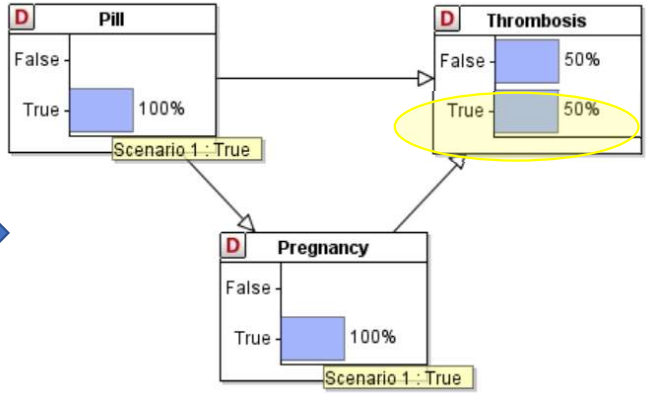
But the structural model gives the correct results…..





When pregnancy is False the probability of thrombosis goes up from 8% to 10% if taking the pill



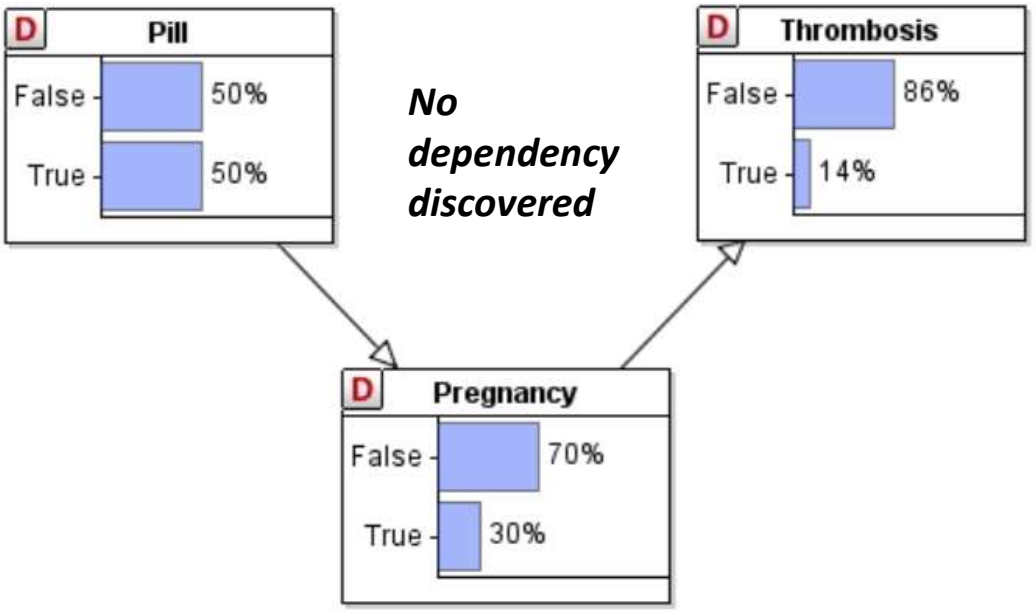


When pregnancy is True the probability of thrombosis goes up from 20% to 50%

So, in both cases (pregnancy true and false) the probability of thrombosis increases when taking the pill

Causal ‘discovery’ algorithms will generally fail for such examples

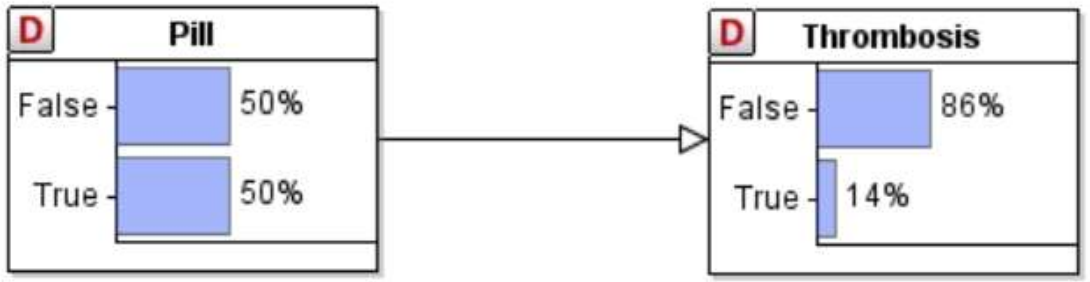
Algorithms that attempt to ‘discover’ causal relationship from data will generally be incapable of ‘learning’ that there is a direct dependency between Pill and Thrombosis



This model is not only causally flawed but it provides flawed predictions.

