Scalable Inference and Learning for High-Level Probabilistic Models

Guy Van den Broeck

KU Leuven

Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Approximate symmetries
  - Lifted learning

Outline

- Motivation
  - **Why high-level representations?**
  - Why high-level reasoning?
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Approximate symmetries
  - Lifted learning

Graphical Model Learning

Medical Records

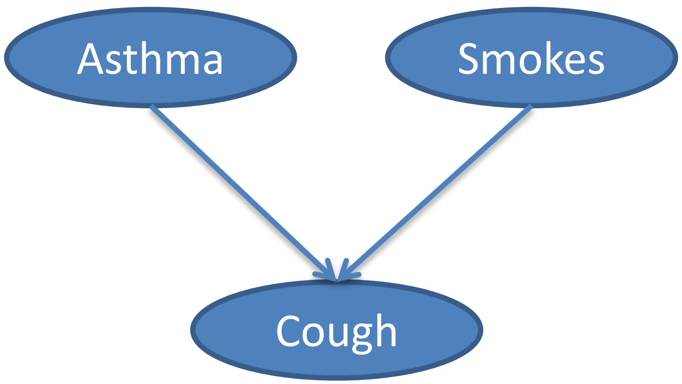
Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Graphical Model Learning

Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

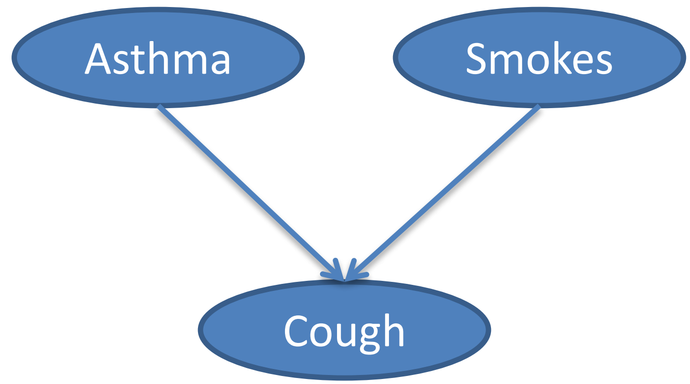


Graphical Model Learning

Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0



Big data

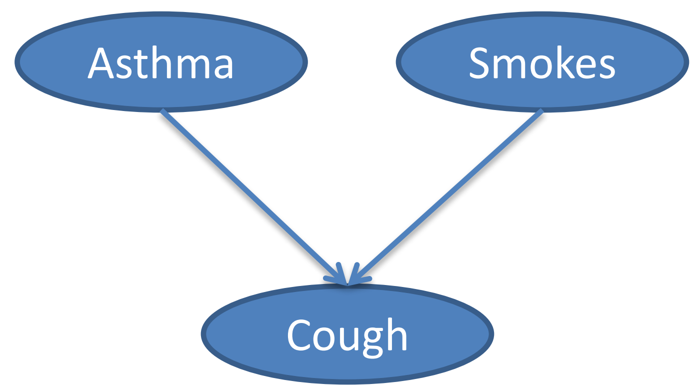
Graphical Model Learning

Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Frank	1	?	?



Graphical Model Learning

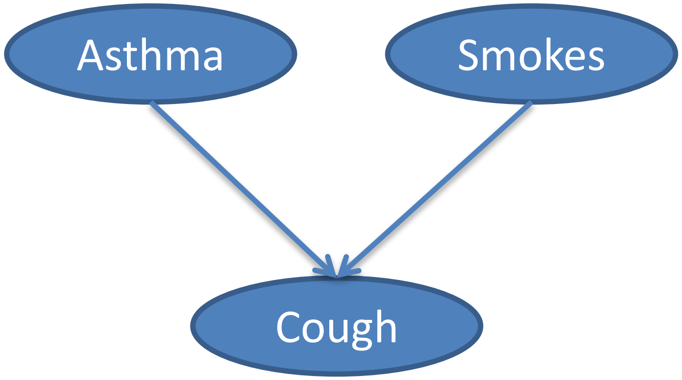
Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Frank	1	?	?

Frank	1	0.3	0.2



Graphical Model Learning

Medical Records

Bayesian Network

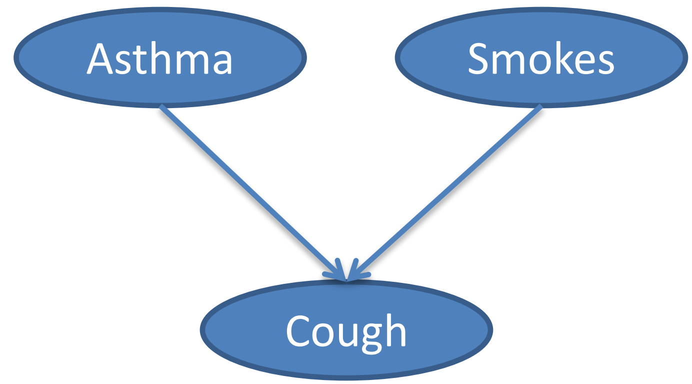
Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Frank	1	?	?

Friends

Brothers

Frank	1	0.3	0.2



Graphical Model Learning

Medical Records

Bayesian Network

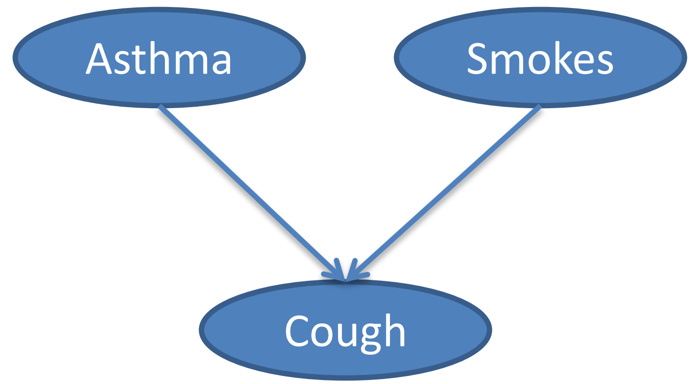
Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Friends

Brothers

Frank	1	?	?

Frank	1	0.3	0.2



Graphical Model Learning

Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

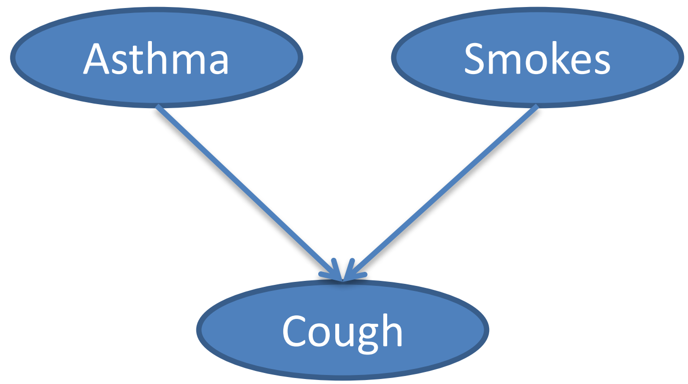
Frank	1	?	?

Friends

Brothers

Frank	1	0.3	0.2

Frank	1	0.2	0.6



Graphical Model Learning

Medical Records

Bayesian Network

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

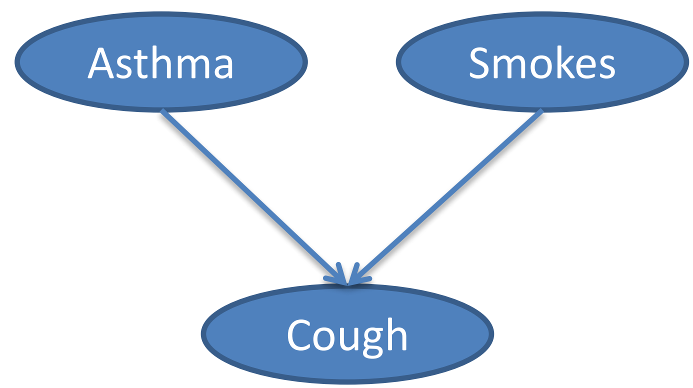
Frank	1	?	?

Friends

Brothers

Frank	1	0.3	0.2

Frank	1	0.2	0.6

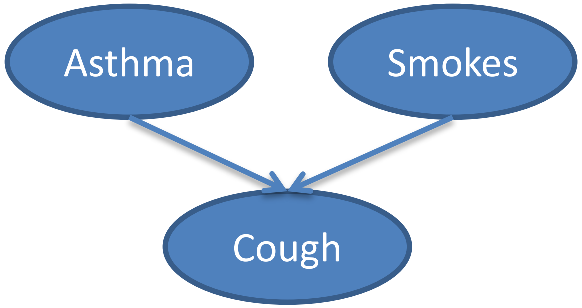


Rows are **independent** during learning and inference!

Statistical Relational Representations

Augment graphical model with relations between entities (rows).

<u>Intuition</u>



+ Friends have similar smoking habits

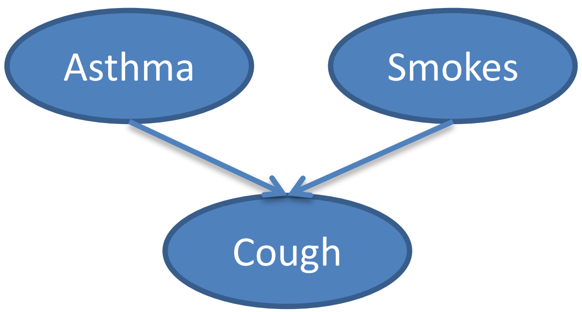
+ Asthma can be hereditary

<u>Markov Logic</u>

Statistical Relational Representations

Augment graphical model with relations between entities (rows).

<u>Intuition</u>



+ Friends have similar smoking habits

+ Asthma can be hereditary

<u>Markov Logic</u>

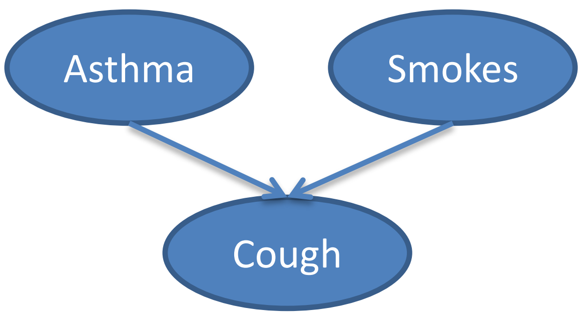
2.1 Asthma ⇒ Cough

3.5 Smokes ⇒ Cough

Statistical Relational Representations

Augment graphical model with relations between entities (rows).

Intuition



+ Friends have similar smoking habits

+ Asthma can be hereditary

Markov Logic

2.1 Asthma(x) ⇒ Cough(x)

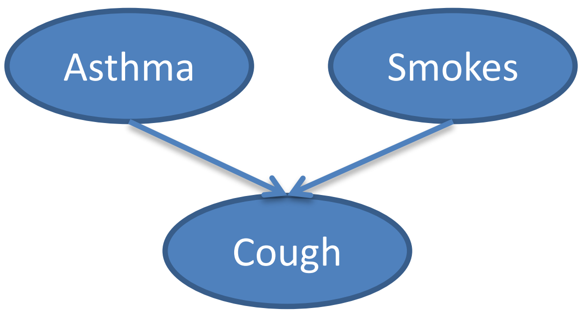
3.5 Smokes(x) ⇒ Cough(x)

Logical variables refer to entities

Statistical Relational Representations

Augment graphical model with relations between entities (rows).

<u>Intuition</u>



+ Friends have similar smoking habits

+ Asthma can be hereditary

<u>Markov Logic</u>

2.1 Asthma(x) ⇒ Cough(x)

3.5 Smokes(x) ⇒ Cough(x)

1.9 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

1.5 Asthma (x) ∧ Family(x,y) ⇒ Asthma (y)

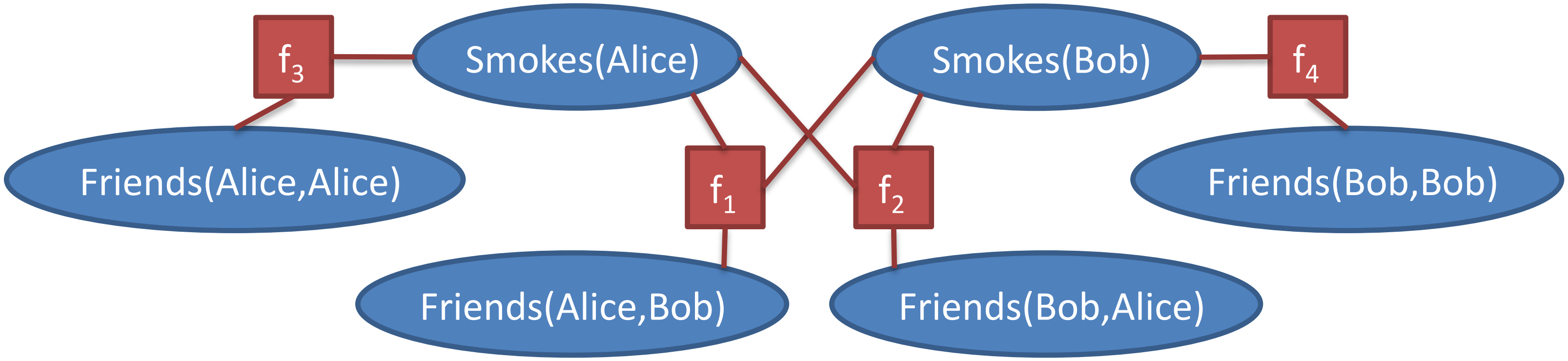
Equivalent Graphical Model

- Statistical relational model (e.g., MLN)

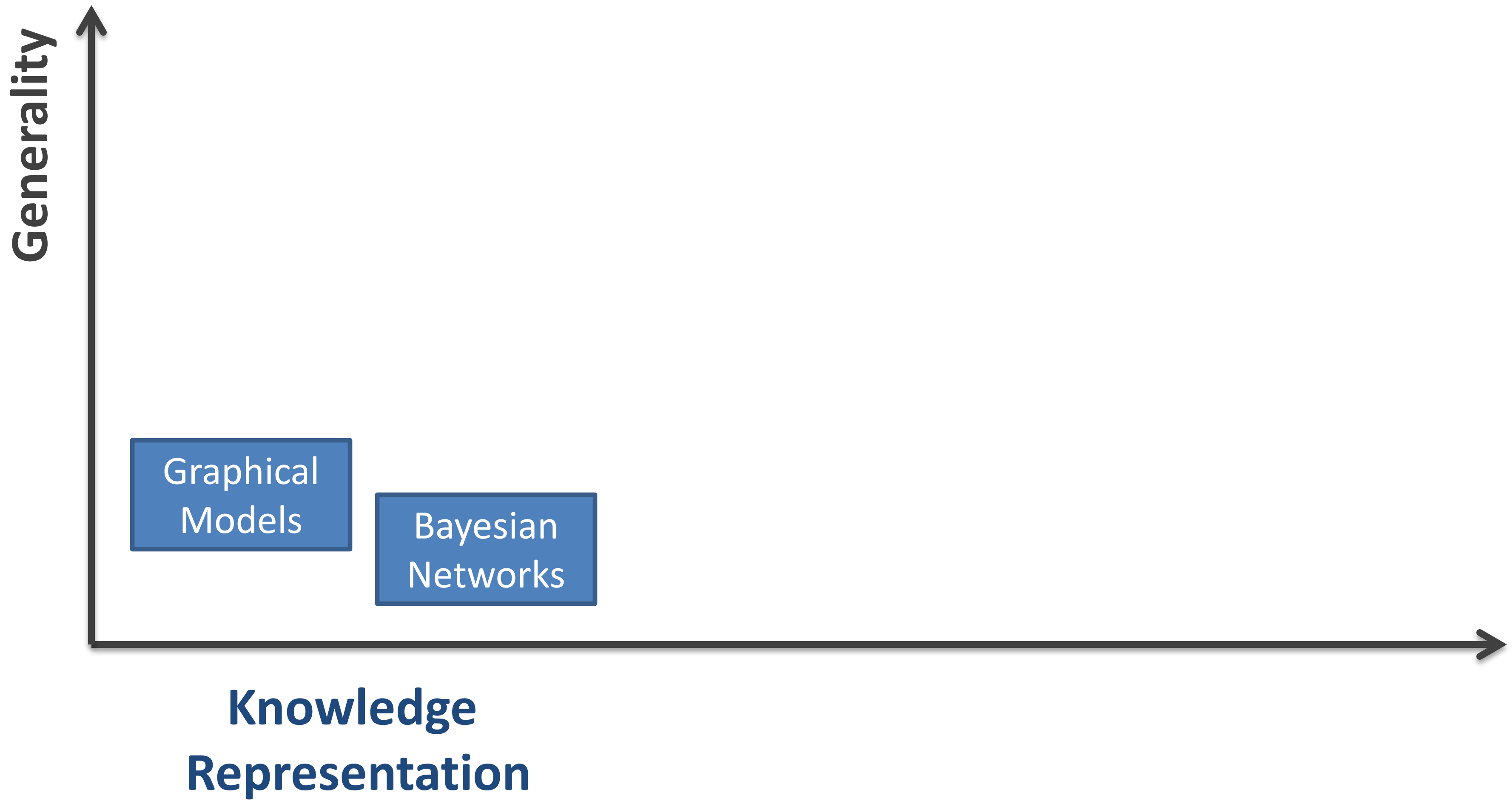
1.9 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

- Ground atom/tuple = **random variable** in {true,false}
  e.g., Smokes(Alice), Friends(Alice,Bob), etc.

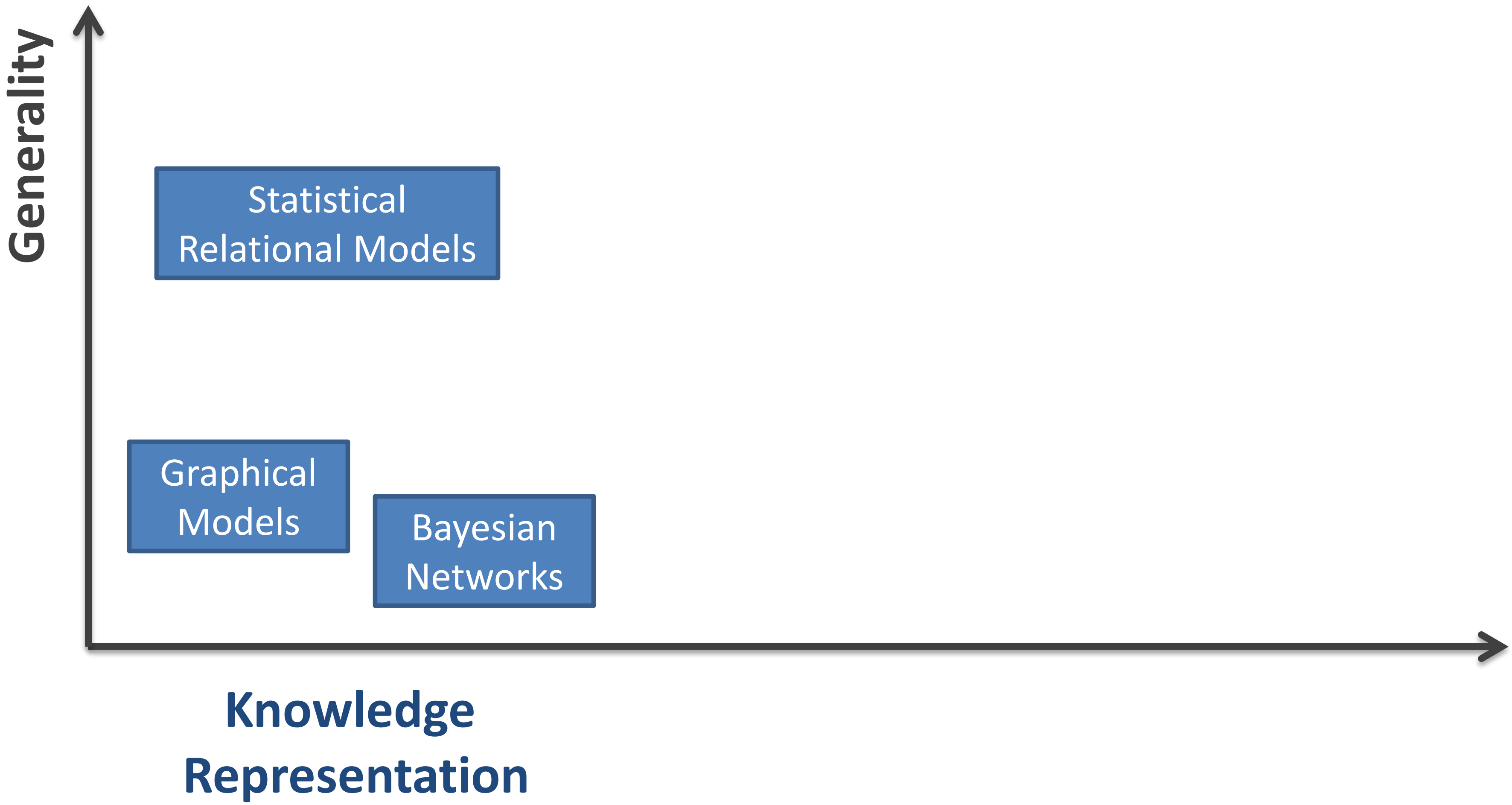
- Ground formula = **factor** in propositional factor graph