Ignoring Pregnancy also leads to flawed model - even if the causal link from Pill to Thrombosis is assumed

Using the same dataset to 'learn' the effect of Pill on Thrombosis directly – i.e. not considering Pregnancy – will also lead to a flawed model



We really need a LOT of data – generally more than the 'biggest' of 'big data'

For just 3 Binary variables there are $2^3 = 8$ combinations of state values

Obviously not all are as likely as others but it will be difficult to learn without, say, an average of at least 10 per state combination

That's 80 records minimum (typical medical studies struggle to get 50 participants)

If we have, say 150 variables then clearly we have to have a lot of data

But organizations now routinely collect 'big data' - many terabytes per day

So 150 variables is surely a breeze …..

If the average number of states for each variable is, say 5 (most will be numeric variables which will require more) then there are 5^{150} combinations of state values.

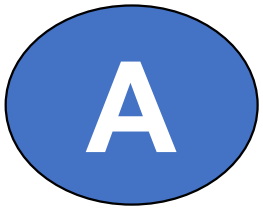
That's a BIGGER number than the estimated number of atoms in the observable universe (10^{80})

In general, no matter how 'big' the data you manage to collect there will NEVER be enough for accurate learning without a lot of expert knowledge.

But presumably, if we have knowledge about the full structure of a model then for 'small' models at least it should be easy to learn from 'big data'? Actually no …….

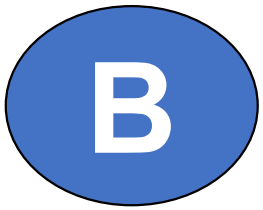
A machine learning fable

Medical condition A



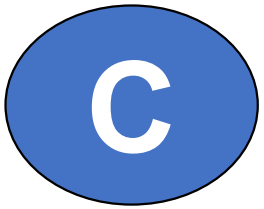
1 in 1000 people

Medical condition B



1 in 1000 people

Medical condition C
(also known as “FiroziliRalitNoNeOba”



?

It is widely believed that people with A or B usually also have C

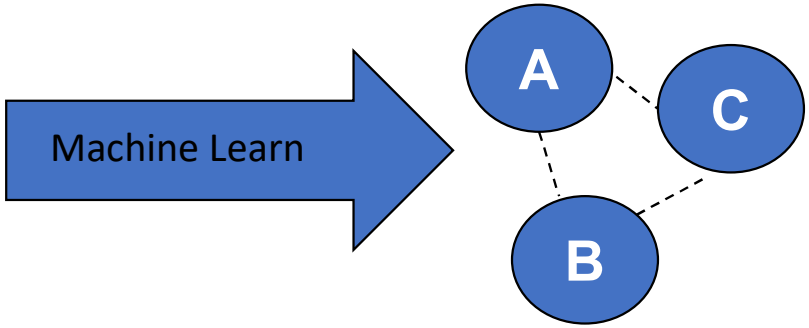
Bill's massive patient dataset

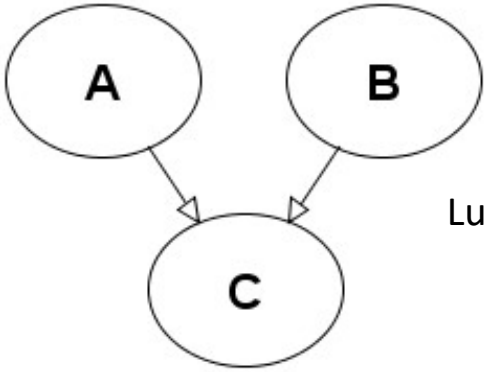
Patient number	A	B	C
1	No	No	No
2	No	No	No
3	Yes	No	Yes
4	No	No	No
5	No	No	No
6	No	No	No
7	Yes	No	Yes
8	No	Yes	Yes
9	No	No	No
10	No	No	No
11	No	No	No
12	No	Yes	Yes
13	No	No	No
14	No	No	No
	….	…	…
	….	…	…
600,000	No	No	No

Bill’s machine learning mate Fred

Fred

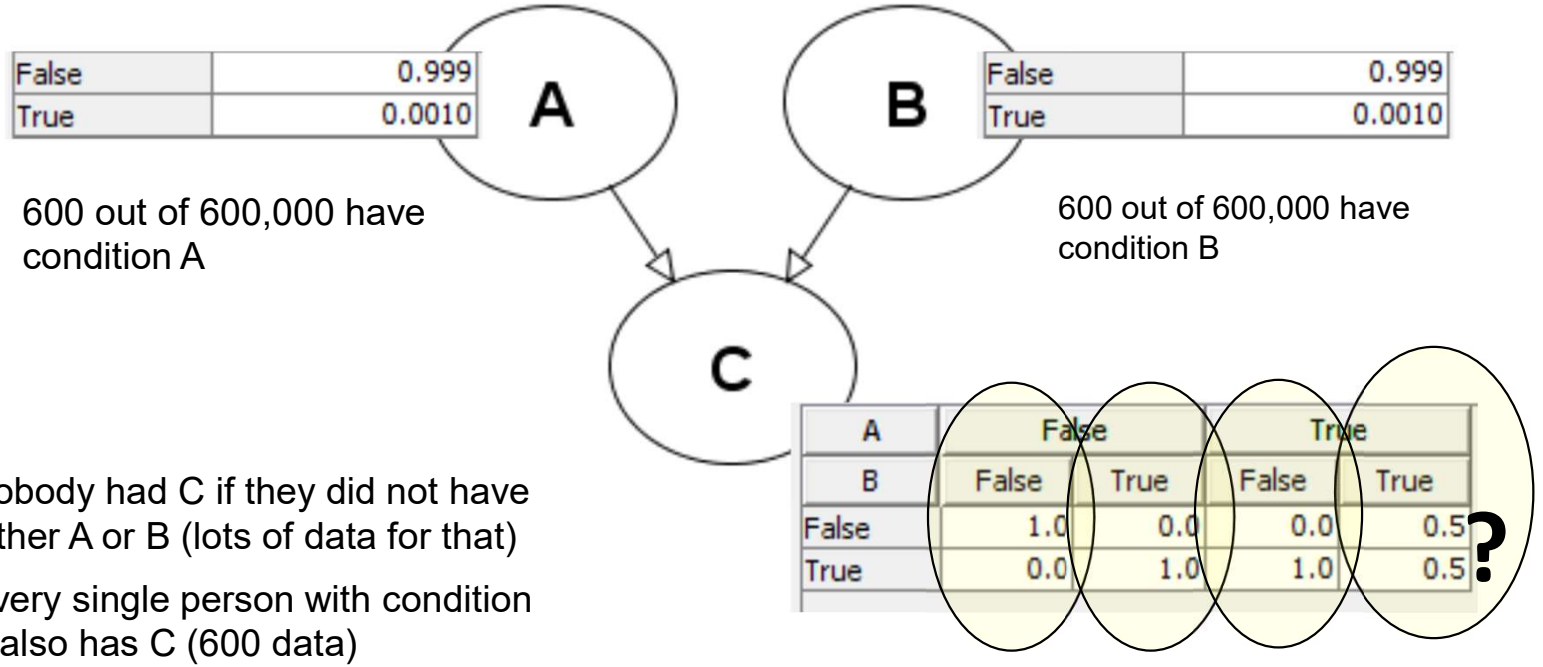
Patient number	A	B	C
1	No	No	No
2	No	No	No
3	Yes	No	Yes
4	No	No	No
5	No	No	No
6	No	No	No
7	Yes	No	Yes
8	No	Yes	Yes
9	No	No	No
10	No	No	No
11	No	No	No
12	No	Yes	Yes
13	No	No	No
14	No	No	No
	….	…	…
	….	…	…
600,000	No	No	No





Ludmilla’s model

Fred's learnt model



600 out of 600,000 have condition A

600 out of 600,000 have condition B

Nobody had C if they did not have either A or B (lots of data for that)

Every single person with condition A also has C (600 data)

and every single person with B also has C (600 data)

But not a single person in the database had both A and B (unsurprising given the rarity of each)

Hence we have no information about these probabilities. So algorithms generally assign equal probabilities here.

Or maybe it makes sense to assign True =1 since C is true when either A or B is true