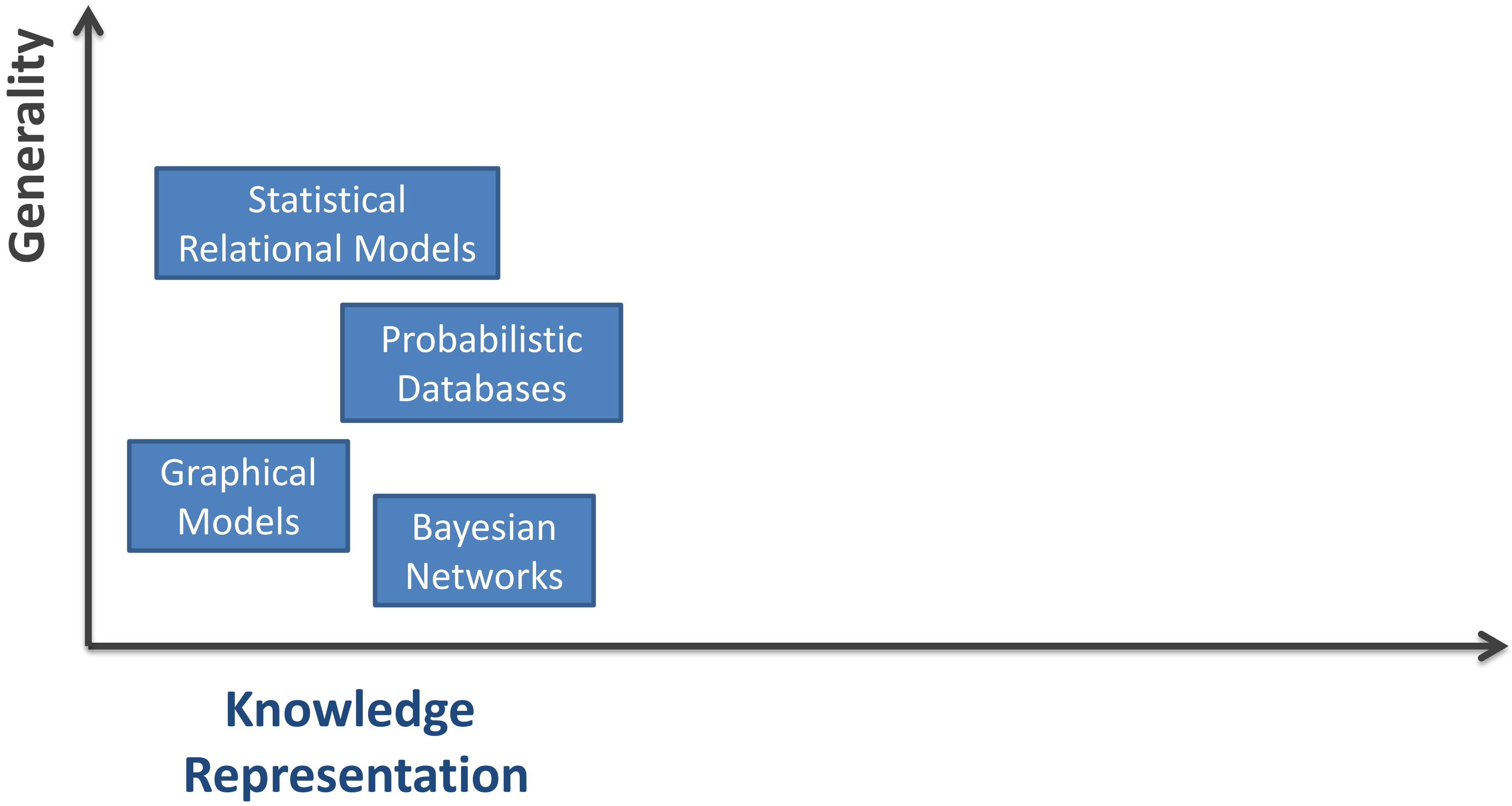
Research Overview



Research Overview



Research Overview



Probabilistic Databases

- Tuple-independent probabilistic databases

Actor

Name	Prob
Brando	0.9
Cruise	0.8
Coppola	0.1

WorkedFor

Actor	Director	Prob
Brando	Coppola	0.9
Coppola	Brando	0.2
Cruise	Coppola	0.1

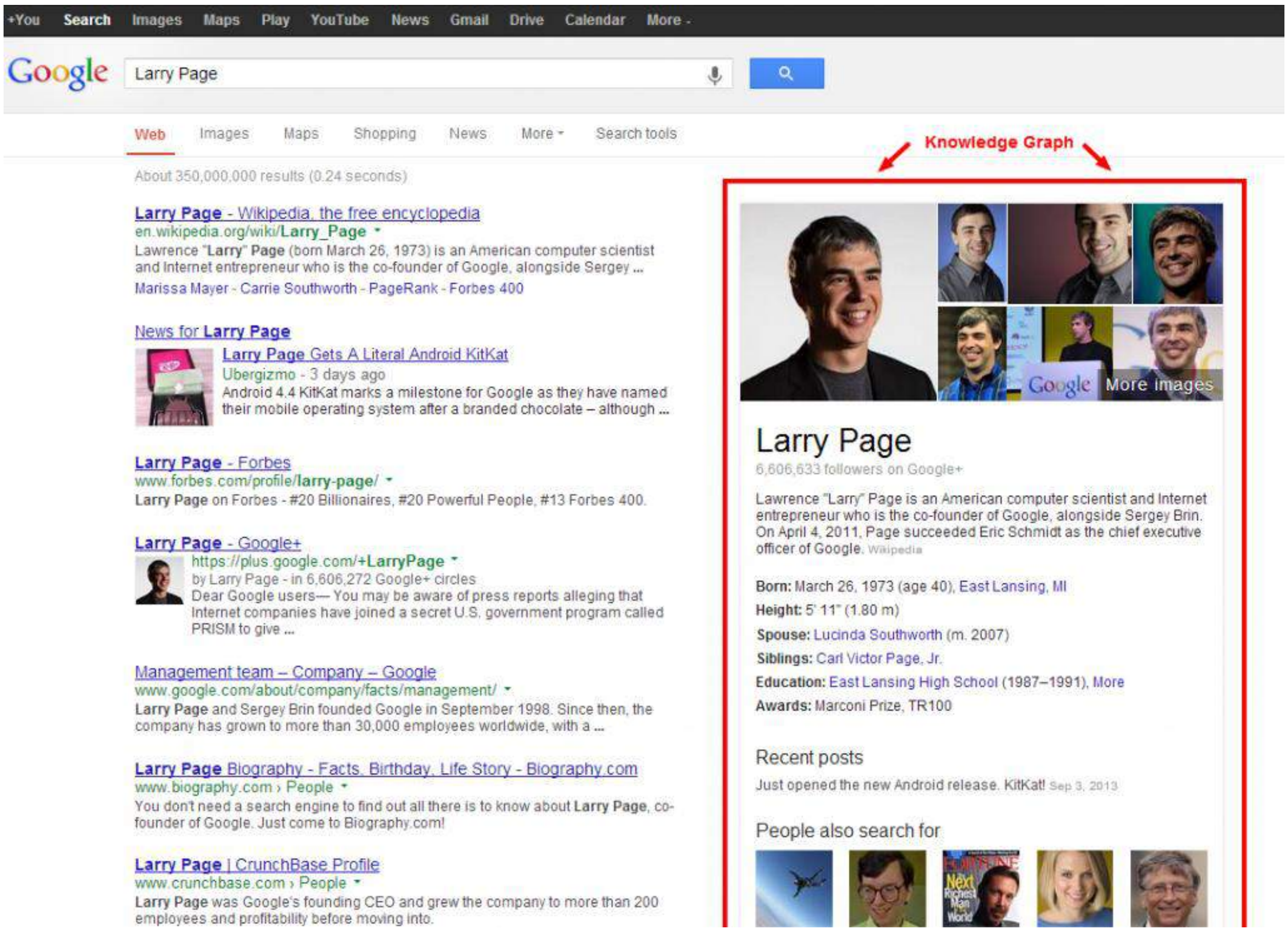
- Query: SQL or First-order logic

```
SELECT Actor.name
FROM Actor, WorkedFor
WHERE Actor.name = WorkedFor.actor
```

$Q(x) = \exists y\ \mathrm{Actor}(x) \wedge \mathrm{WorkedFor}(x,y)$

- Learned from the web, large text corpora, ontologies, etc., using **statistical** machine learning.

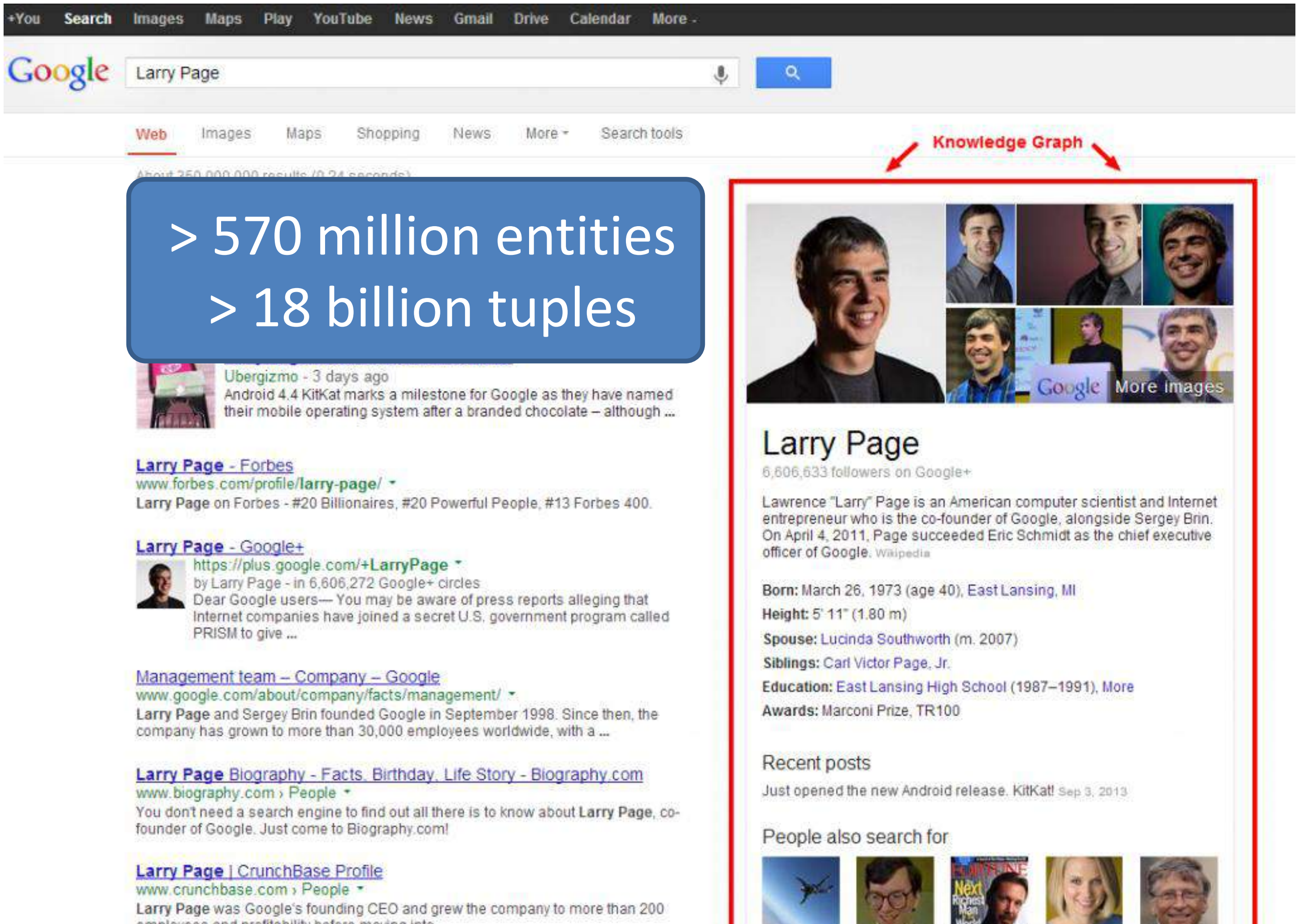
Google Knowledge Graph



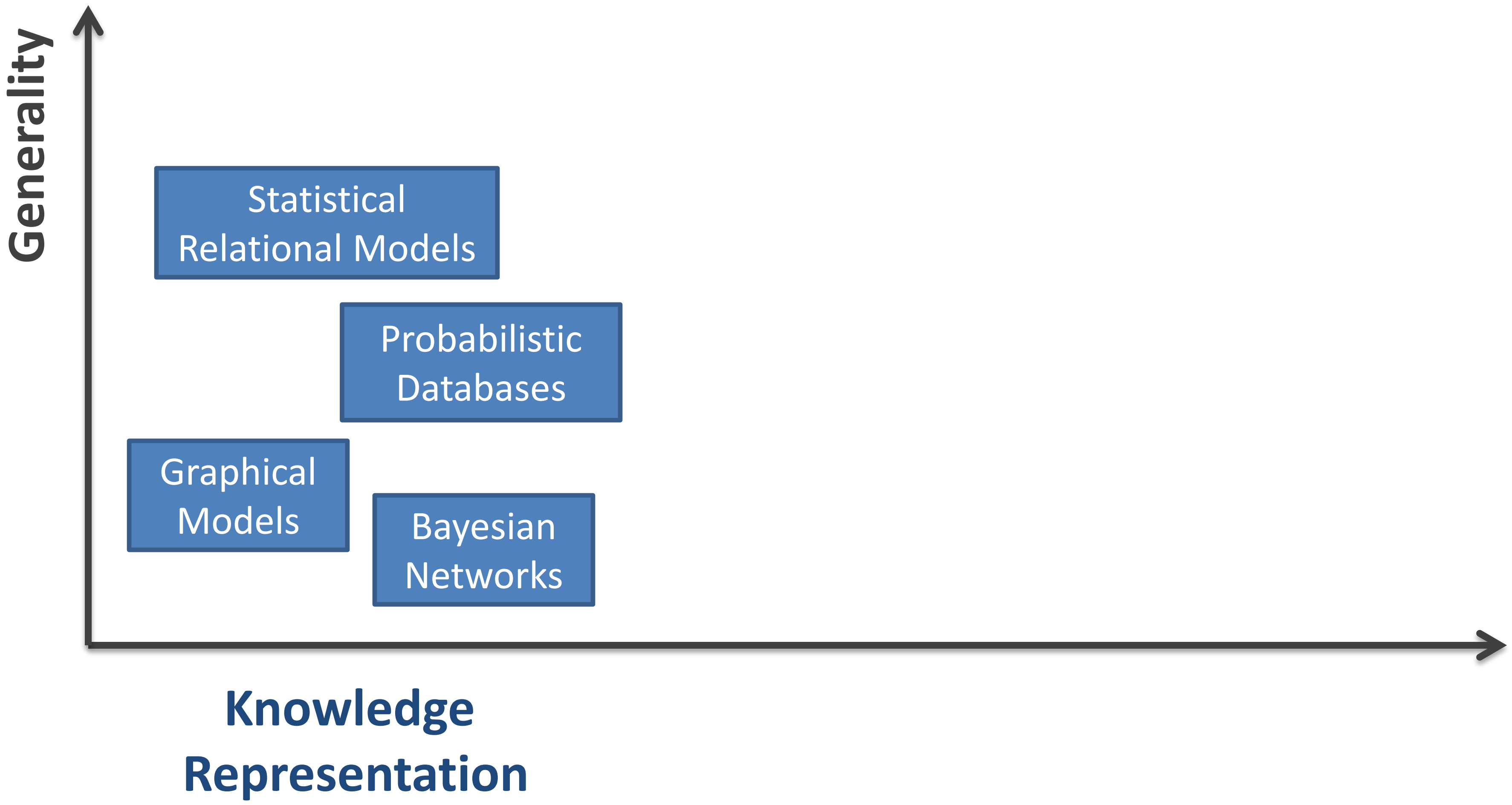
Google Knowledge Graph

> 570 million entities

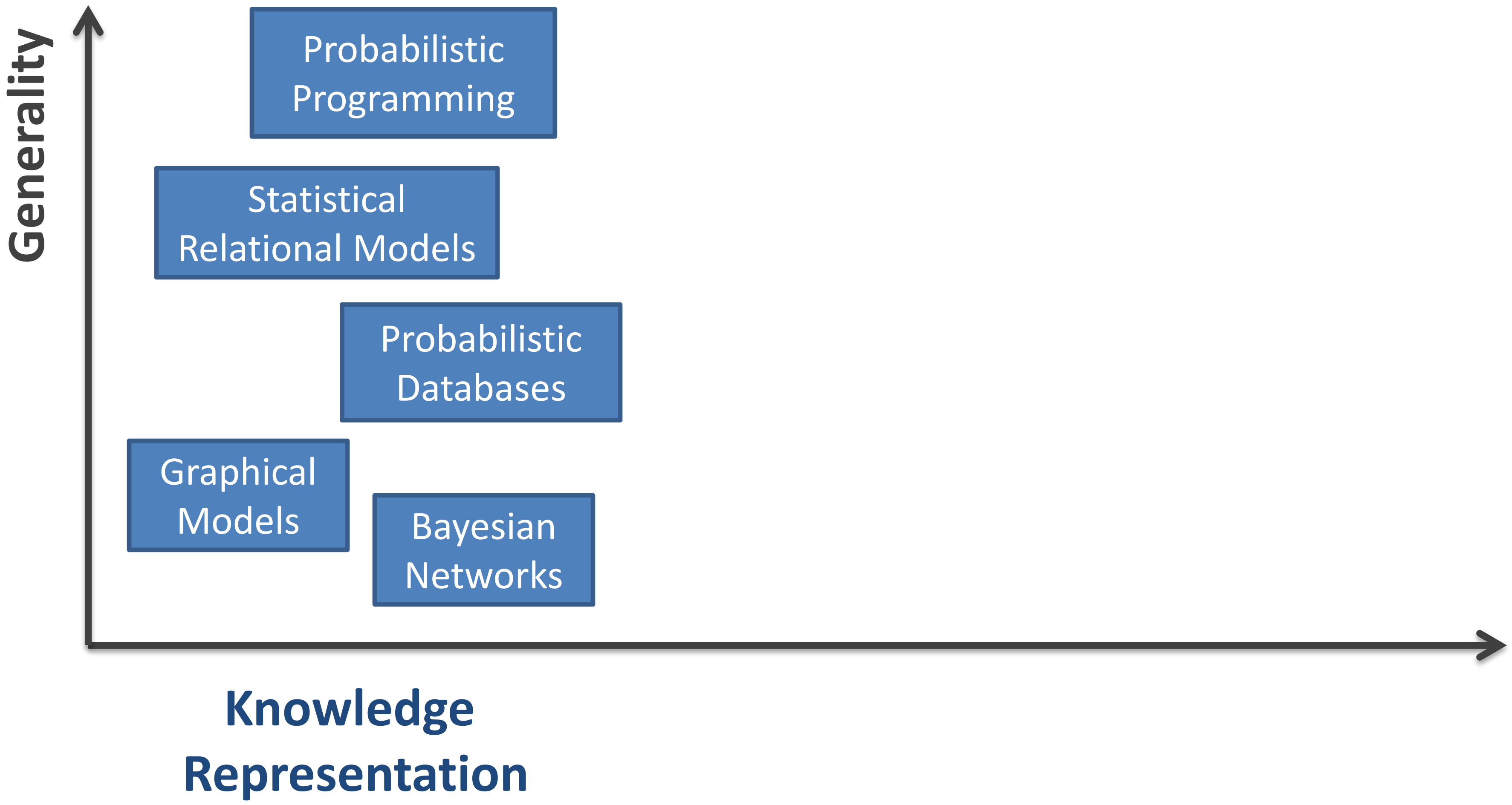
> 18 billion tuples



Research Overview



Research Overview



Probabilistic Programming

- Programming language + random variables
- Reason about distribution over executions

  As going from hardware circuits to programming languages
- *ProbLog*: Probabilistic logic programming/datalog
- Example: Gene/protein interaction networks

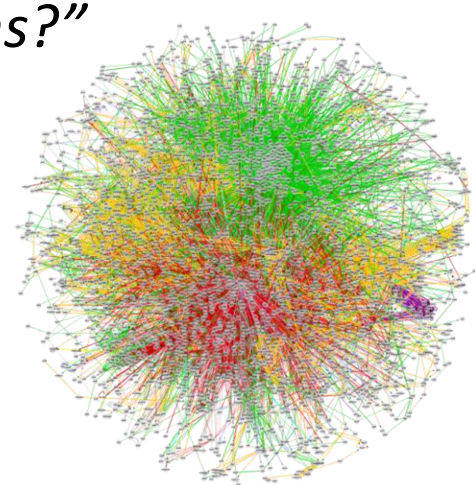
  Edges (interactions) have probability

  "Does there exist a path connecting two proteins?"

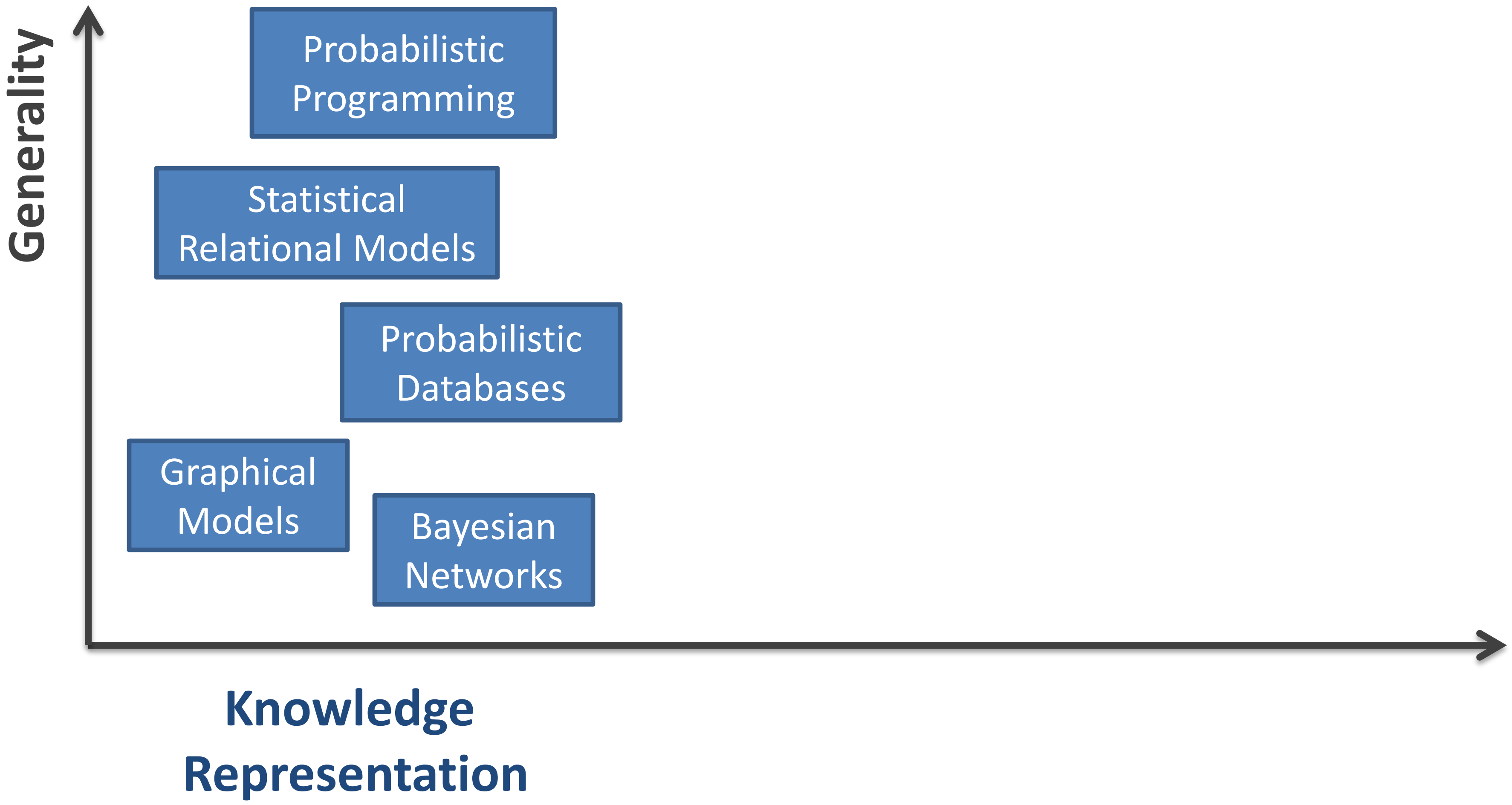
```
path(X,Y) :- edge(X,Y).
path(X,Y) :- edge(X,Z), path(Z,Y).
```

  Cannot be expressed in first-order logic

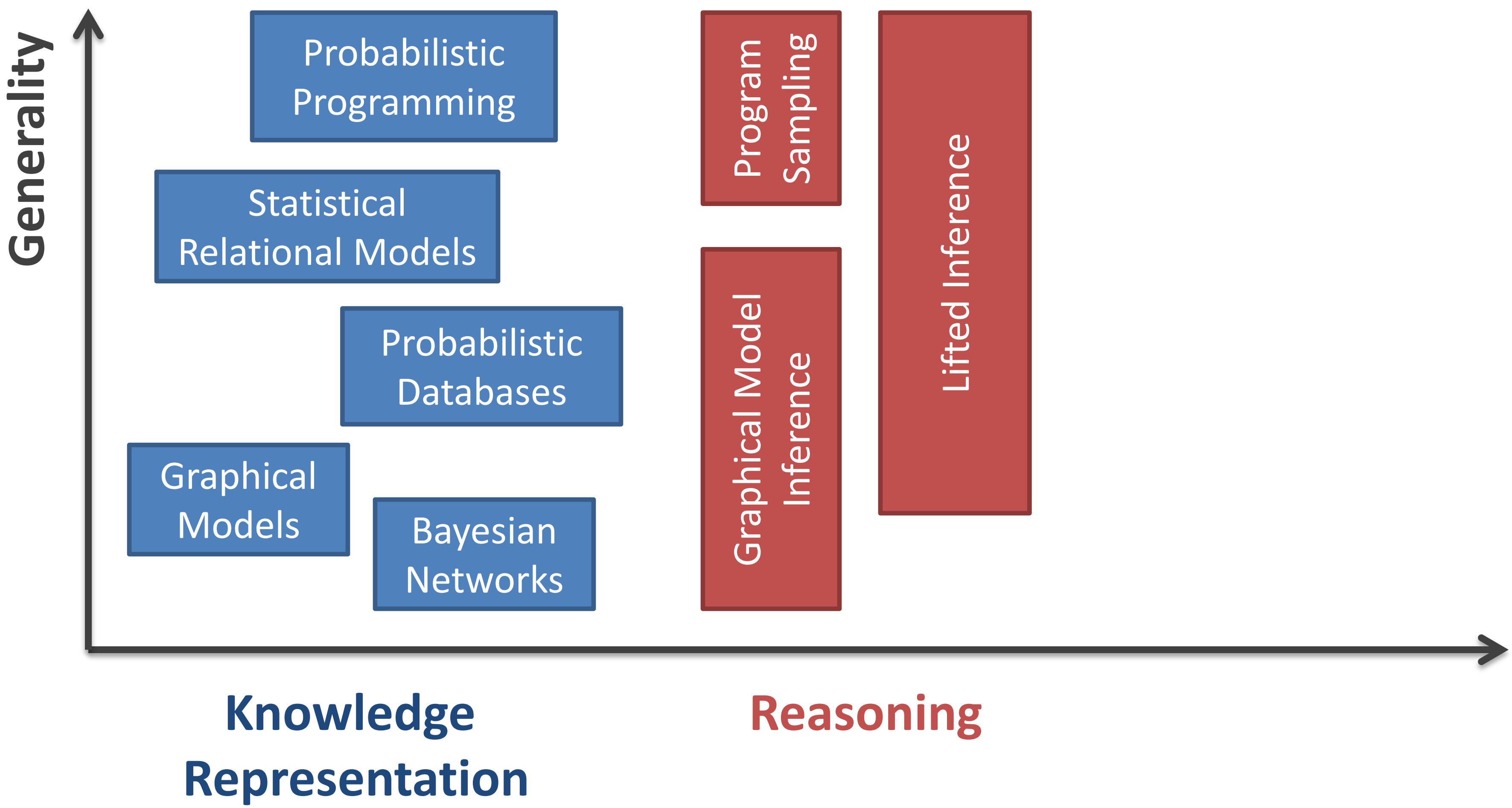
  Need a full-fledged programming language!



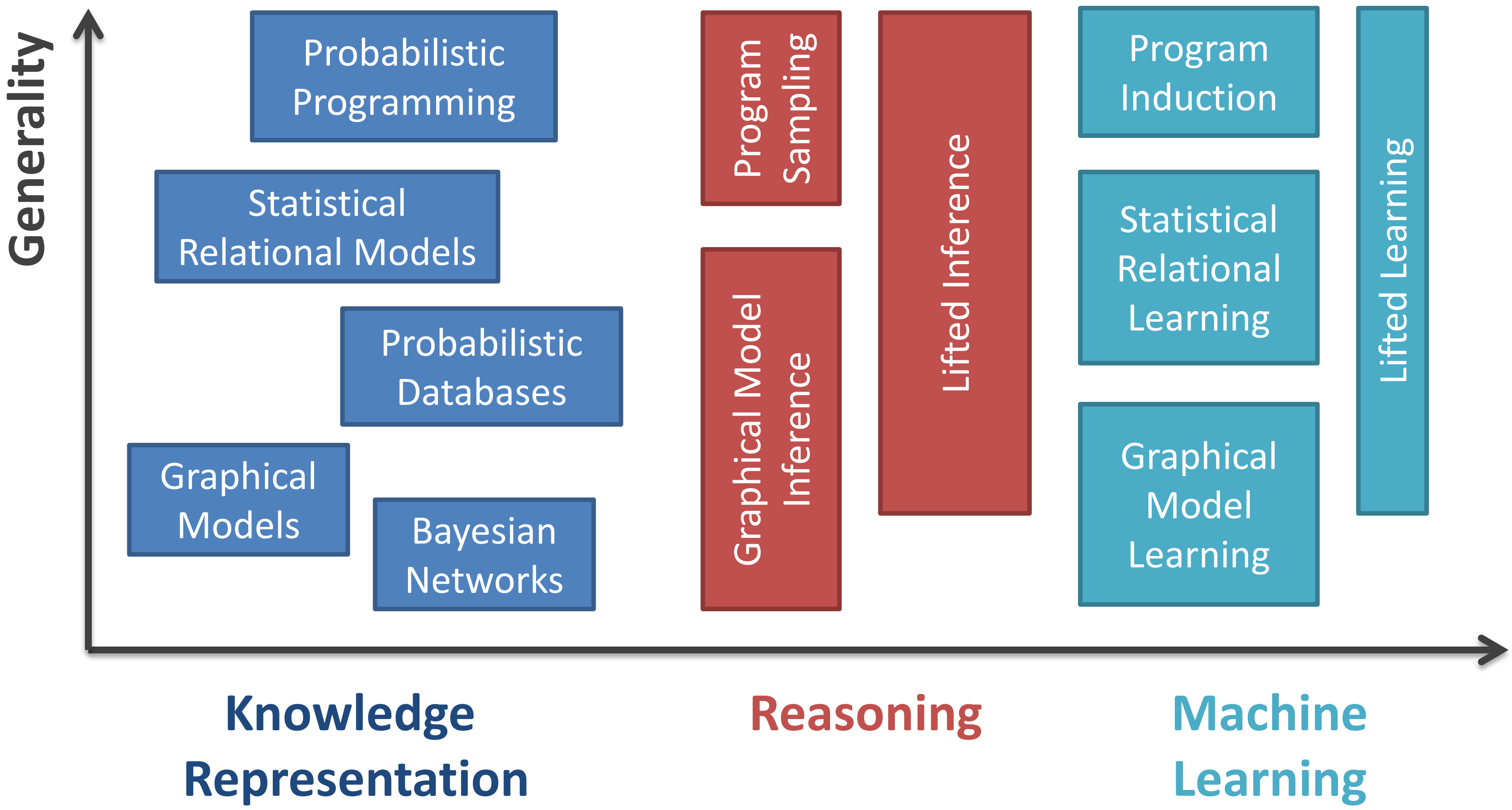
Research Overview



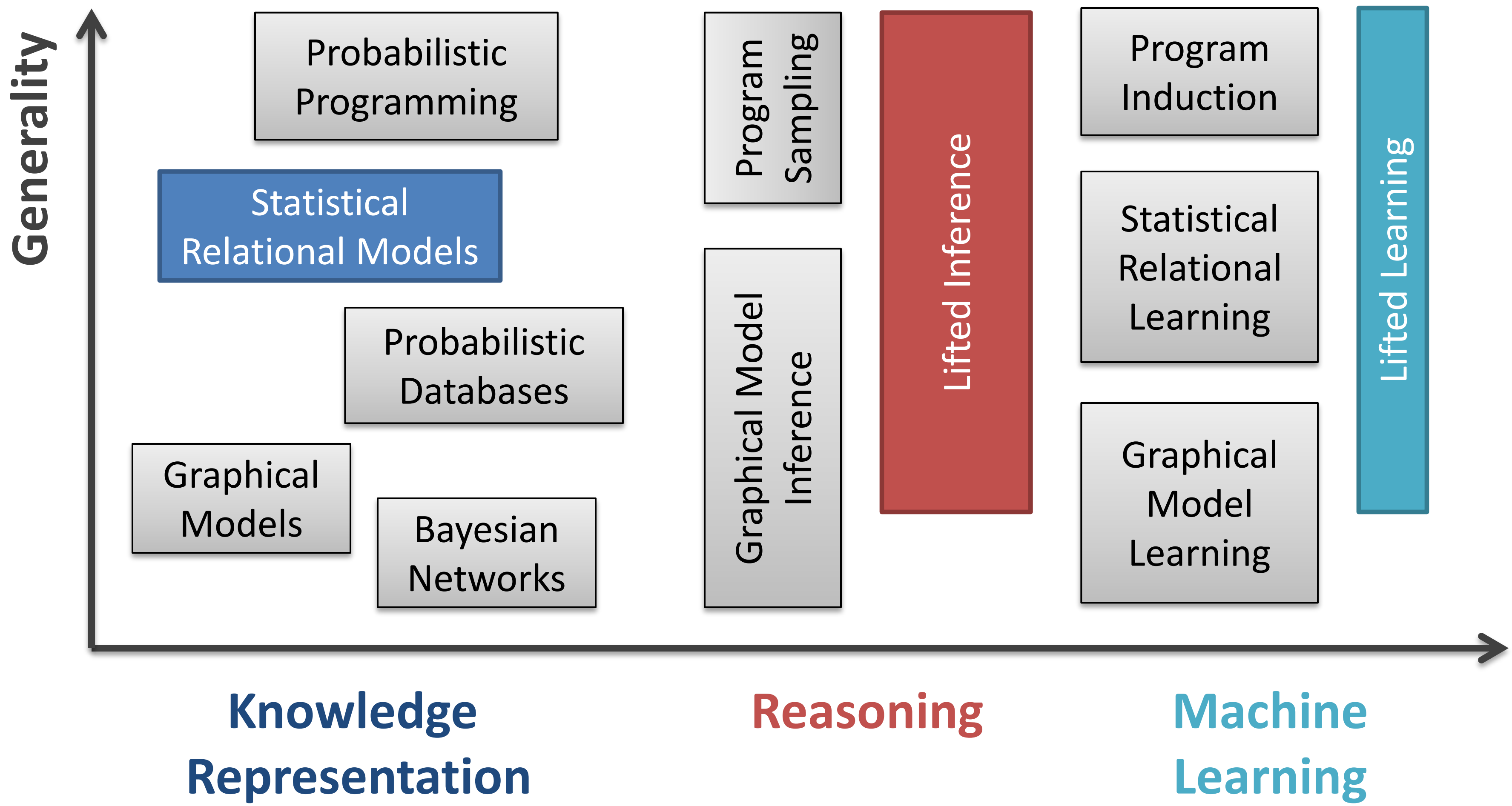
Research Overview



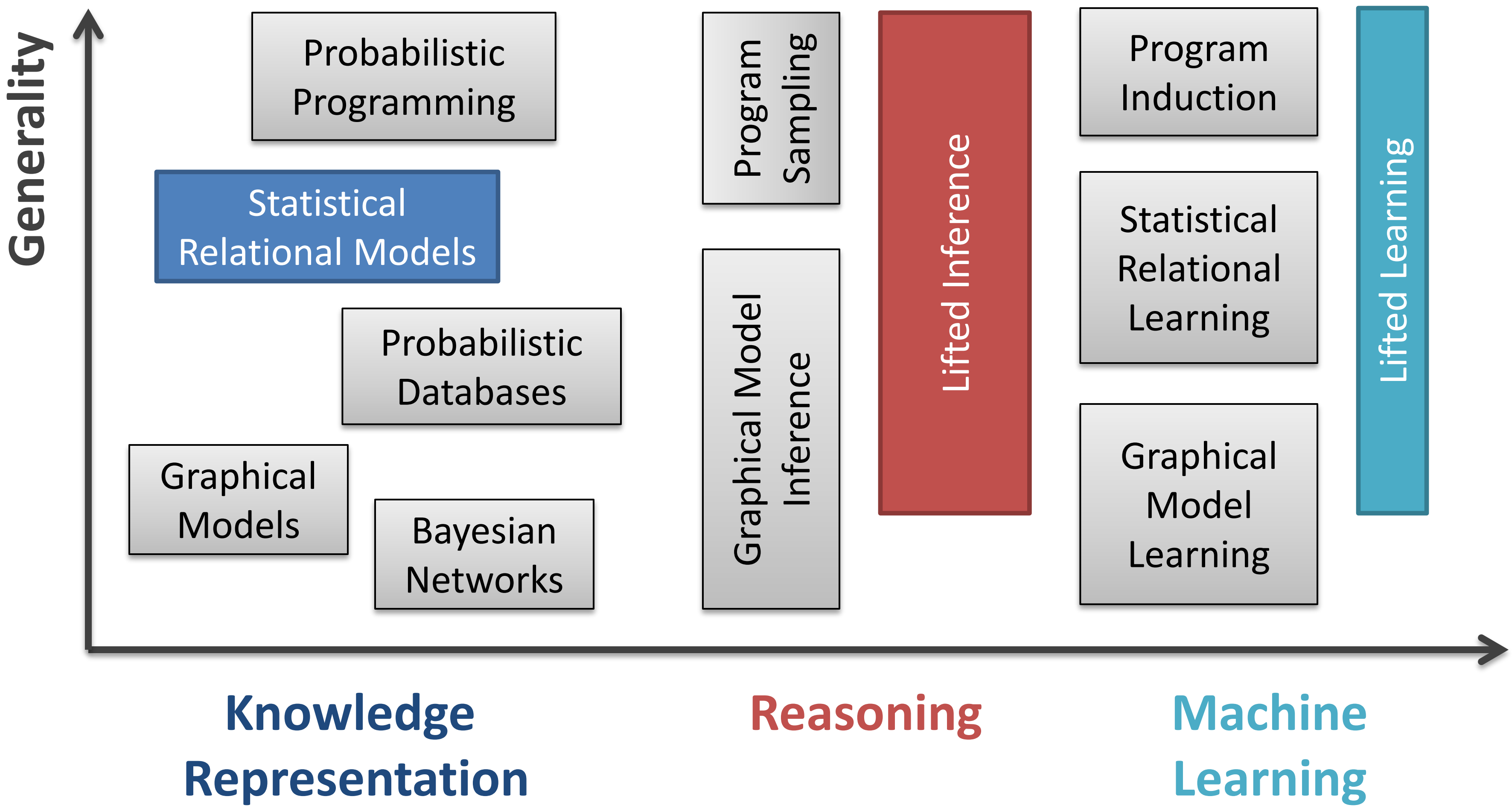
Research Overview



Research Overview



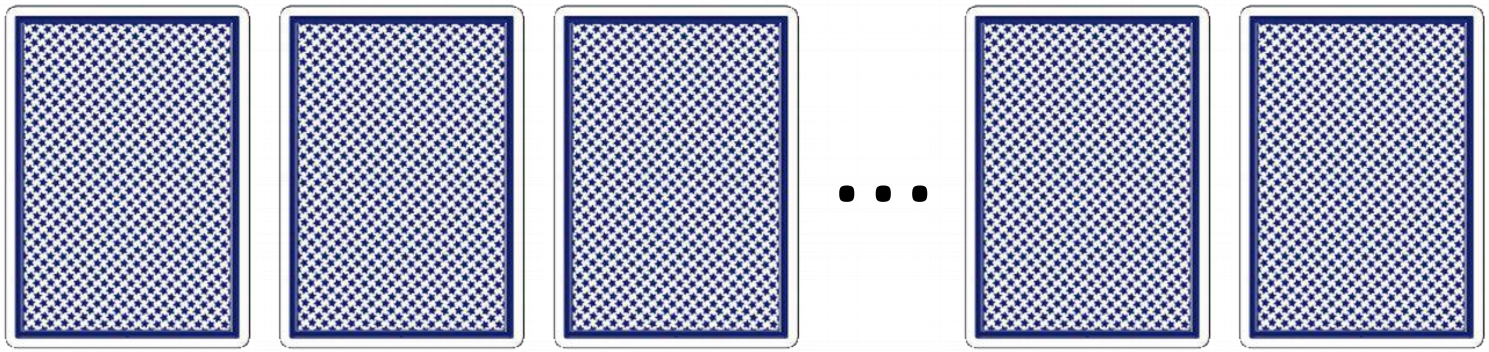
Not about: [VdB, et al.; AAAI'10, AAAI'15, ACML'15, DMLG'11], [Gribkoff, Suciu, Vdb; Data Eng.'14], [Gribkoff, VdB, Suciu; UAI'14, BUDA'14] , [Kisa, VdB, et al.; KR'14], [Kimmig, VdB, De Raedt; AAAI'11], [Fierens, VdB, et al., PP'12, UAI'11, TPLP'15] , [Renkens, Kimmig, VdB, De Raedt; AAAI'14], [Nitti, VdB, et al.; ILP'11], [Renkens, VdB, Nijssen; ILP'11, MLJ'12], [VHaaren, VdB; ILP'11], [Vlasselaer, VdB, et al.; PLP'14] , [Choi, VdB, Darwiche; KRR'15], [De Raedt et al.;'15], [Kimmig et al.;'15], [VdB, Mohan, et al.;'15]



Outline

- Motivation
  - Why high-level representations?
  - **Why high-level reasoning?**
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Approximate symmetries
  - Lifted learning