Ludmilla's knowledge

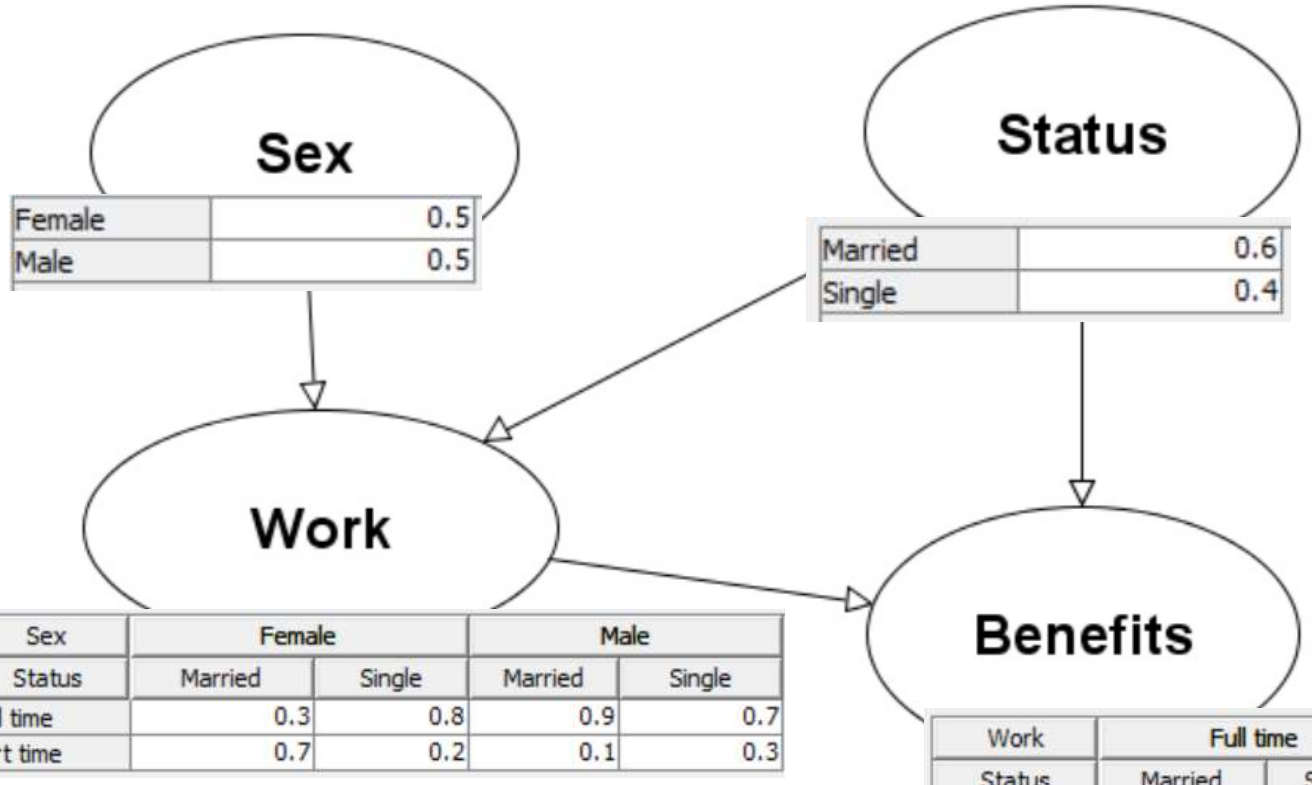
The name of Condition C - FiroziliRalitNoNeOba - is a Russian word

Its literal translation is:

> 'A person suffering from either Firoz or Ralit but not both'.
>
> 'Firoz' is the Russian word for condition A and 'Ralit' is the Russian word for condition B.

So Ludmilla knows the correct probability value for C when both A and B are "True" is 0 (***not*** 0.5 as assumed from lack of data and ***not*** 1 as seemed intuitive).

Learning probability tables with data and knowledge



Female	0.5
Male	0.5

Married	0.6
Single	0.4

Sex	Female		Male	
Status	Married	Single	Married	Single
Full time	0.3	0.8	0.9	0.7
Part time	0.7	0.2	0.1	0.3

Work	Full time		Part time	
Status	Married	Single	Married	Single
False	0.8	0.9	0.4	0.8
True	0.2	0.1	0.6	0.2