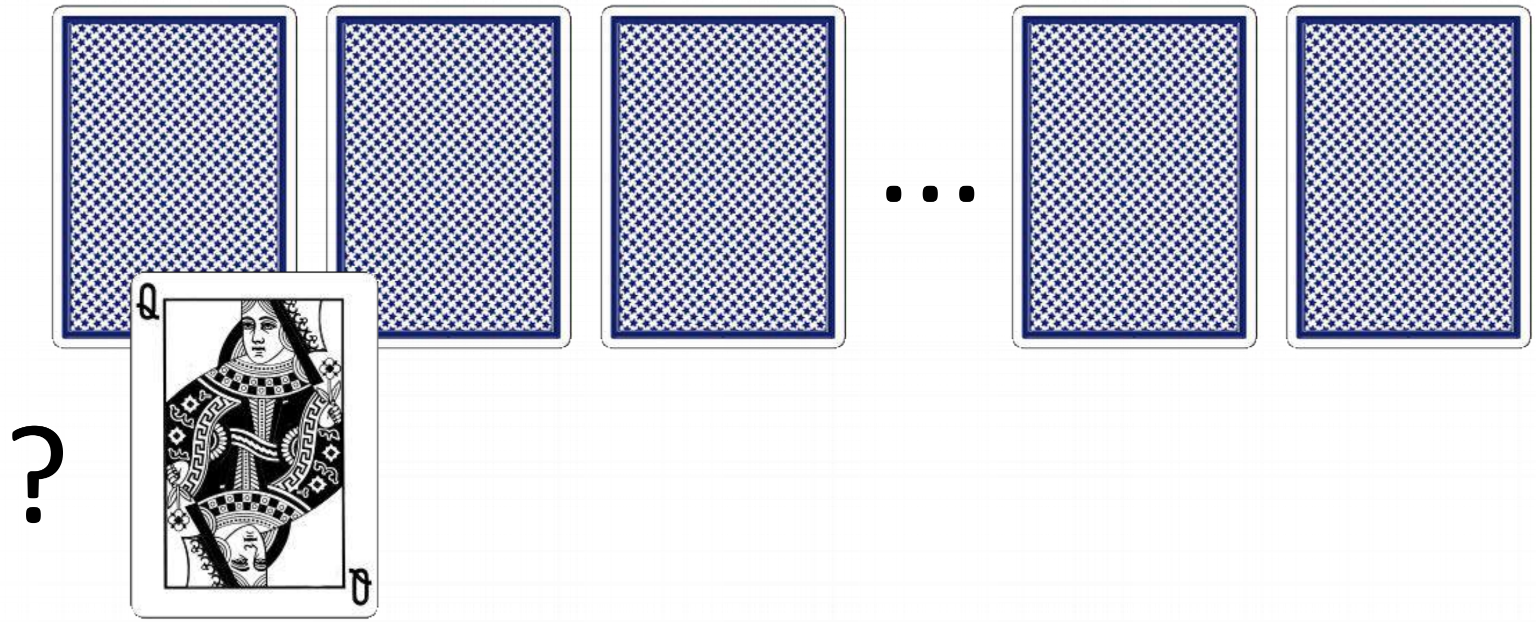
A Simple Reasoning Problem



- 52 playing cards
- Let us ask some simple questions

[Van den Broeck; AAAI-KRR'15]

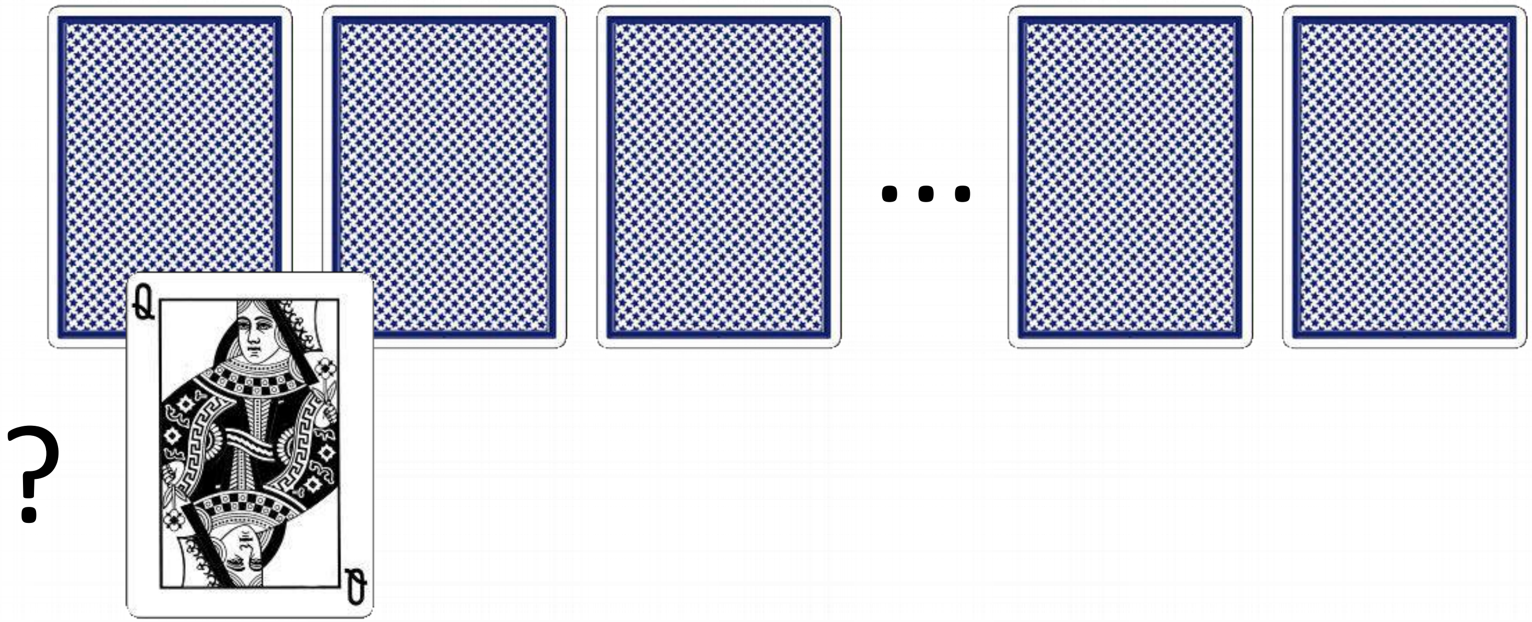
A Simple Reasoning Problem



Probability that Card1 is Q?

[Van den Broeck; AAAI-KRR'15]

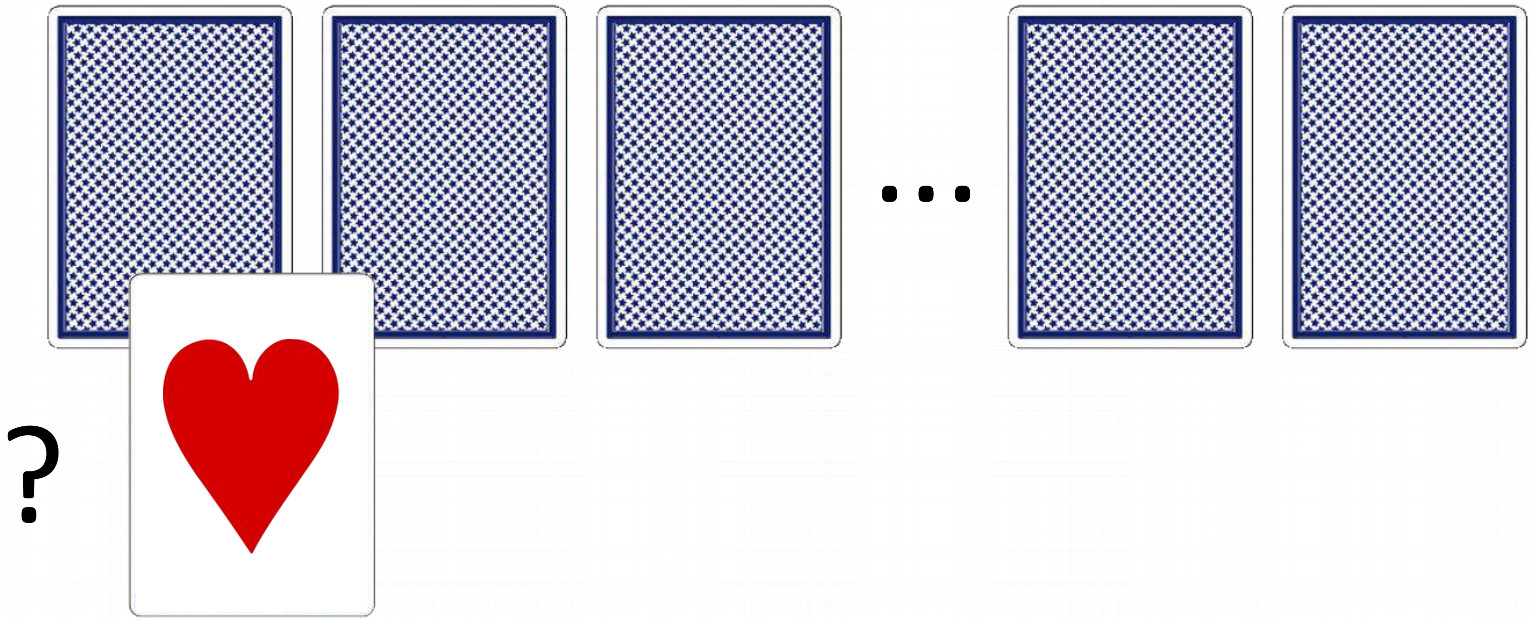
A Simple Reasoning Problem



Probability that Card1 is Q? 1/13

[Van den Broeck; AAAI-KRR'15]

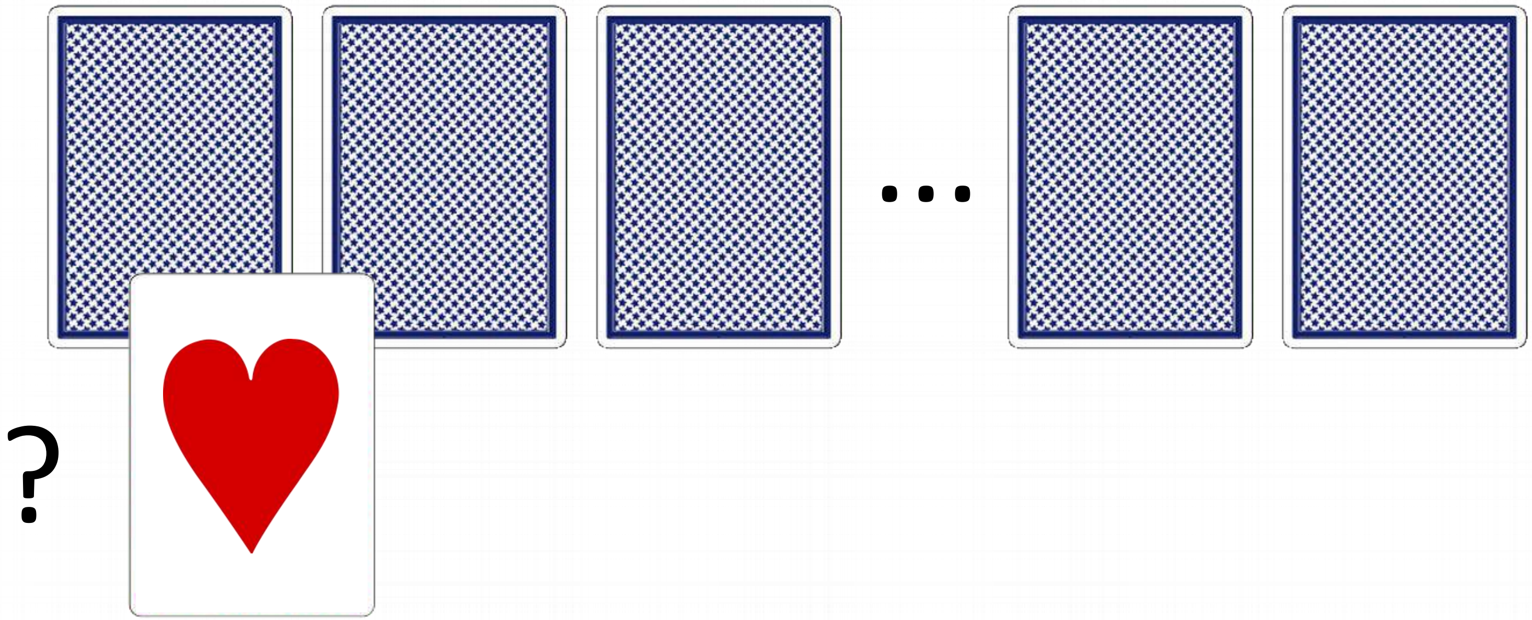
A Simple Reasoning Problem



Probability that Card1 is Hearts?

[Van den Broeck; AAAI-KRR’15]

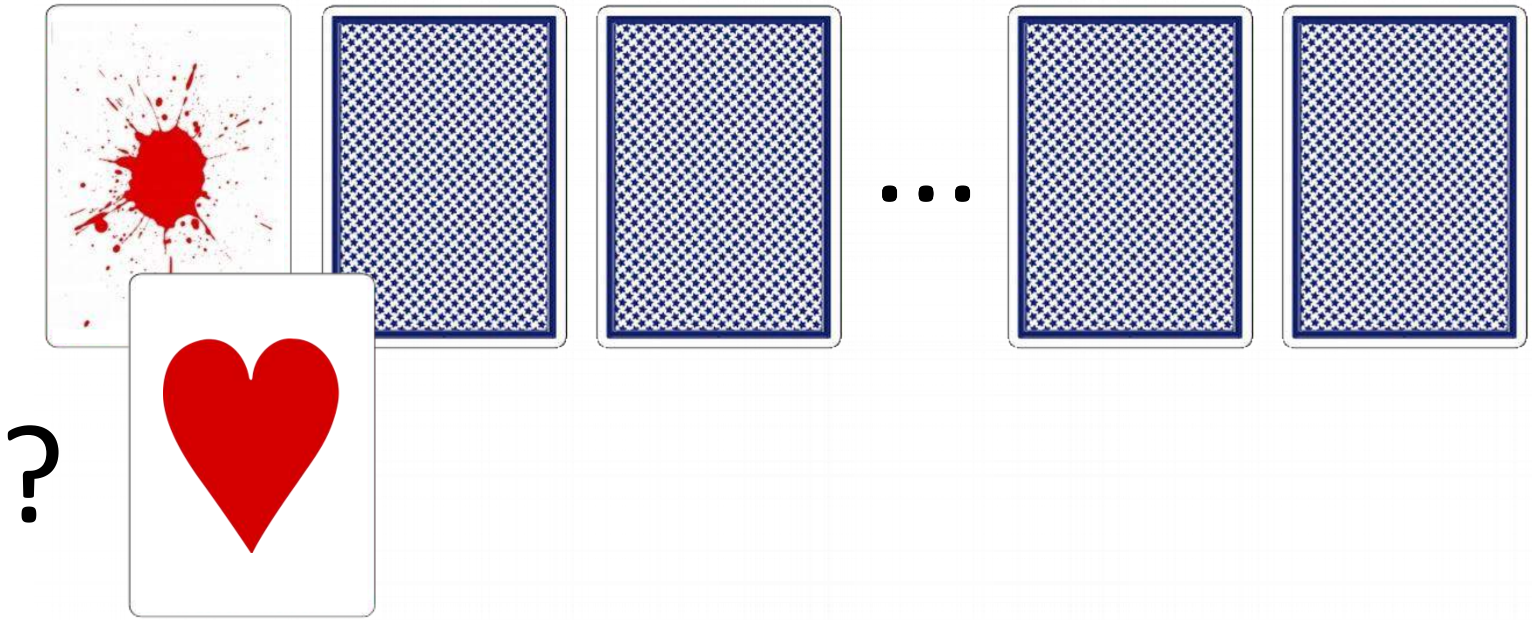
A Simple Reasoning Problem



Probability that Card1 is Hearts? 1/4

[Van den Broeck; AAAI-KRR'15]

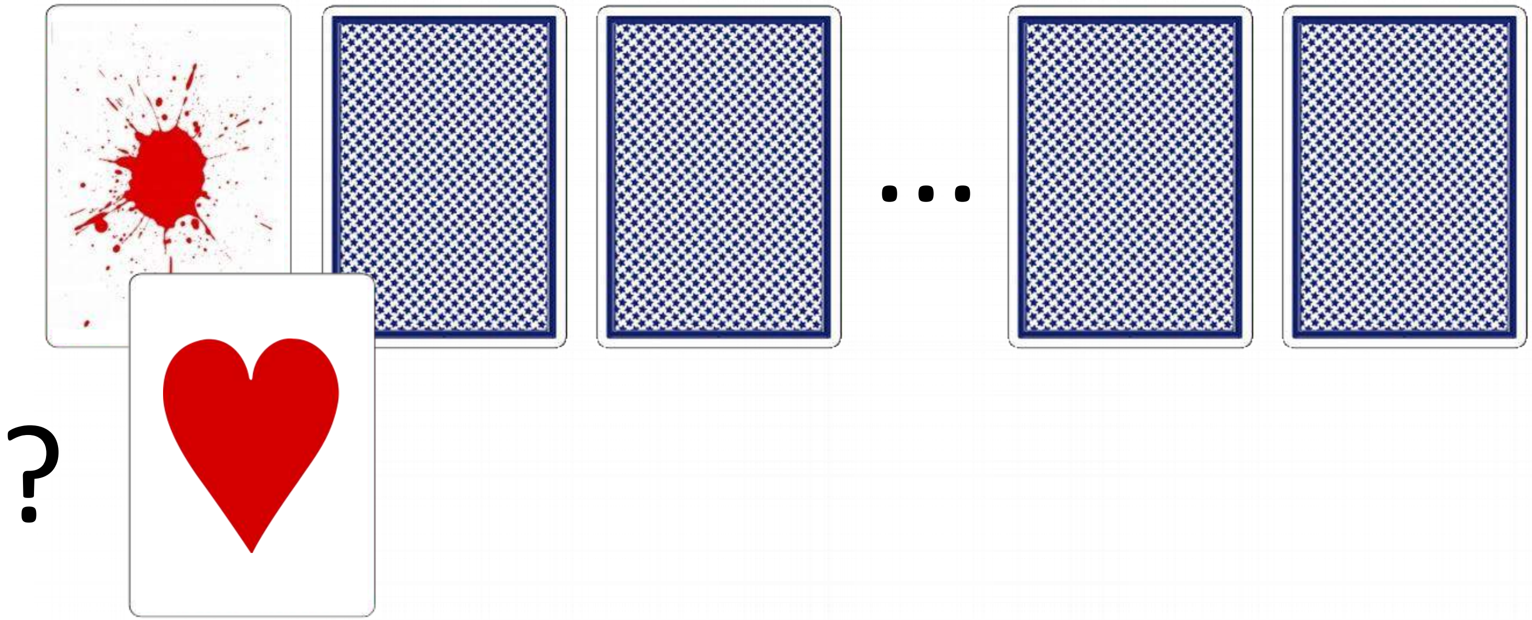
A Simple Reasoning Problem



Probability that Card1 is Hearts given that Card1 is red?

[Van den Broeck; AAAI-KRR'15]

A Simple Reasoning Problem

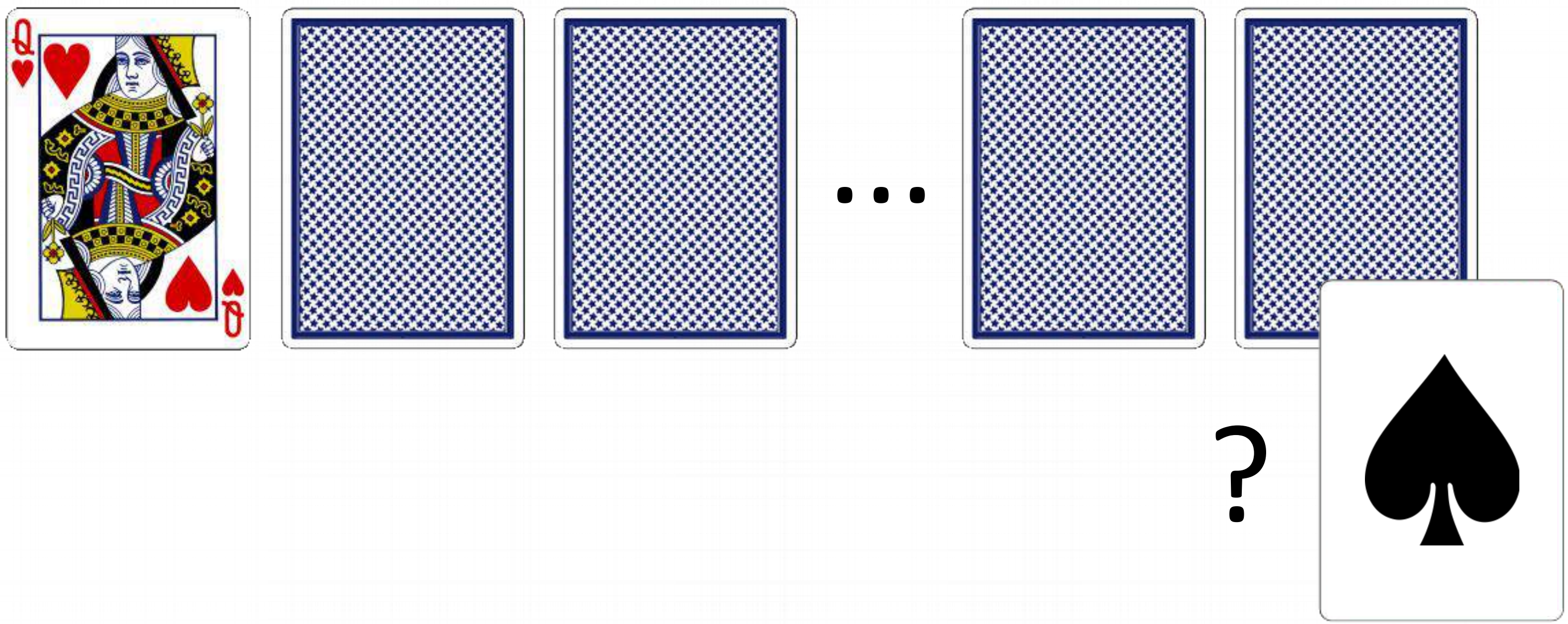


Probability that Card1 is Hearts given that Card1 is red?