Model with expert priors

Data

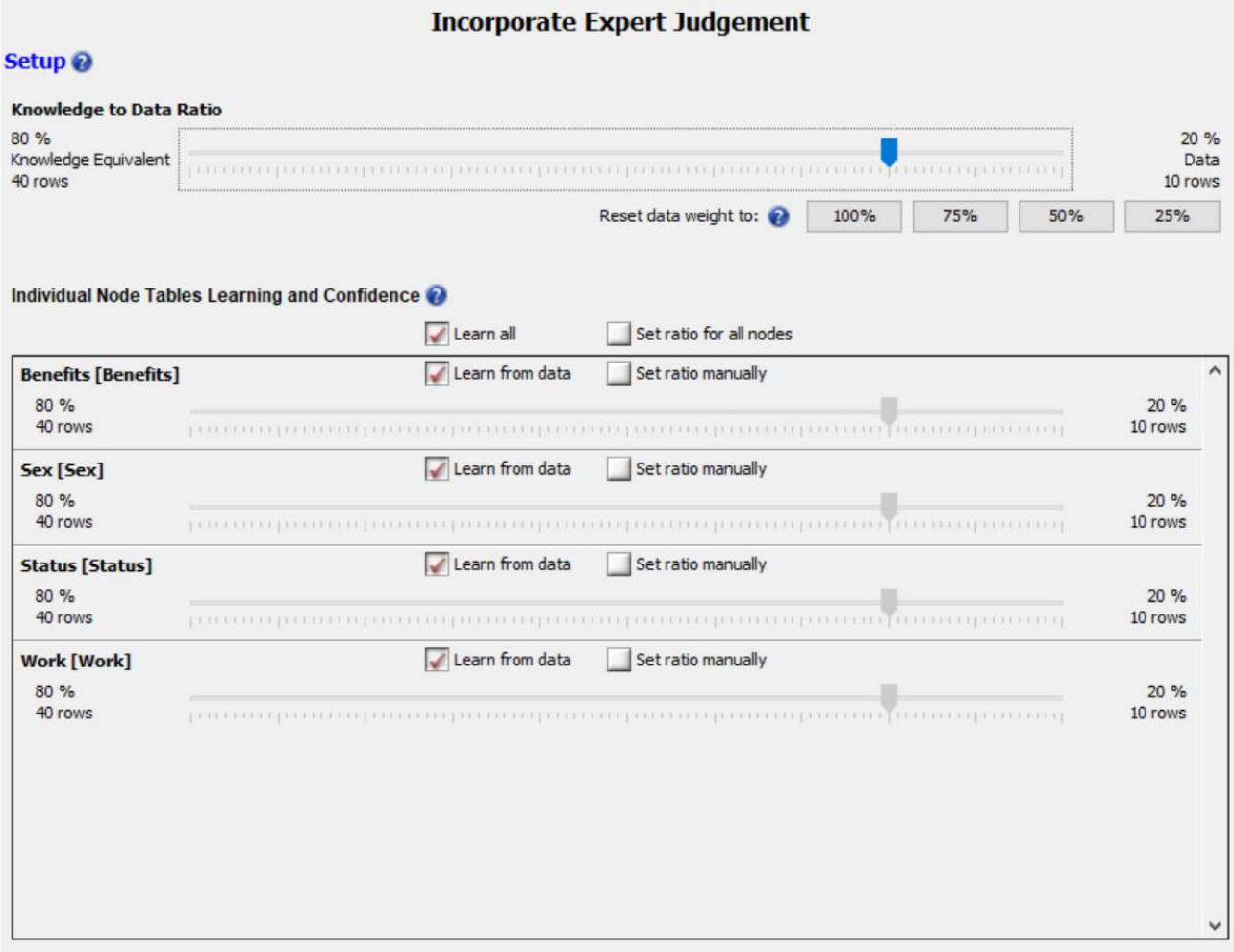
Sex	Status	Work	Benefits
Female	Single	Full time	FALSE
Female	Married	Part time	TRUE
Male	Single	Full time	FALSE
Female	Married	Part time	FALSE
Female	Married	Full time	FALSE
Male	Married	Part time	TRUE
Female	Married	Part time	FALSE
Male	Married	Full time	TRUE
Female	Married	Part time	TRUE
Male	Single	Full time	TRUE

Learning probability tables with data and knowledge

In AgenaRisk “Learning” select “incorporate expert judgment” to bring up this dialog

For each node you can select how much weighting to give to the prior knowledge compared to the data.

In the example we use a default 80% weighting for all nodes to knowledge. This essentially assumes that for every 10 rows of data the knowledge is equivalent 40 rows



Learning probability tables with data and knowledge

Prior expert knowledge

Work	Full time		Part time	
Status	Married	Single	Married	Single
False	0.8	0.9	0.4	0.8
True	0.2	0.1	0.6	0.2

Learnt from data set in Table 1

Work	Full time		Part time	
Status	Married	Single	Married	Single
False	0.5	0.667	0.4	0.5
True	0.5	0.333	0.6	0.5

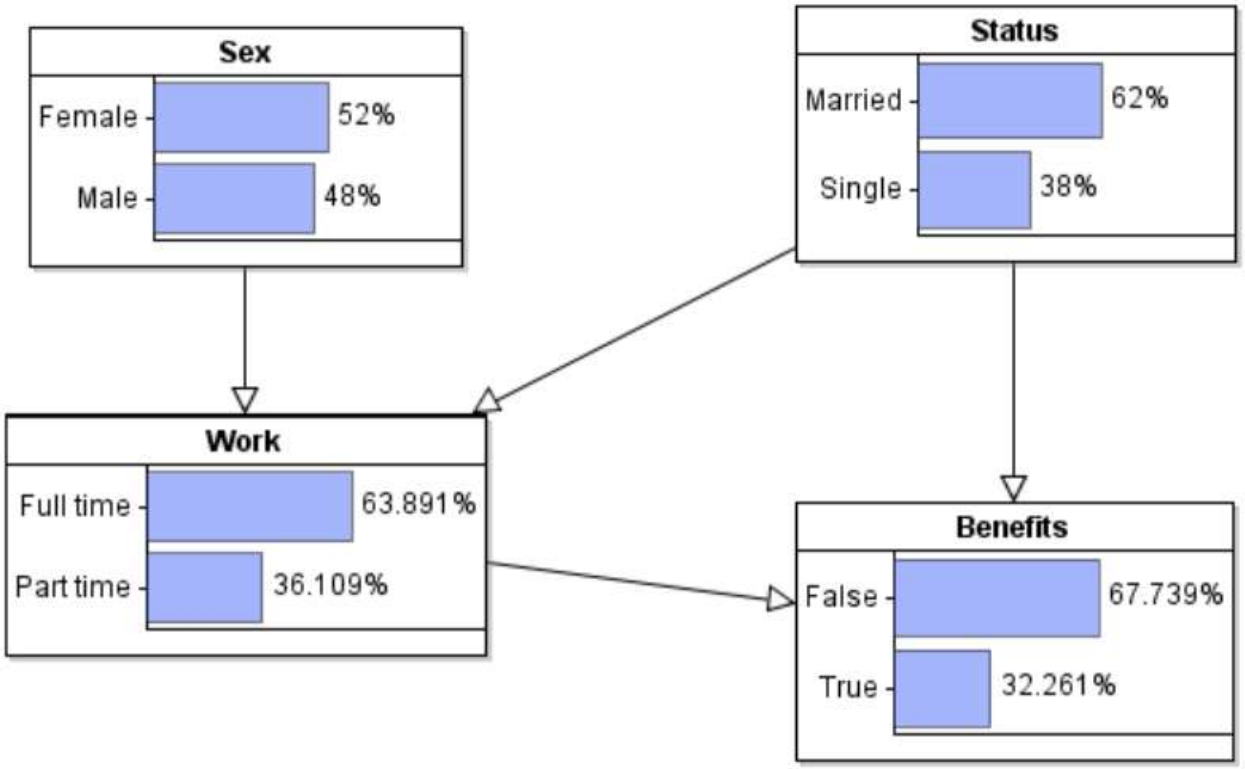
***r* = 0.8**

***1* - *r* = 0.2**

Work	Full time		Part time	
Status	Married	Single	Married	Single
False	0.74	0.853	0.4	0.74
True	0.26	0.147	0.6	0.26

Combined knowledge and data

Learning probability tables with data and knowledge



Model with expert priors (rated 80%) + Data

Handling missing data values

The easily implemented methods assume data is missing at random

But whether data are collected from individuals by choice (e.g. online surveys, volunteers in studies, experiments) or automatically, there are almost inevitable systemic reasons accounting for missing data

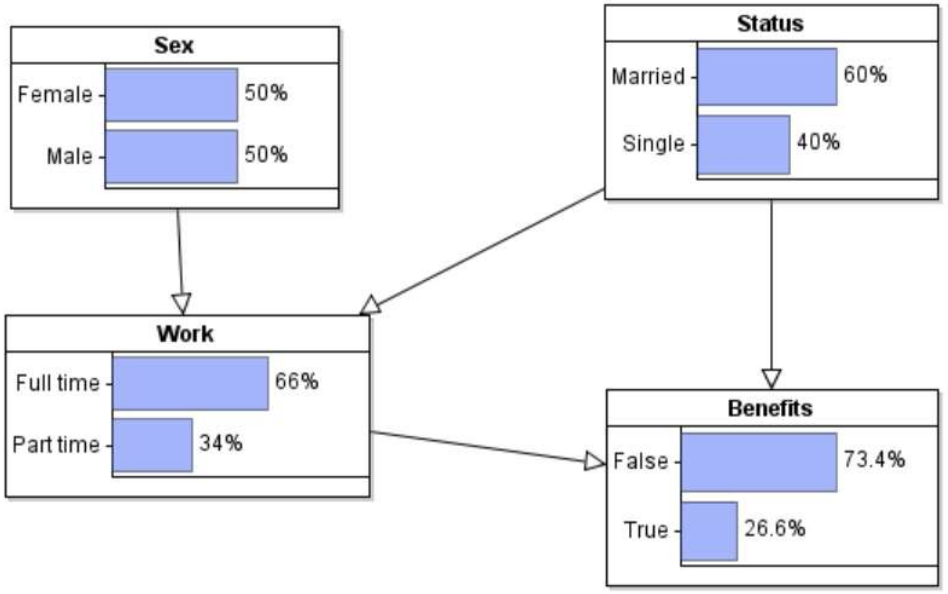
Examples:

In any dataset where individuals are asked to give their salary, missing values are much more likely to be from those with very low or very high salaries

Males are systematically less likely to answer certain types of questions than females and vice versa

Any system automatically mining data about people from what is available online will have far more missing attribute values for older people