1/2

[Van den Broeck; AAAI-KRR'15]

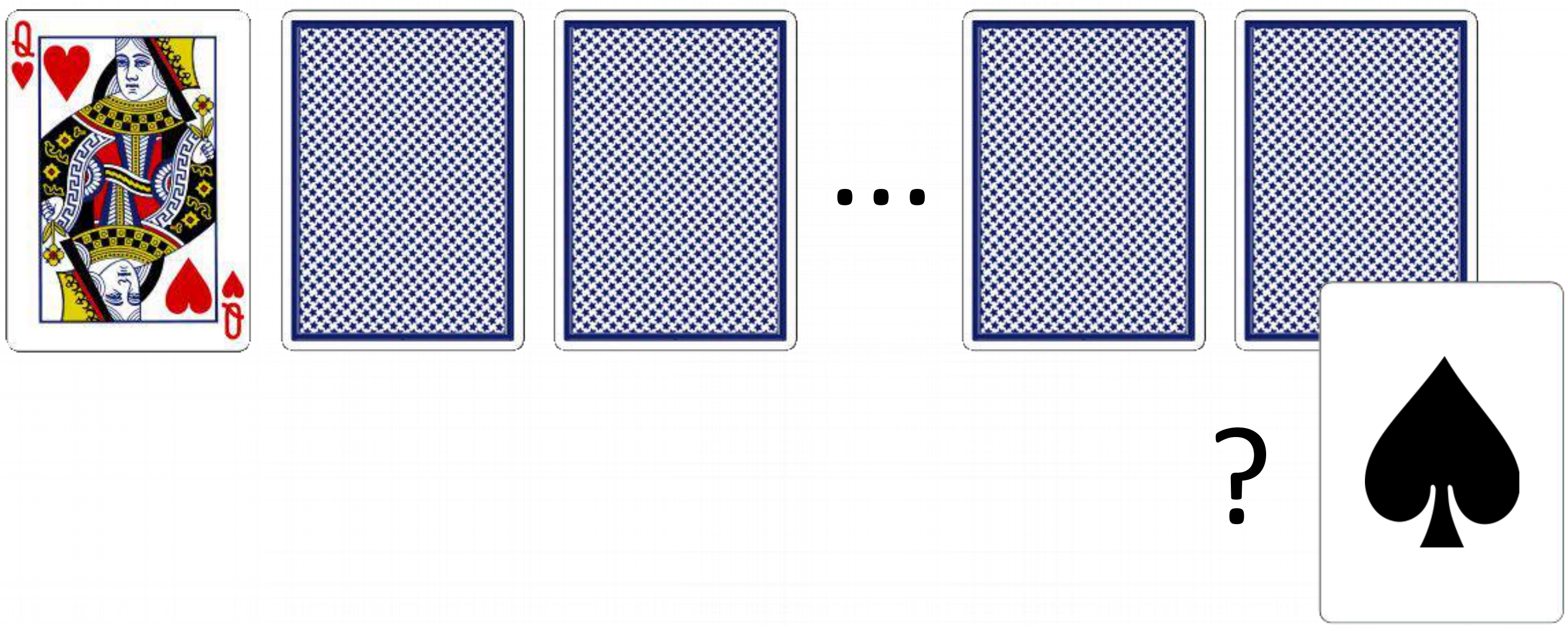
A Simple Reasoning Problem



Probability that Card52 is Spades given that Card1 is QH?

[Van den Broeck; AAAI-KRR'15]

A Simple Reasoning Problem



Probability that Card52 is Spades given that Card1 is QH?

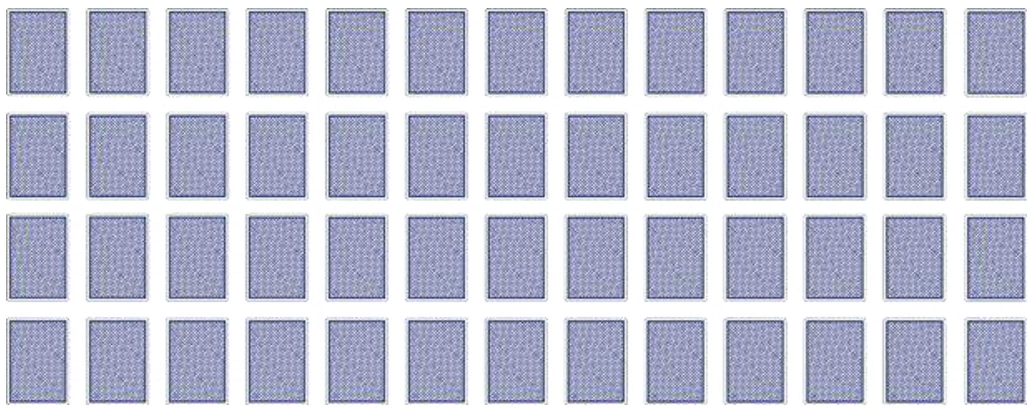
13/51

[Van den Broeck; AAAI-KRR'15]

Automated Reasoning

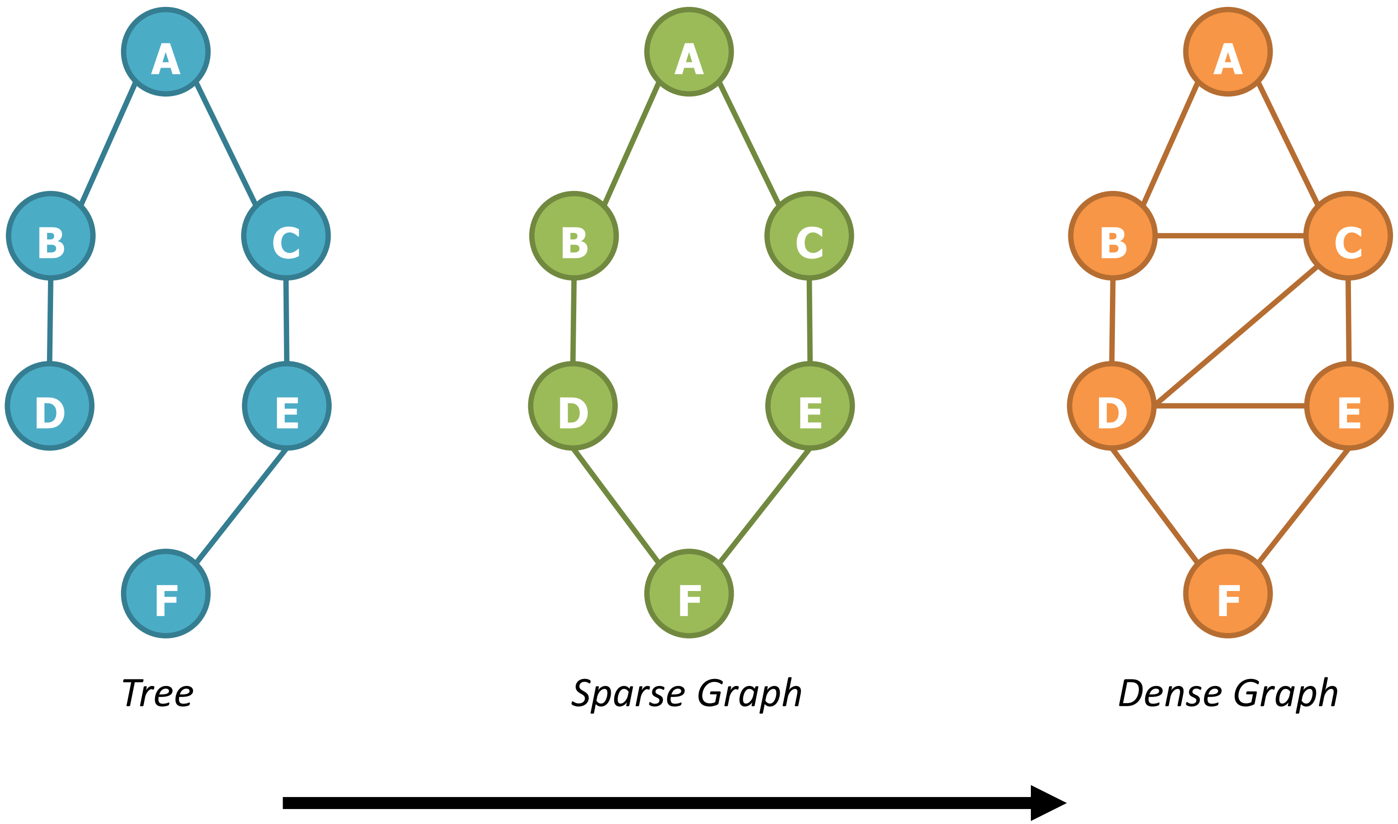
Let us automate this:

1. Probabilistic graphical model (e.g., factor graph)



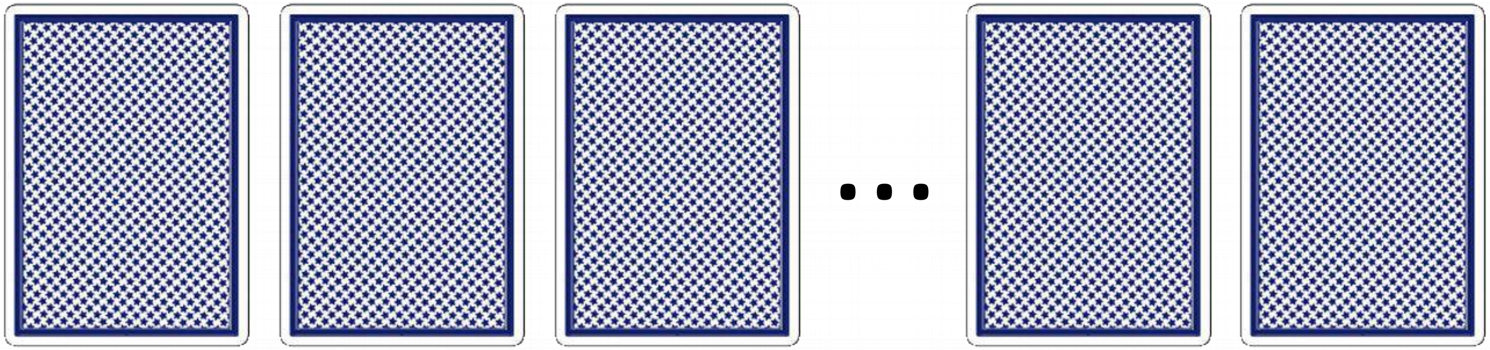
2. Probabilistic inference algorithm
   (e.g., variable elimination or junction tree)

Classical Reasoning



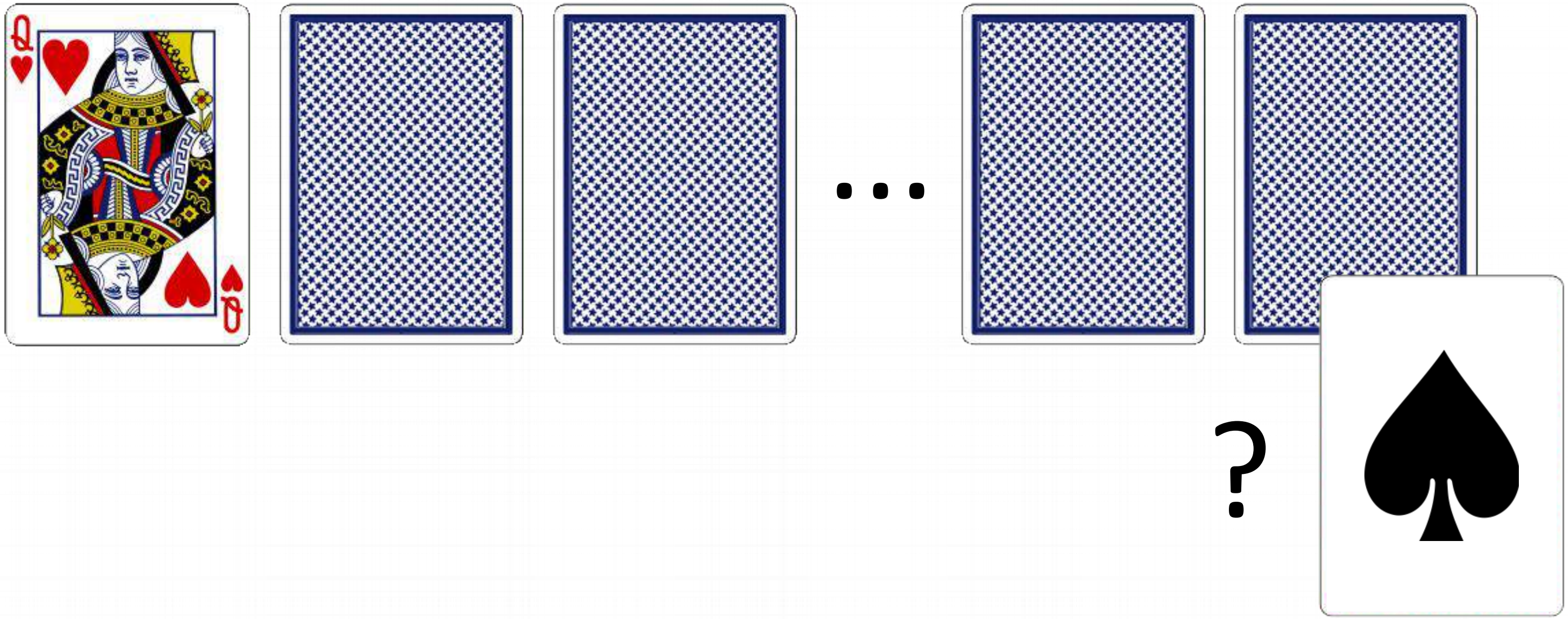
- Higher treewidth
- Fewer conditional independencies
- Slower inference

Is There Conditional Independence?



$$P(\text{Card52} \mid \text{Card1}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2})$$

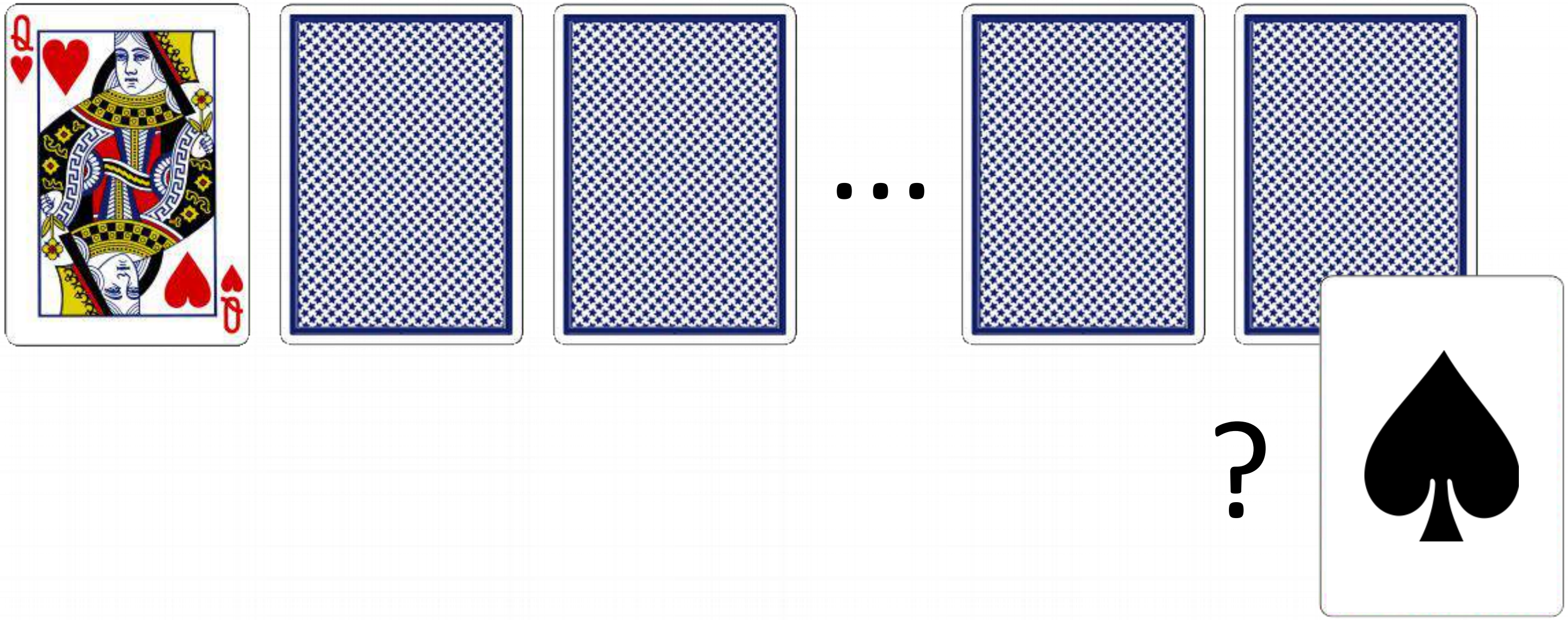
Is There Conditional Independence?



$$P(\text{Card52} \mid \text{Card1}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2})$$

$$? \stackrel{?}{=} ?$$

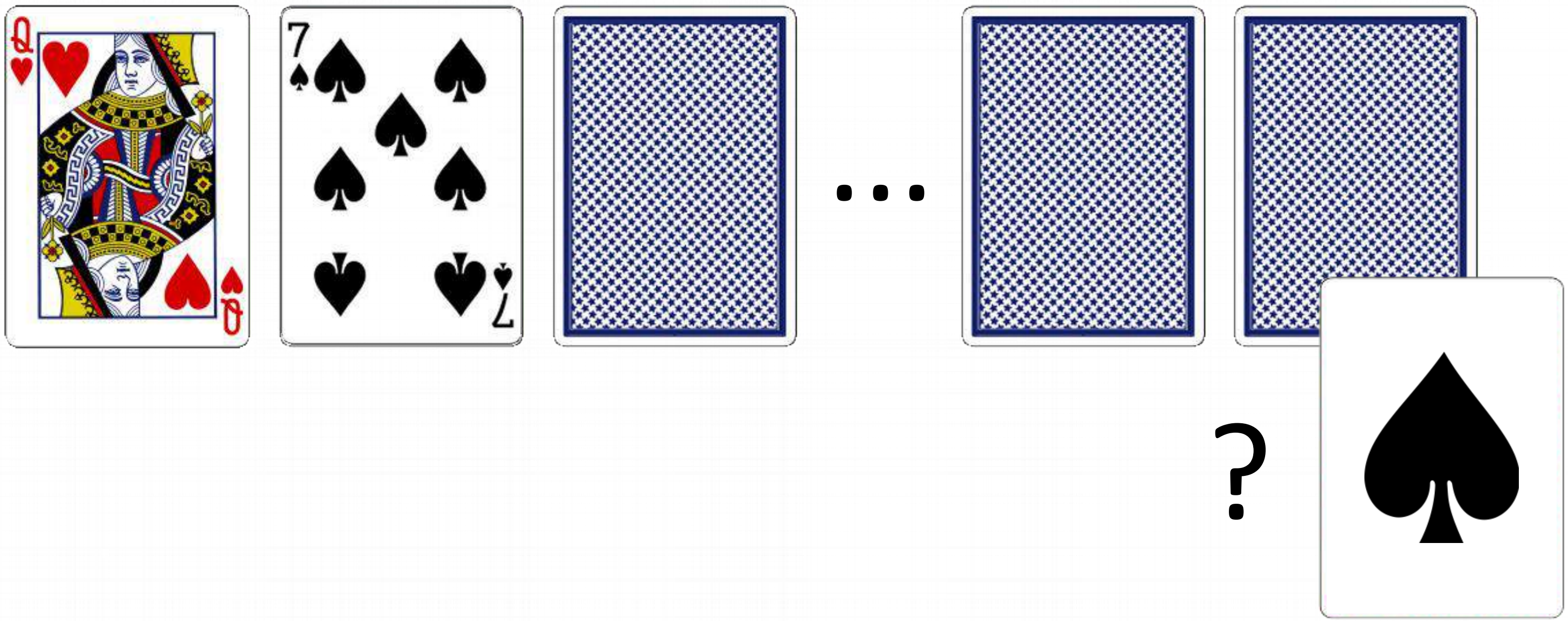
Is There Conditional Independence?



$$P(\text{Card52} \mid \text{Card1}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2})$$

$$13/51 \stackrel{?}{=} ?$$

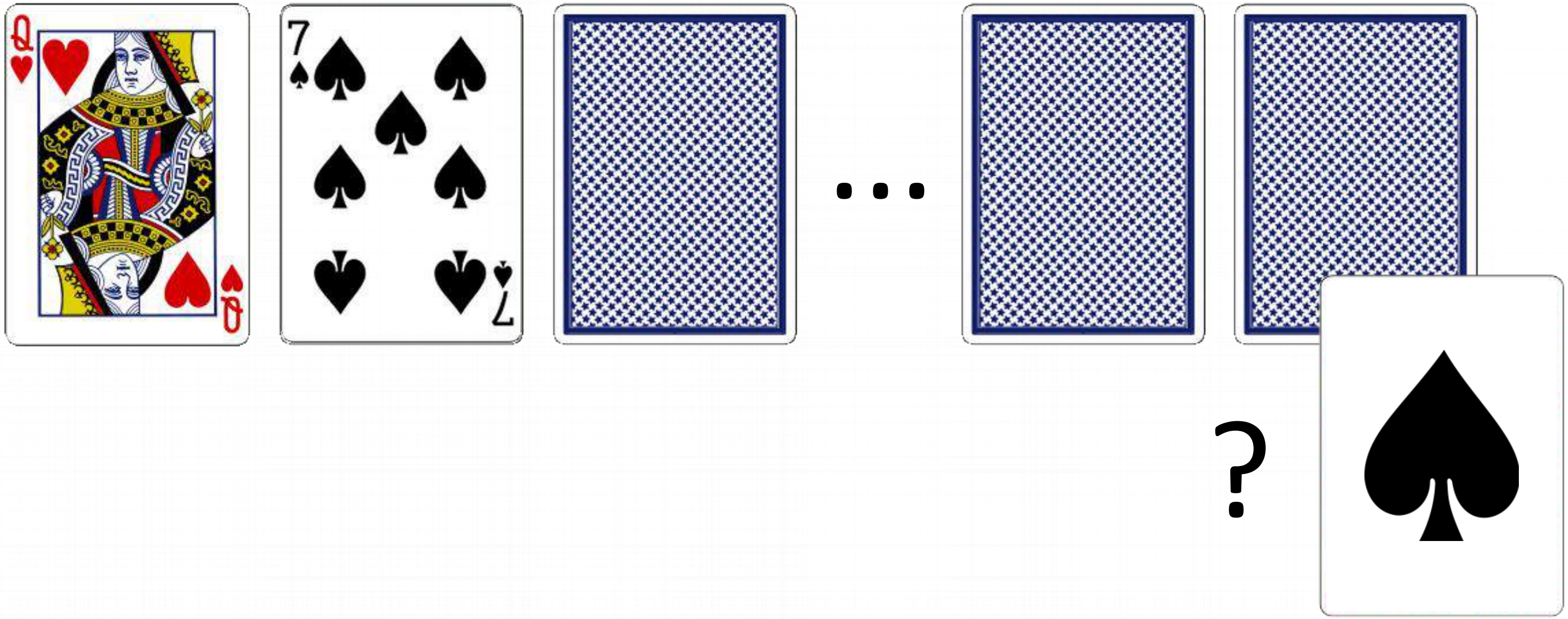
Is There Conditional Independence?



$P(\text{Card52} \mid \text{Card1}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2})$

$13/51 \stackrel{?}{=} ?$

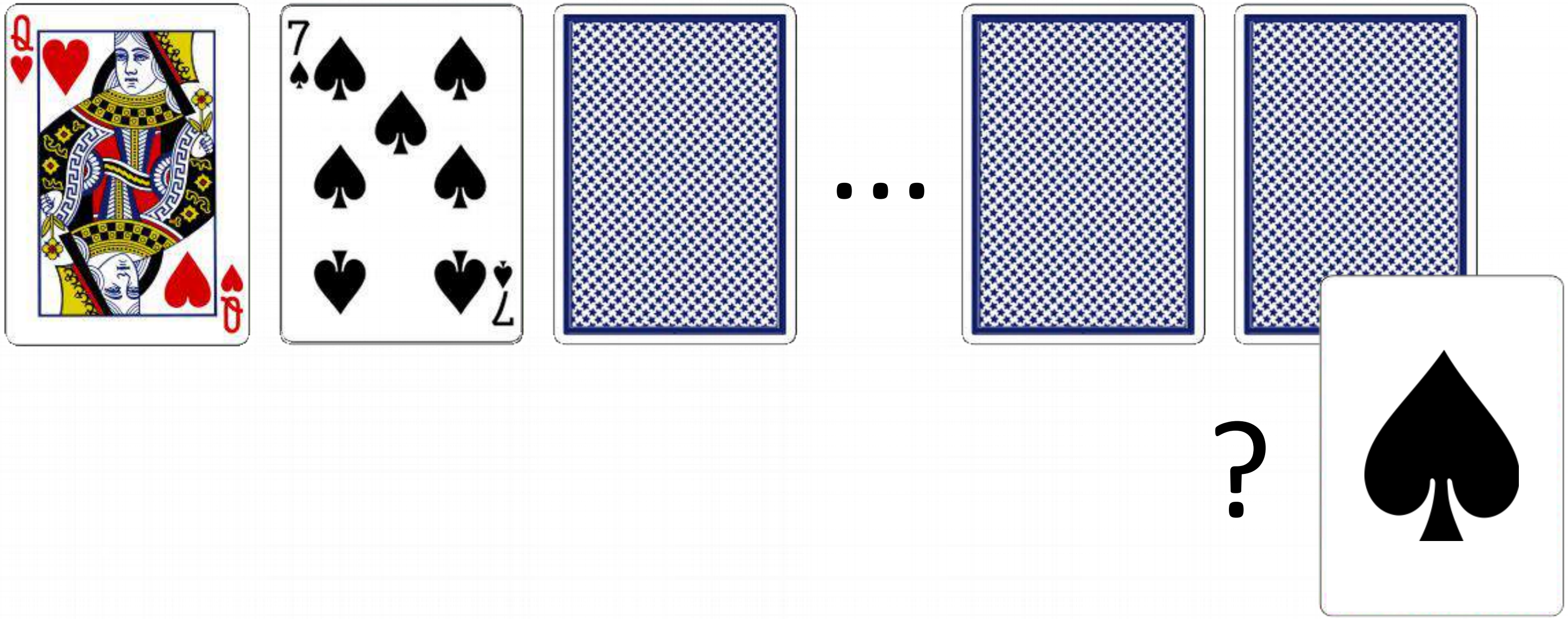
Is There Conditional Independence?



$$P(\text{Card52} \mid \text{Card1}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2})$$

$$13/51 \neq 12/50$$

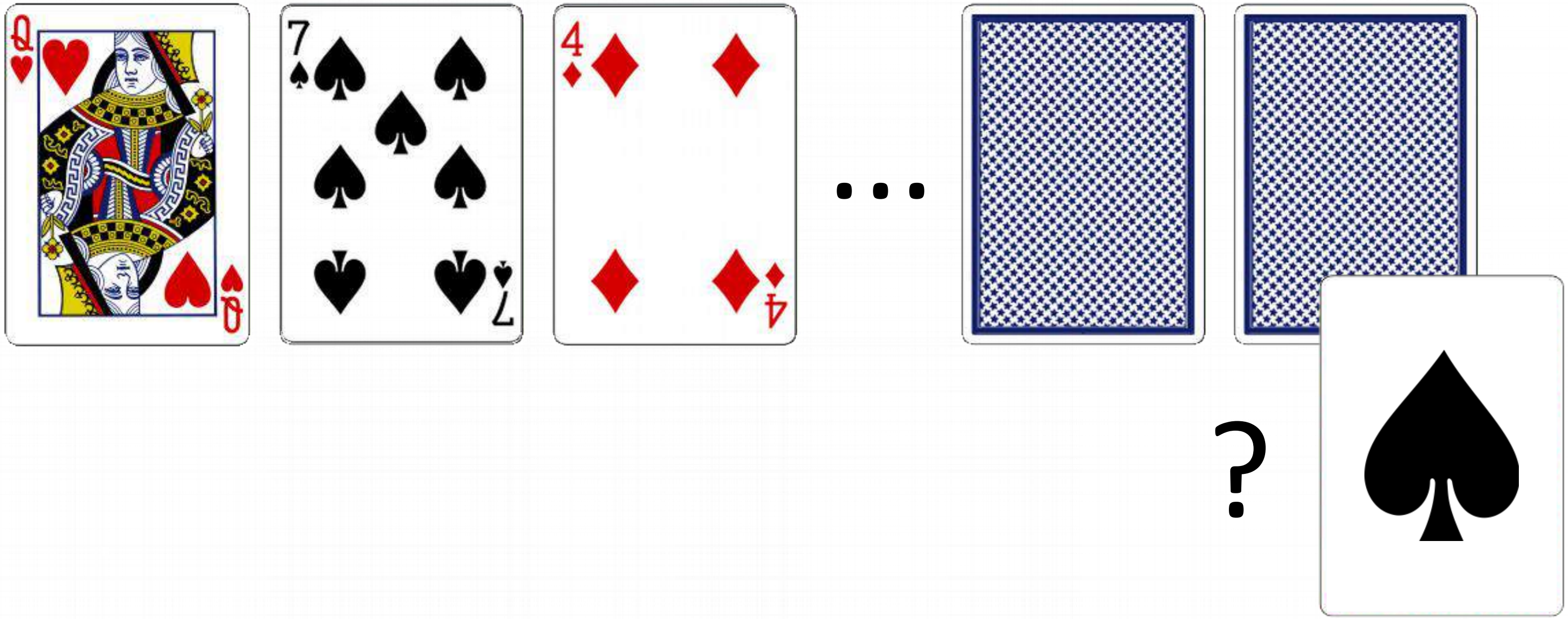
Is There Conditional Independence?



P(Card52 | Card1) ≠ P(Card52 | Card1, Card2)

13/51 ≠ 12/50

Is There Conditional Independence?

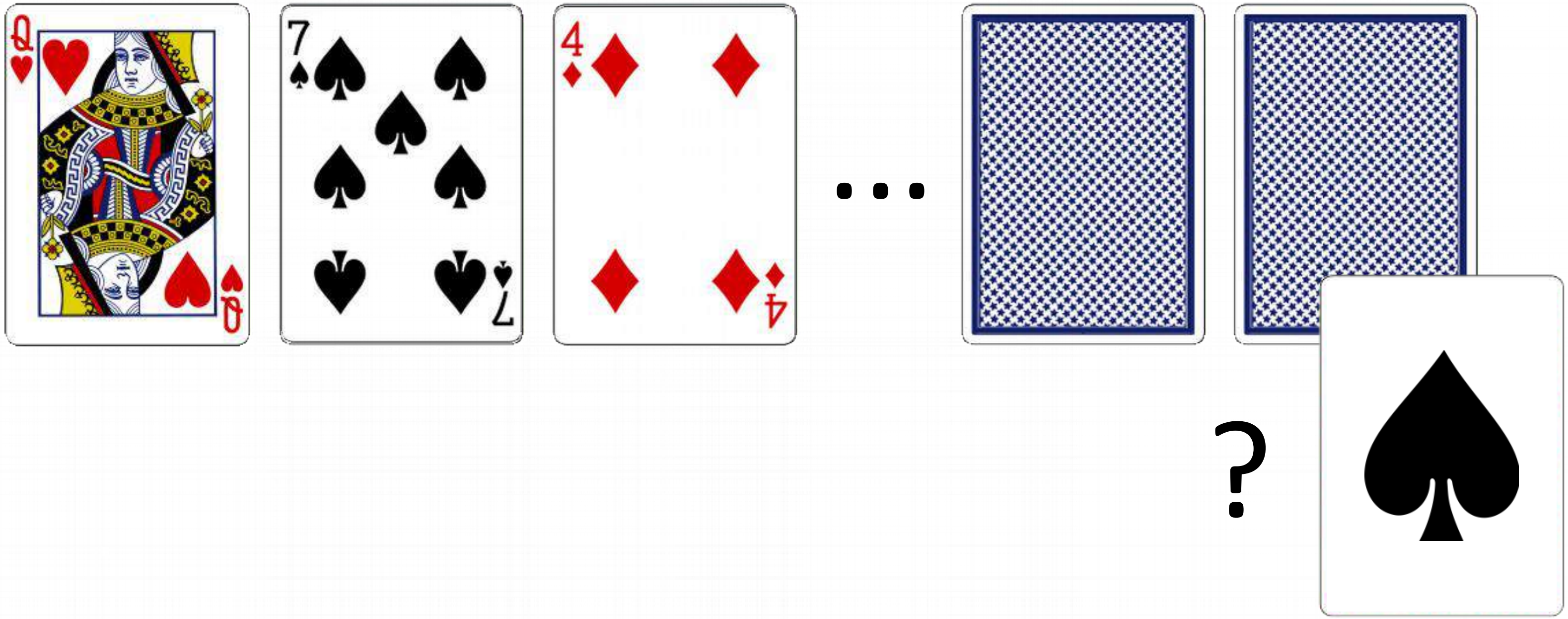


$$P(\text{Card52} \mid \text{Card1}) \neq P(\text{Card52} \mid \text{Card1}, \text{Card2})$$

$$13/51 \neq 12/50$$

$$P(\text{Card52} \mid \text{Card1}, \text{Card2}) \stackrel{?}{=} P(\text{Card52} \mid \text{Card1}, \text{Card2}, \text{Card3})$$

Is There Conditional Independence?



P(Card52 | Card1) ≠ P(Card52 | Card1, Card2)

13/51 ≠ 12/50

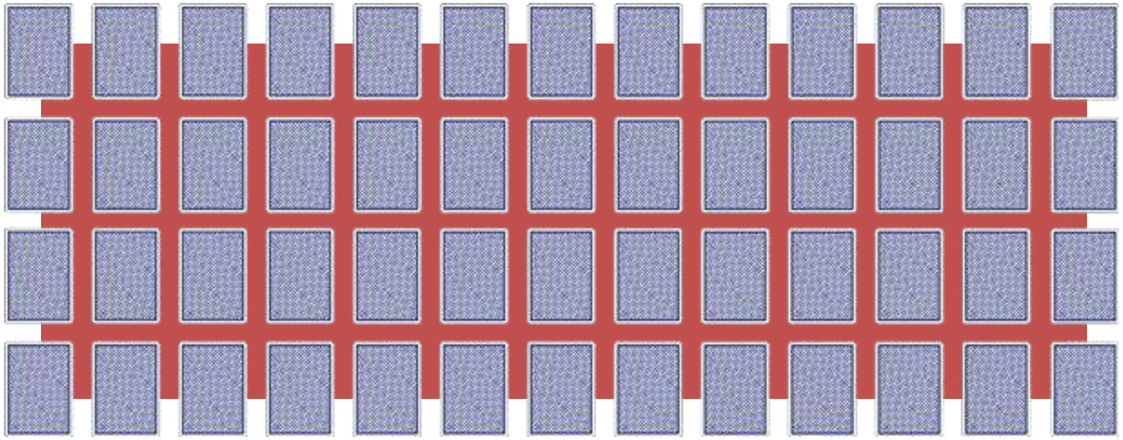
P(Card52 | Card1, Card2) ≠ P(Card52 | Card1, Card2, Card3)

12/50 ≠ 12/49

Automated Reasoning

Let us automate this:

1. Probabilistic graphical model (e.g., factor graph)
   is fully connected!

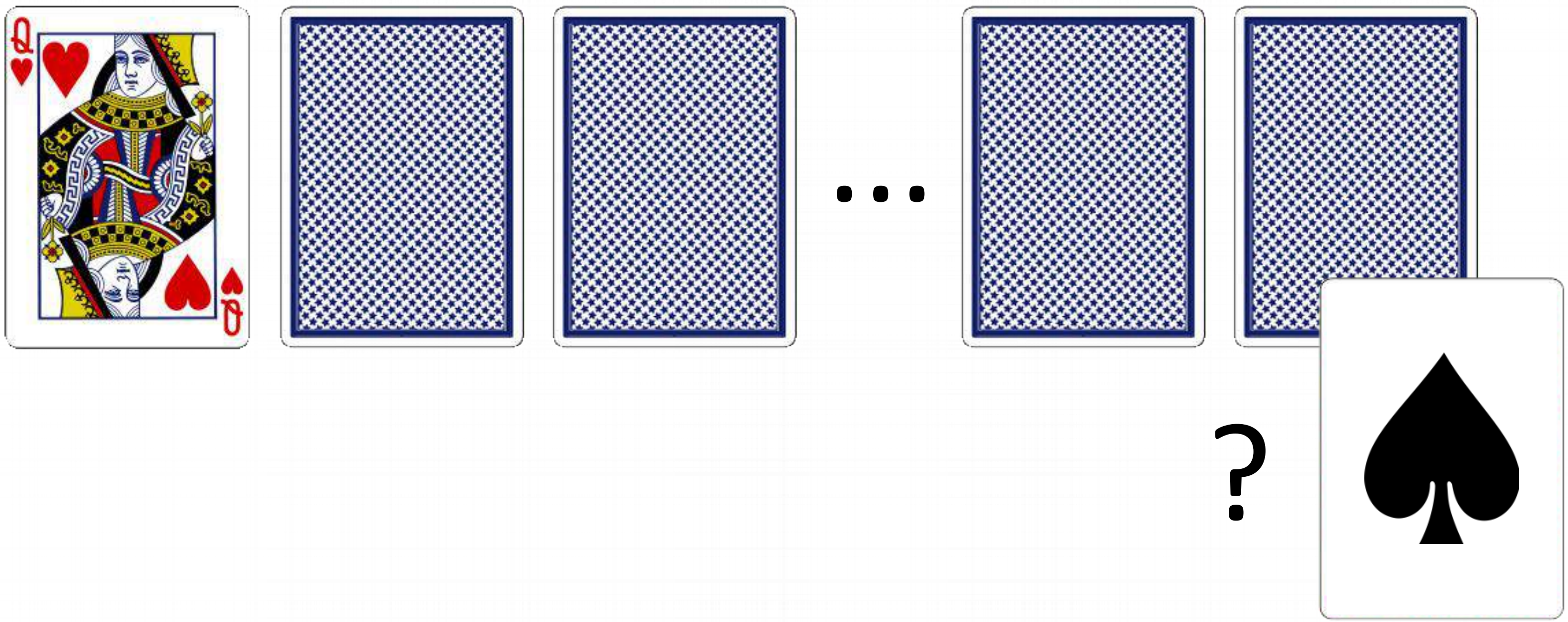


(artist's impression)

2. Probabilistic inference algorithm
   (e.g., variable elimination or junction tree)
   builds a table with $\mathbf{52^{52}}$ rows

[Van den Broeck; AAAI-KRR'15]

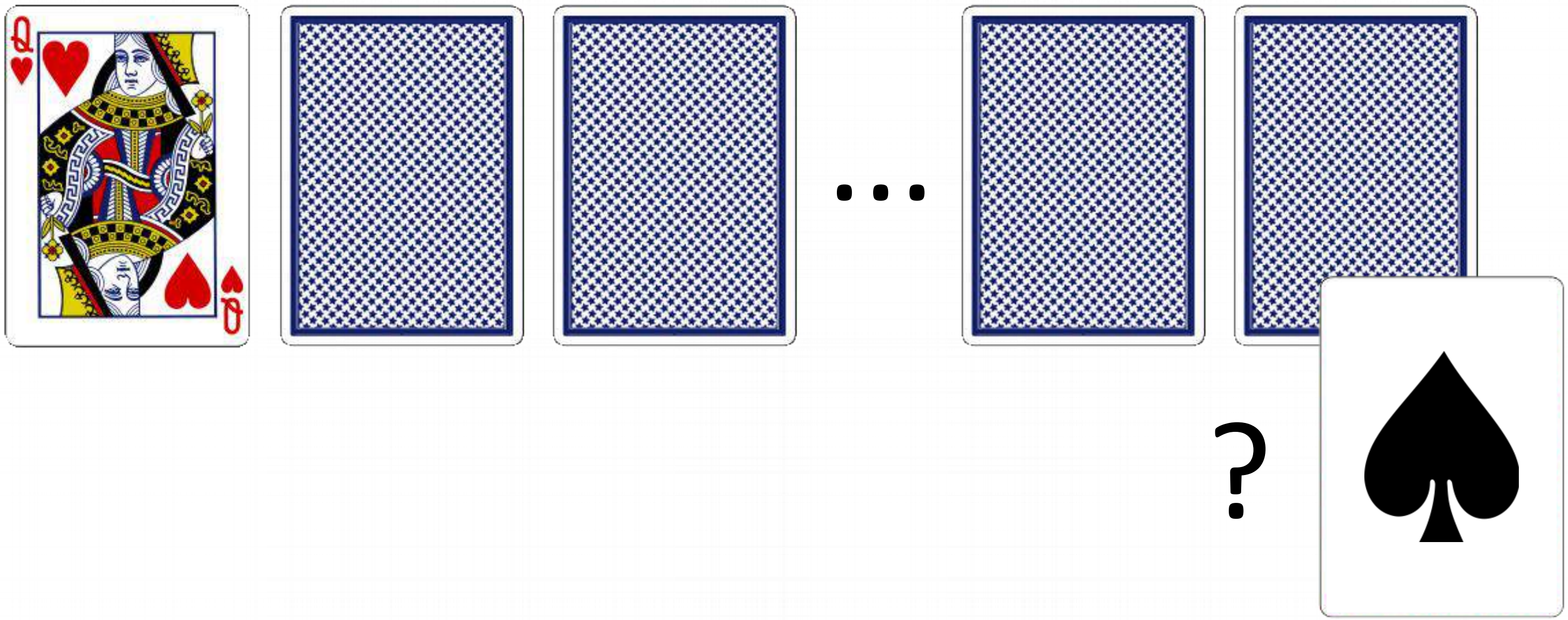
What's Going On Here?