Example with Missing Values

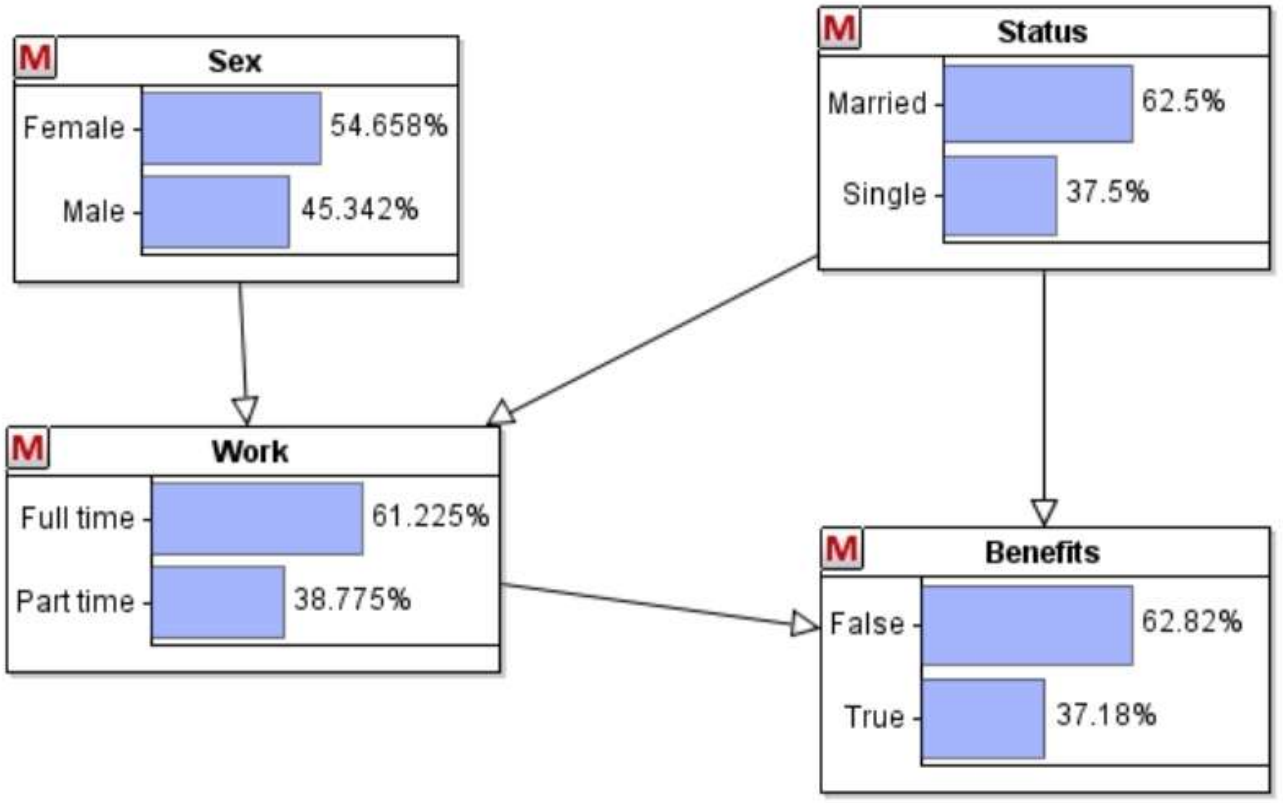


Model with expert priors

Sex	Status	Work	Benefits
Female	Single	Full time	FALSE
Female	Married	Part time	TRUE
Male	Single	Full time	FALSE
Female	Married		FALSE
Female	Married	Full time	
Male	Married	Part time	TRUE
Female	Married	Part time	FALSE
	Married	Full time	TRUE
Female	Married	Part time	TRUE
Male	Single	Full time	TRUE