Probability that Card52 is Spades given that Card1 is QH?

[Van den Broeck; AAAI-KRR’15]

What's Going On Here?

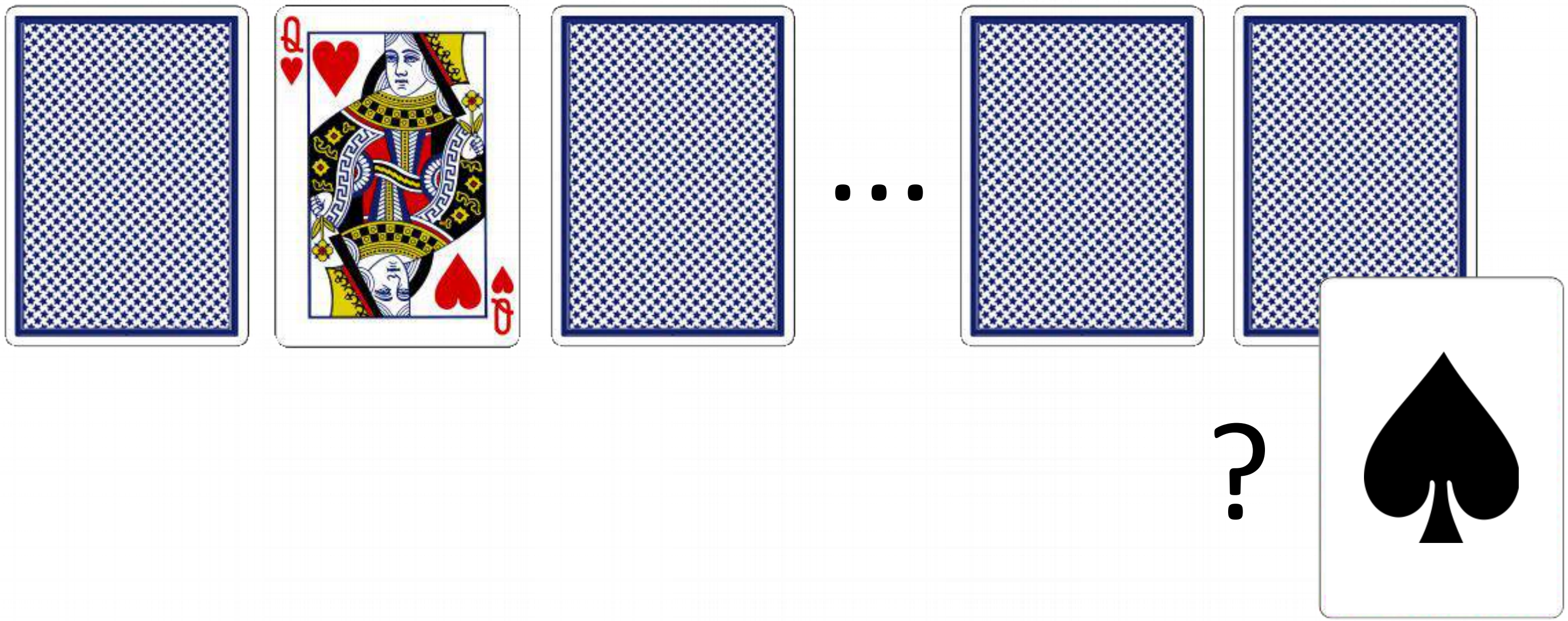


Probability that Card52 is Spades given that Card1 is QH?

13/51

[Van den Broeck; AAAI-KRR'15]

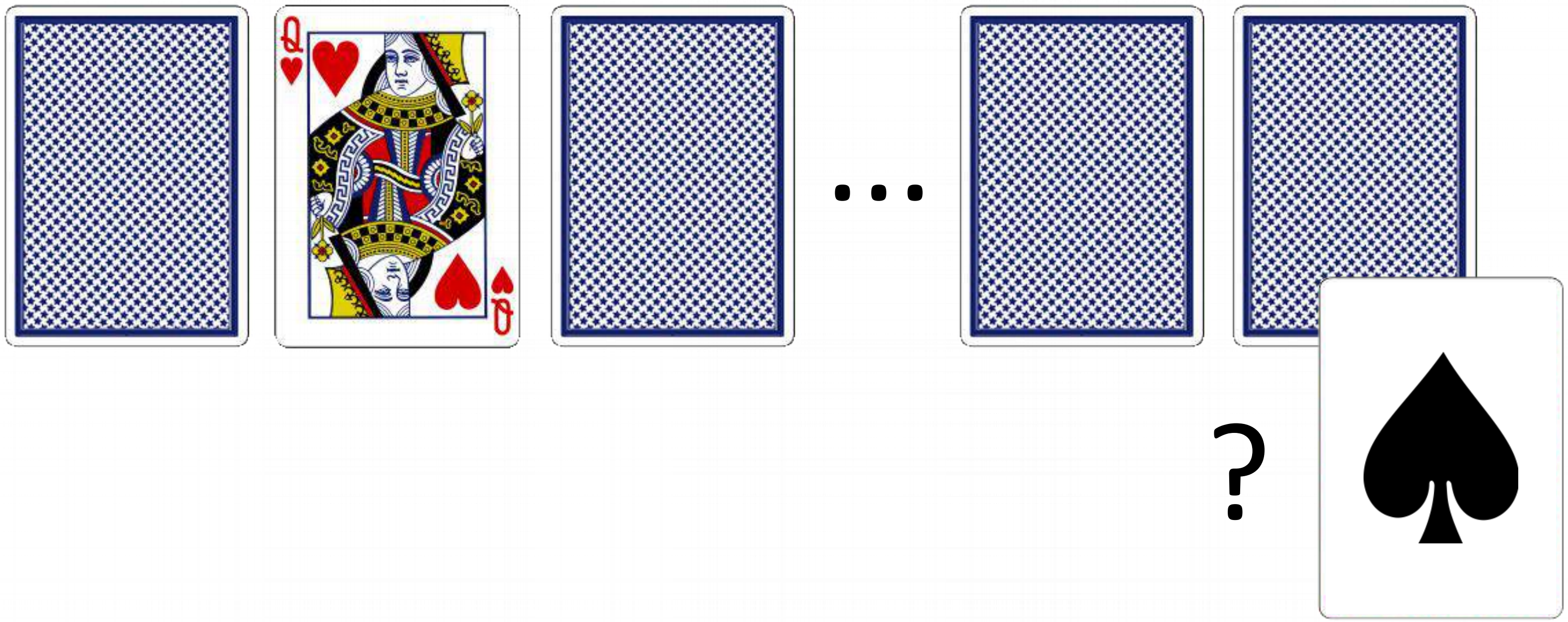
What's Going On Here?



Probability that Card52 is Spades given that Card2 is QH?

[Van den Broeck; AAAI-KRR'15]

What's Going On Here?

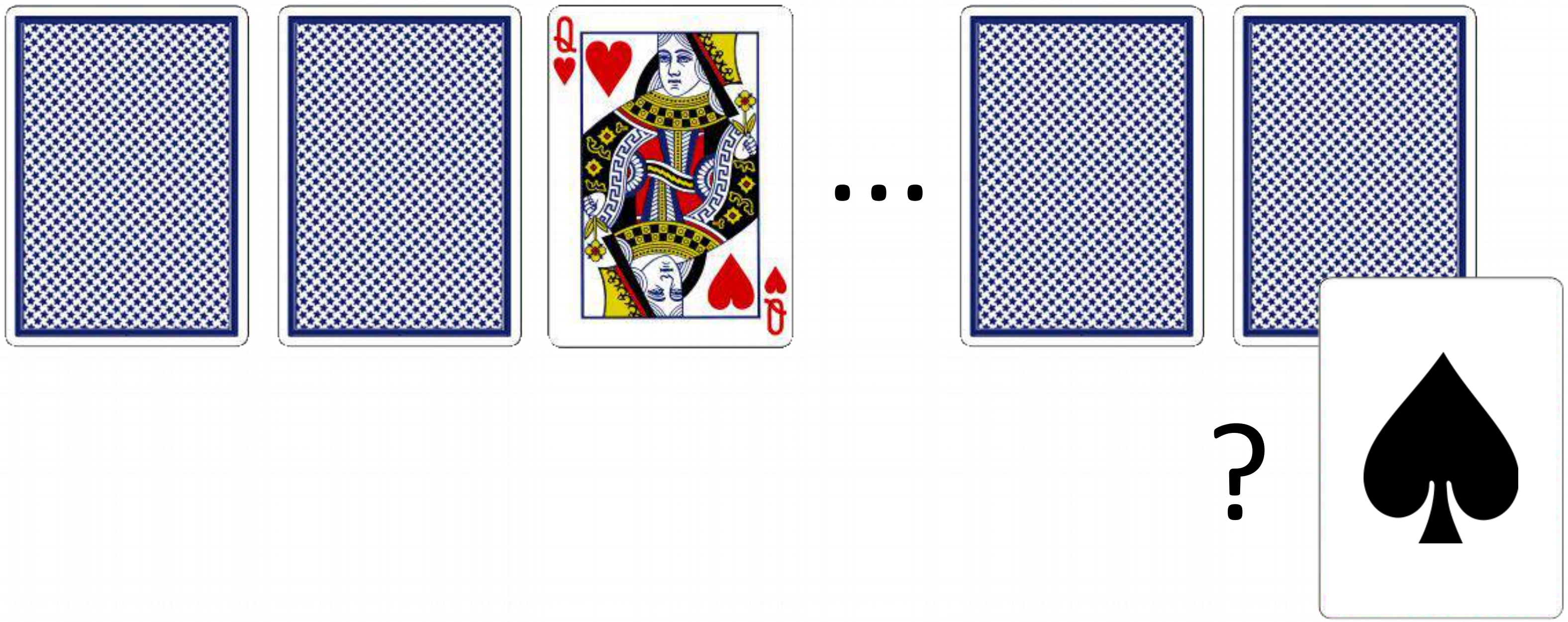


Probability that Card52 is Spades given that Card2 is QH?

13/51

[Van den Broeck; AAAI-KRR’15]

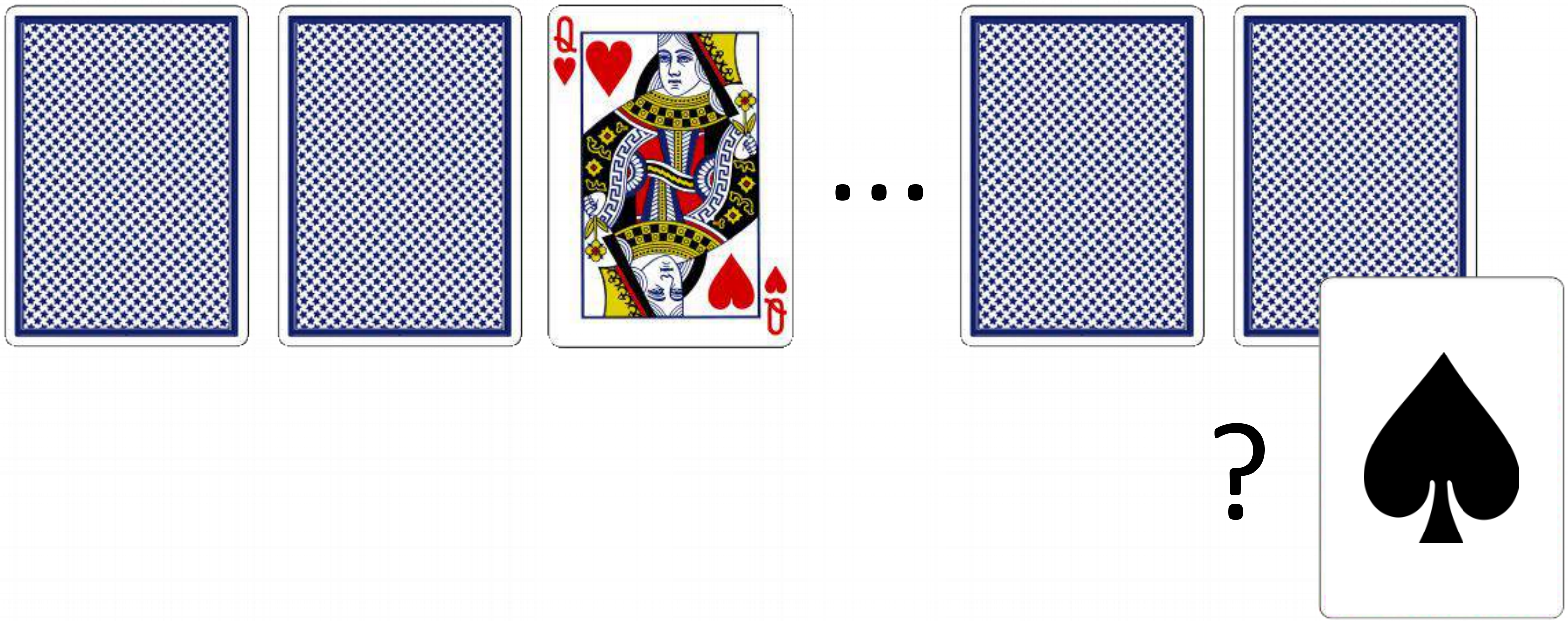
What's Going On Here?



Probability that Card52 is Spades given that Card3 is QH?

[Van den Broeck; AAAI-KRR'15]

What's Going On Here?

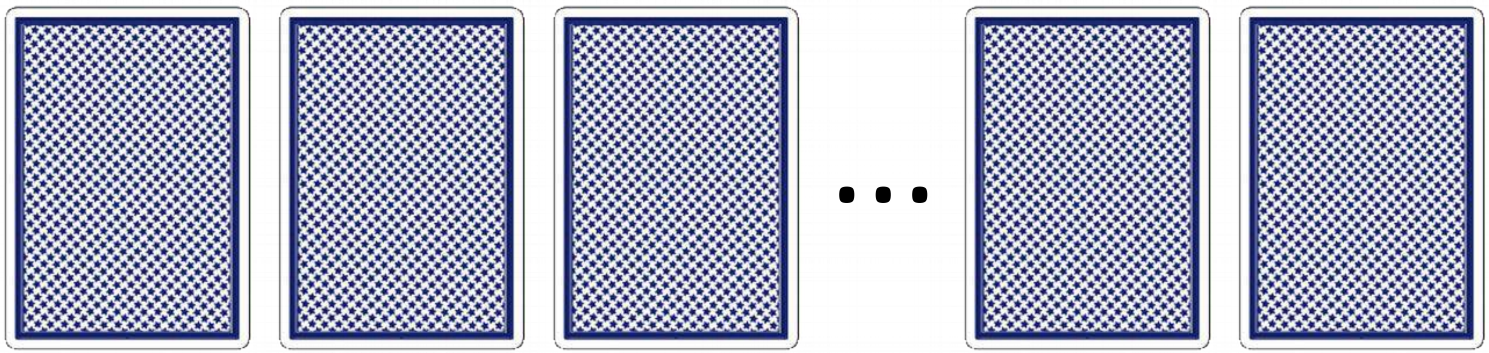


Probability that Card52 is Spades given that Card3 is QH?

13/51

[Van den Broeck; AAAI-KRR'15]

Tractable Probabilistic Inference



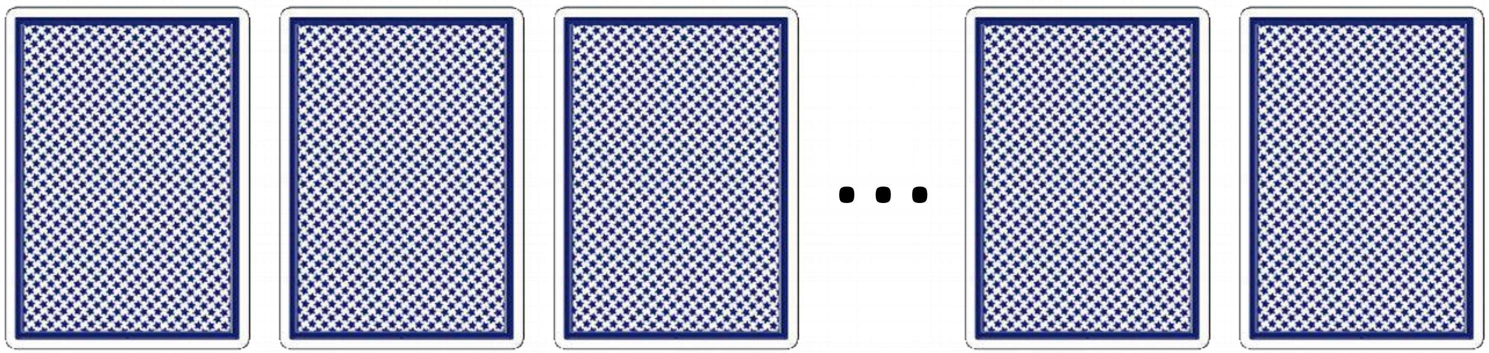
Which property makes inference tractable?

~~Traditional belief: Independence~~

What's going on here?

[Niepert, Van den Broeck; AAAI'14], [Van den Broeck; AAAI-KRR'15]

Tractable Probabilistic Inference



Which property makes inference tractable?

~~Traditional belief: Independence~~

What's going on here?

- High-level reasoning
- Symmetry
- Exchangeability

⇒ Lifted Inference

[Niepert, Van den Broeck; AAAI'14], [Van den Broeck; AAAI-KRR'15]

Other Examples of Lifted Inference

- Syllogisms & First-order resolution
- Reasoning about populations

We are investigating a rare disease. The disease is more rare in women, presenting only in **one in every two billion women** and **one in every billion men**. Then, assuming there are **3.4 billion men** and **3.6 billion women** in the world, the probability that **more than five people** have the disease is

$$1 - \sum_{n=0}^{5} \sum_{f=0}^{n} \binom{3.6 \cdot 10^9}{f} \left(1 - 0.5 \cdot 10^{-9}\right)^{3.6 \cdot 10^9 - f} \left(0.5 \cdot 10^{-9}\right)^{f}$$

$$\times \binom{3.4 \cdot 10^9}{(n-f)} \left(1 - 10^{-9}\right)^{3.4 \cdot 10^9 - (n-f)} \left(10^{-9}\right)^{(n-f)}$$

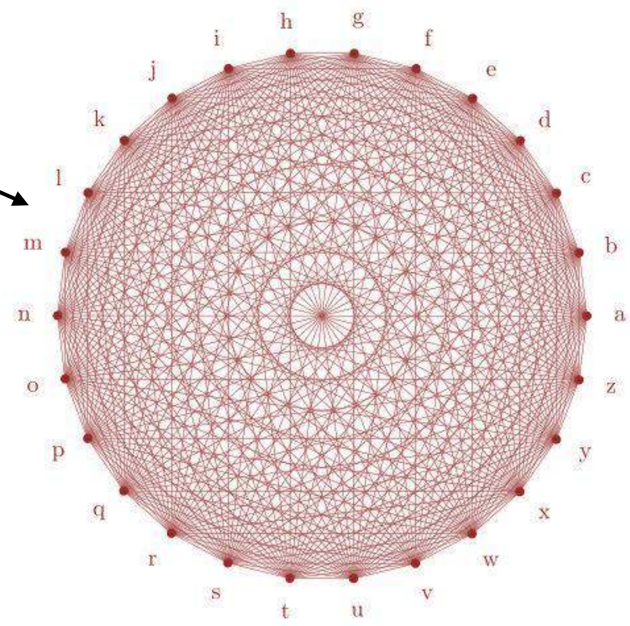
[Van den Broeck; AAAI-KRR'15], [Van den Broeck; PhD'13]

Equivalent Graphical Model

- Statistical relational model (e.g., MLN)

3.14 FacultyPage(x) ∧ Linked(x,y) ⇒ CoursePage(y)

- As a probabilistic graphical model:
  - 26 pages; 728 variables; 676 factors
  - 1000 pages; 1,002,000 variables; 1,000,000 factors



- Highly intractable?
  - **Lifted inference** in milliseconds!

Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- **Intuition: Inference rules**
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Approximate symmetries
  - Lifted learning

Weighted Model Counting

- Model = solution to a propositional logic formula Δ
- Model counting = #SAT

Δ = (Rain ⇒ Cloudy)

Rain	Cloudy	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

+

#SAT = 3

Weighted Model Counting

- Model = solution to a propositional logic formula Δ
- Model counting = #SAT
- Weighted model counting (WMC)
  - Weights for assignments to variables
  - Model weight is product of variable weights w(.)