Data

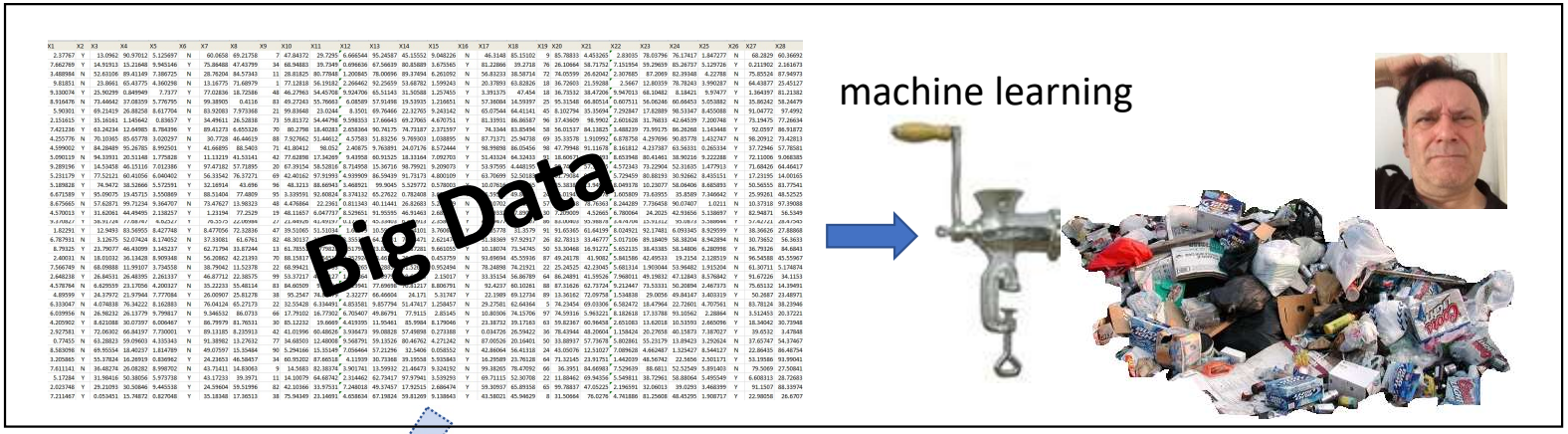
Example with Missing Values

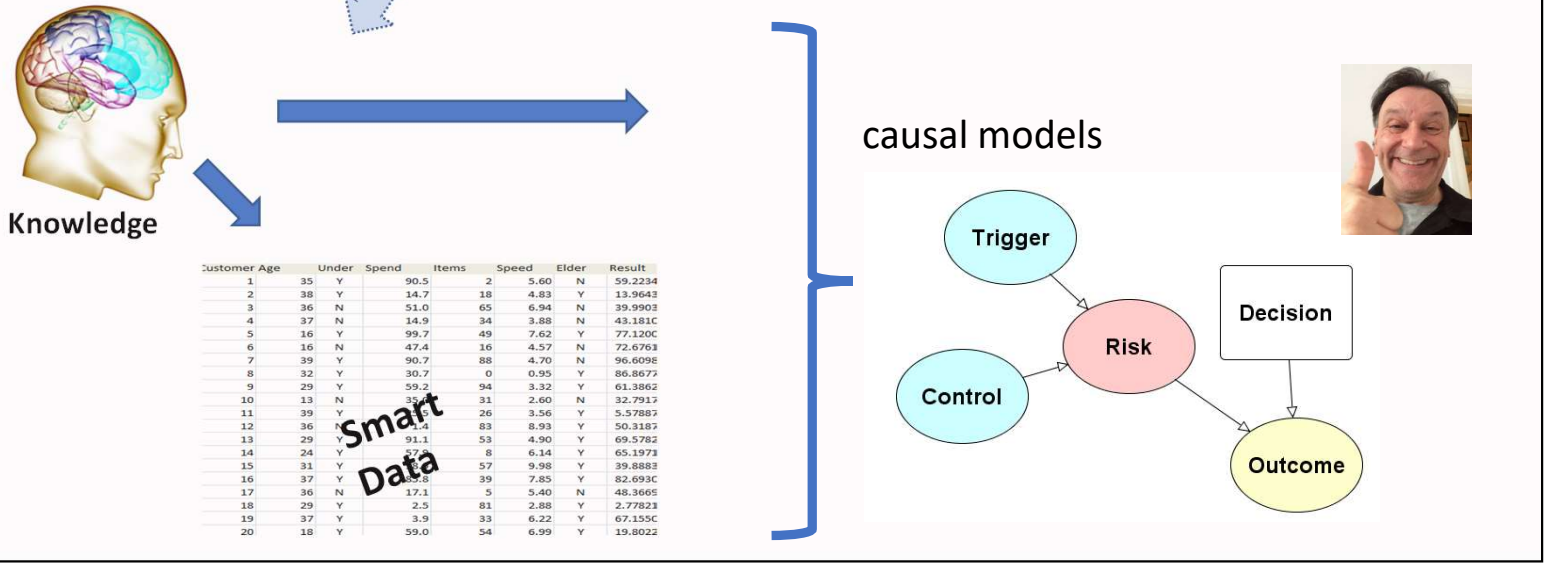


AgenaRisk uses EM learning algorithm for missing values

Model with expert priors (rated 75%) + Data + Missing data estimates

Big Data … or Smart Data?





Summary

- The various fancy machine learning algorithms all achieve similar accuracy
- With sufficient data they can get to reasonably accurate predictions of the outcome (~80% AUC)
- But they cannot handle ‘what if’ analysis or ‘back propagation’
- Causal models can
- Machine learning algorithms cannot ‘learn’ causality no matter how ‘big’ the data because:
  - we can have causation without correlation
  - we can have ‘unfaithfulness’ whereby an attribute appears to have no impact on another even though it does
- Good causal models require (expert) knowledge plus data – ‘smart data’ approach
- We can automatically incorporate data with knowledge
- We can also handle missing values – but not always well