Δ = (Rain ⇒ Cloudy)

w(R)=1
w(¬R)=2
w(C)=3
w(¬C)=5

Rain	Cloudy	Model?	Weight
T	T	Yes	1 * 3 = 3
T	F	No	0
F	T	Yes	2 * 3 = 6
F	F	Yes	2 * 5 = 10

+

#SAT = 3

Weighted Model Counting

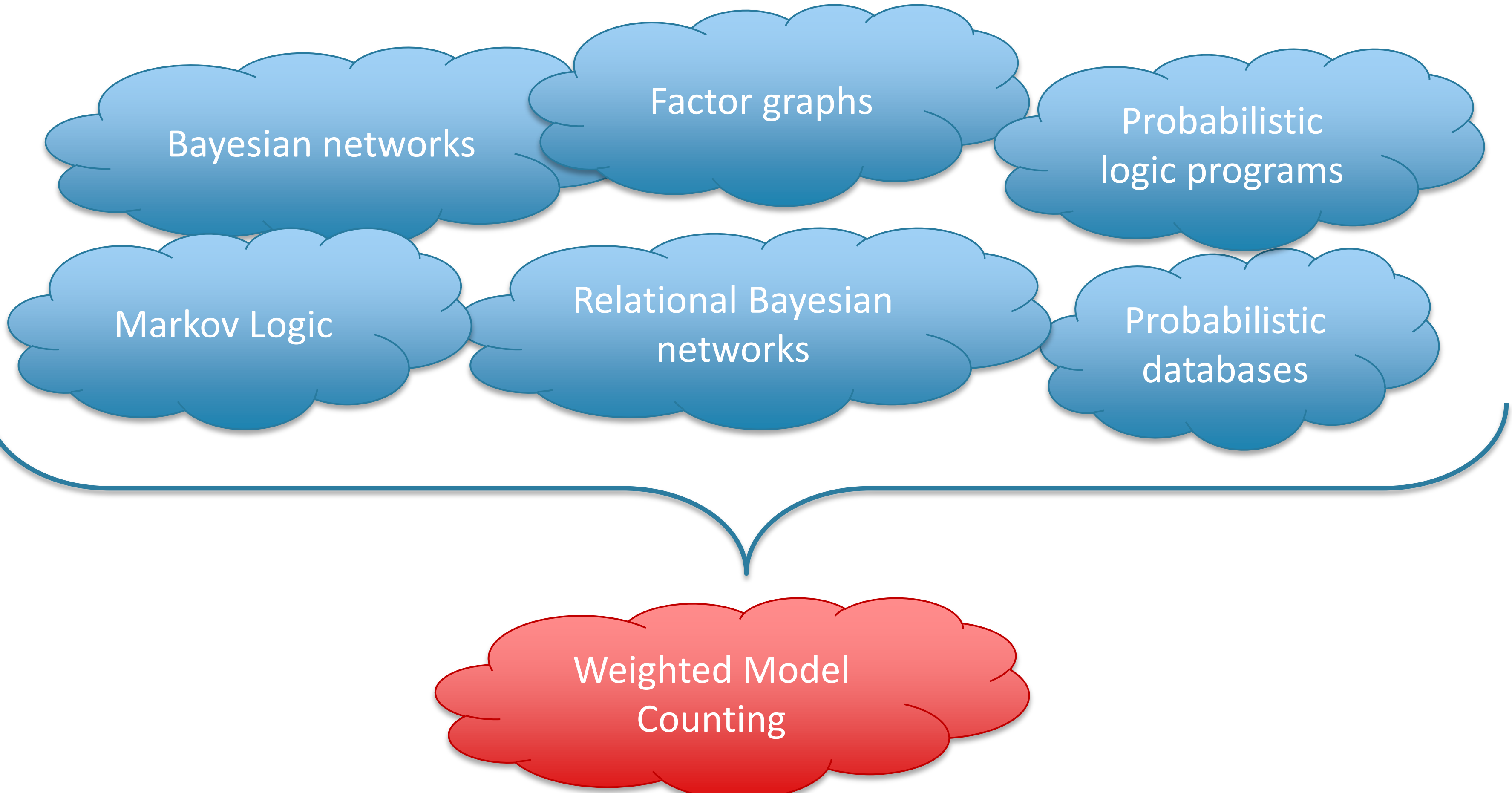
- Model = solution to a propositional logic formula Δ
- Model counting = #SAT
- Weighted model counting (WMC)
  - Weights for assignments to variables
  - Model weight is product of variable weights w(.)

Δ = (Rain ⇒ Cloudy)

w(R)=1
w(¬R)=2
w(C)=3
w(¬C)=5

Rain	Cloudy	Model?	Weight
T	T	Yes	1 * 3 = 3
T	F	No	0
F	T	Yes	2 * 3 = 6
F	F	Yes	2 * 5 = 10
		+	+
		#SAT = 3	**WMC = 19**

Assembly language for probabilistic reasoning



Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

$\Delta = \forall d\ (\text{Rain}(d)$
$\quad\Rightarrow \text{Cloudy}(d))$

Days = {Monday}

Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

Δ = ∀d (Rain(d) ⇒ Cloudy(d))

Days = {Monday}

Rain(M)	Cloudy(M)	Model?
T	T	Yes
T	F	No
F	T	Yes
F	F	Yes

+

#SAT = 3

Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

Δ = ∀d (Rain(d)
⇒ Cloudy(d))

Days = {Monday
Tuesday}

Rain(M)	Cloudy(M)	Rain(T)	Cloudy(T)	Model?
T	T	T	T	Yes
T	F	T	T	No
F	T	T	T	Yes
F	F	T	T	Yes
T	T	T	F	No
T	F	T	F	No
F	T	T	F	No
F	F	T	F	No
T	T	F	T	Yes
T	F	F	T	No
F	T	F	T	Yes
F	F	F	T	Yes
T	T	F	F	Yes
T	F	F	F	No
F	T	F	F	Yes
F	F	F	F	Yes

Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

Δ = ∀d (Rain(d) ⇒ Cloudy(d))

Days = {Monday Tuesday}

Rain(M)	Cloudy(M)	Rain(T)	Cloudy(T)	Model?
T	T	T	T	Yes
T	F	T	T	No
F	T	T	T	Yes
F	F	T	T	Yes
T	T	T	F	No
T	F	T	F	No
F	T	T	F	No
F	F	T	F	No
T	T	F	T	Yes
T	F	F	T	No
F	T	F	T	Yes
F	F	F	T	Yes
T	T	F	F	Yes
T	F	F	F	No
F	T	F	F	Yes
F	F	F	F	Yes

+

#SAT = 9

Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

Δ = ∀d (Rain(d) ⇒ Cloudy(d))

Days = {Monday Tuesday}

w(R)=1
w(¬R)=2
w(C)=3
w(¬C)=5

Rain(M)	Cloudy(M)	Rain(T)	Cloudy(T)	Model?	Weight
T	T	T	T	Yes	1 * 1 * 3 * 3 = 9
T	F	T	T	No	0
F	T	T	T	Yes	2 * 1* 3 * 3 = 18
F	F	T	T	Yes	2 * 1 * 5 * 3 = 30
T	T	T	F	No	0
T	F	T	F	No	0
F	T	T	F	No	0
F	F	T	F	No	0
T	T	F	T	Yes	1 * 2 * 3 * 3 = 18
T	F	F	T	No	0
F	T	F	T	Yes	2 * 2 * 3 * 3 = 36
F	F	F	T	Yes	2 * 2 * 5 * 3 = 60
T	T	F	F	Yes	1 * 2 * 3 * 5 = 30
T	F	F	F	No	0
F	T	F	F	Yes	2 * 2 * 3 * 5 = 60
F	F	F	F	Yes	2 * 2 * 5 * 5 = 100

+

#SAT = 9

Weighted First-Order Model Counting

Model = solution to first-order logic formula Δ

Δ = ∀d (Rain(d) ⇒ Cloudy(d))

Days = {Monday **Tuesday**}

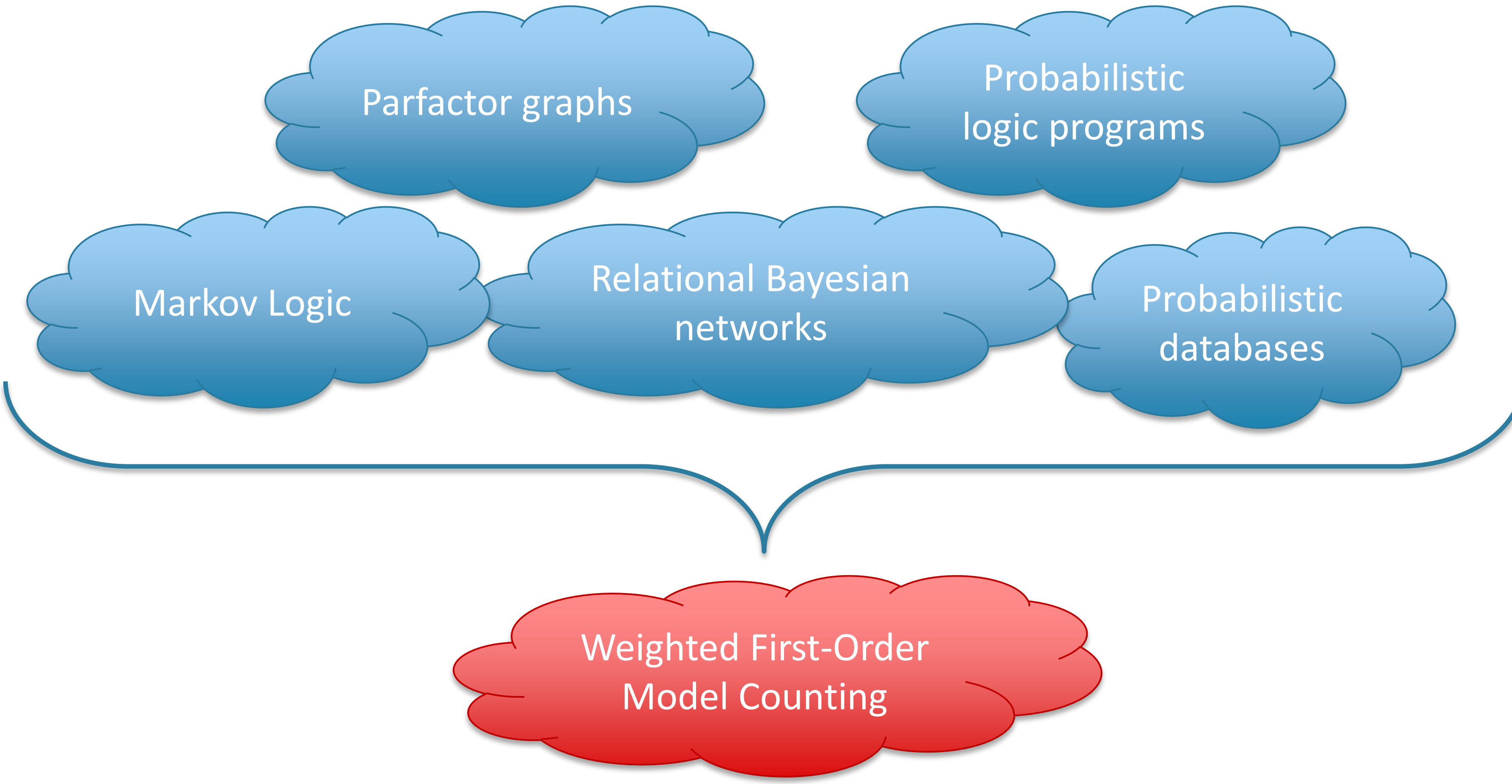
w(R)=1
w(¬R)=2
w(C)=3
w(¬C)=5

Rain(M)	Cloudy(M)	Rain(T)	Cloudy(T)	Model?	Weight
T	T	T	T	Yes	1 * 1 * 3 * 3 = 9
T	F	T	T	No	0
F	T	T	T	Yes	2 * 1* 3 * 3 = 18
F	F	T	T	Yes	2 * 1 * 5 * 3 = 30
T	T	T	F	No	0
T	F	T	F	No	0
F	T	T	F	No	0
F	F	T	F	No	0
T	T	F	T	Yes	1 * 2 * 3 * 3 = 18
T	F	F	T	No	0
F	T	F	T	Yes	2 * 2 * 3 * 3 = 36
F	F	F	T	Yes	2 * 2 * 5 * 3 = 60
T	T	F	F	Yes	1 * 2 * 3 * 5 = 30
T	F	F	F	No	0
F	T	F	F	Yes	2 * 2 * 3 * 5 = 60
F	F	F	F	Yes	2 * 2 * 5 * 5 = 100

+ #SAT = 9

+ WFOMC = 361

Assembly language for high-level probabilistic reasoning



[VdB et al.; IJCAI'11, PhD'13, KR'14, UAI'14]

WFOMC Inference: Example

- FO-Model Counting: $w(R) = w(\neg R) = 1$
- Apply inference rules backwards (step 4-3-2-1)

WFOMC Inference: Example

- FO-Model Counting: $w(R) = w(\neg R) = 1$
- Apply inference rules backwards (step 4-3-2-1)

---

4. Δ = (Stress(Alice) ⇒ Smokes(Alice))

Domain = {Alice}

WFOMC Inference: Example

- FO-Model Counting: $w(R) = w(\neg R) = 1$
- Apply inference rules backwards (step 4-3-2-1)

---

4. $\boxed{\Delta = (\text{Stress(Alice)} \Rightarrow \text{Smokes(Alice)})}$ $\boxed{\text{Domain} = \{\text{Alice}\}}$

→ 3 models

WFOMC Inference: Example

- FO-Model Counting: $w(R) = w(\neg R) = 1$
- Apply inference rules backwards (step 4-3-2-1)

---

4. $\boxed{\Delta = (\text{Stress(Alice)} \Rightarrow \text{Smokes(Alice)})}$ $\boxed{\text{Domain} = \{\text{Alice}\}}$

   → 3 models

---

3. $\boxed{\Delta = \forall x, (\text{Stress}(x) \Rightarrow \text{Smokes}(x))}$ $\boxed{\text{Domain} = \{n \text{ people}\}}$

WFOMC Inference: Example

- FO-Model Counting: $w(R) = w(\neg R) = 1$
- Apply inference rules backwards (step 4-3-2-1)

---

4. $\boxed{\Delta = (\text{Stress}(\text{Alice}) \Rightarrow \text{Smokes}(\text{Alice}))}$ $\boxed{\text{Domain} = \{\text{Alice}\}}$

→ 3 models

---

3. $\boxed{\Delta = \forall x, (\text{Stress}(x) \Rightarrow \text{Smokes}(x))}$ $\boxed{\text{Domain} = \{n \text{ people}\}}$

→ 3^n models

WFOMC Inference: Example

3. $\Delta = \forall x, (\text{Stress}(x) \Rightarrow \text{Smokes}(x))$

Domain = {n people}

$\rightarrow 3^n$ models

WFOMC Inference: Example

3. $\Delta = \forall x, (Stress(x) \Rightarrow Smokes(x))$

Domain = {n people}

→ 3^n models

2. $\Delta = \forall y, (ParentOf(y) \land Female \Rightarrow MotherOf(y))$

D = {n people}

WFOMC Inference: Example

3. $\Delta = \forall x, (Stress(x) \Rightarrow Smokes(x))$

Domain = {n people}

$\rightarrow 3^n$ models

2. $\Delta = \forall y, (ParentOf(y) \land Female \Rightarrow MotherOf(y))$

D = {n people}

If Female = true?

$\Delta = \forall y, (ParentOf(y) \Rightarrow MotherOf(y))$

$\rightarrow 3^n$ models

WFOMC Inference: Example

3. $\Delta = \forall x, (Stress(x) \Rightarrow Smokes(x))$

Domain = {n people}

$\rightarrow 3^n$ models

2. $\Delta = \forall y, (ParentOf(y) \wedge Female \Rightarrow MotherOf(y))$

D = {n people}

If Female = true? $\Delta = \forall y, (ParentOf(y) \Rightarrow MotherOf(y))$ $\rightarrow 3^n$ models

If Female = false? $\Delta = true$ $\rightarrow 4^n$ models

WFOMC Inference: Example

3. $\Delta = \forall x, (\text{Stress}(x) \Rightarrow \text{Smokes}(x))$

Domain = {n people}

$\rightarrow 3^n$ models

2. $\Delta = \forall y, (\text{ParentOf}(y) \land \text{Female} \Rightarrow \text{MotherOf}(y))$

D = {n people}

If Female = true? $\Delta = \forall y, (\text{ParentOf}(y) \Rightarrow \text{MotherOf}(y))$ $\rightarrow 3^n$ models

If Female = false? $\Delta = \text{true}$ $\rightarrow 4^n$ models

$\rightarrow 3^n + 4^n$ models

WFOMC Inference: Example

3. $\Delta = \forall x, (Stress(x) \Rightarrow Smokes(x))$

Domain = {n people}

$\rightarrow 3^n$ models

2. $\Delta = \forall y, (ParentOf(y) \wedge Female \Rightarrow MotherOf(y))$

D = {n people}

If Female = true? $\Delta = \forall y, (ParentOf(y) \Rightarrow MotherOf(y))$ $\rightarrow 3^n$ models

If Female = false? $\Delta = true$ $\rightarrow 4^n$ models

$\rightarrow 3^n + 4^n$ models

1. $\Delta = \forall x,y, (ParentOf(x,y) \wedge Female(x) \Rightarrow MotherOf(x,y))$

D = {n people}

WFOMC Inference: Example

3. $\Delta = \forall x, (Stress(x) \Rightarrow Smokes(x))$

Domain = {n people}

$\rightarrow 3^n$ models

2. $\Delta = \forall y, (ParentOf(y) \land Female \Rightarrow MotherOf(y))$

D = {n people}

If Female = true? $\Delta = \forall y, (ParentOf(y) \Rightarrow MotherOf(y))$ $\rightarrow 3^n$ models

If Female = false? $\Delta = true$ $\rightarrow 4^n$ models

$\rightarrow 3^n + 4^n$ models

1. $\Delta = \forall x,y, (ParentOf(x,y) \land Female(x) \Rightarrow MotherOf(x,y))$

D = {n people}

$\rightarrow (3^n + 4^n)^n$ models

Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

Domain = {n people}

Atom Counting: Example

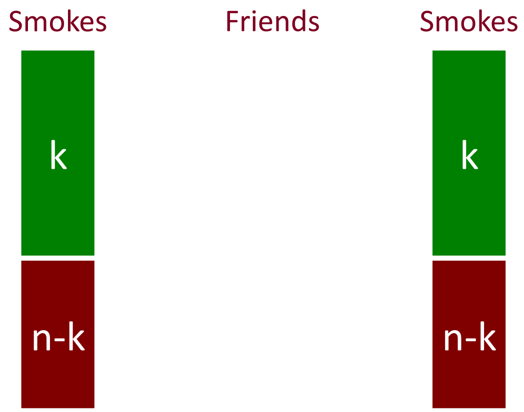
$\Delta = \forall x,y, (\text{Smokes}(x) \wedge \text{Friends}(x,y) \Rightarrow \text{Smokes}(y))$

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

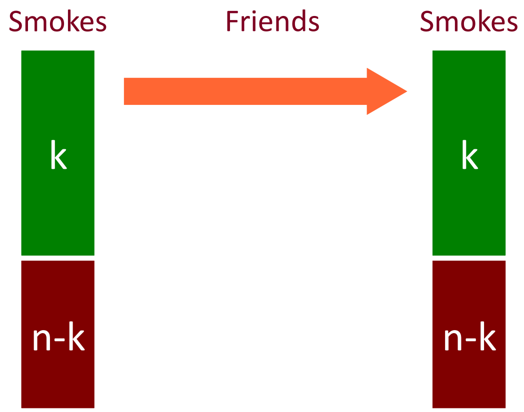
$\Delta = \forall x,y, (\text{Smokes}(x) \wedge \text{Friends}(x,y) \Rightarrow \text{Smokes}(y))$

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

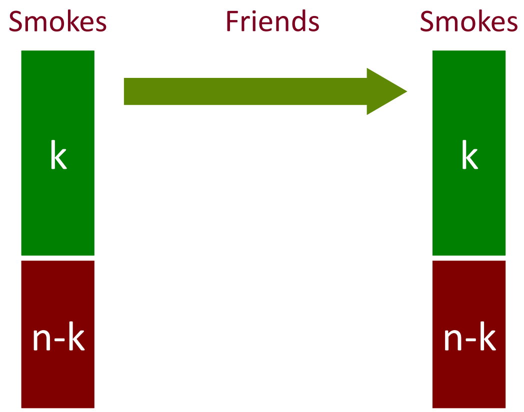
$\color{red}{\Delta}$ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

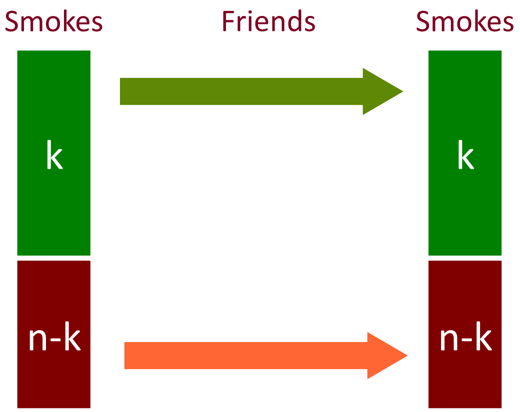
$\Delta = \forall x,y, (Smokes(x) \wedge Friends(x,y) \Rightarrow Smokes(y))$

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

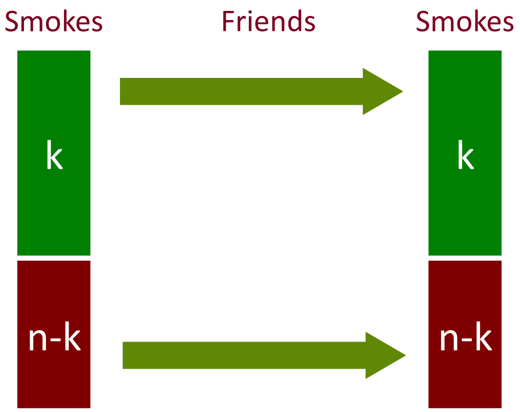
Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

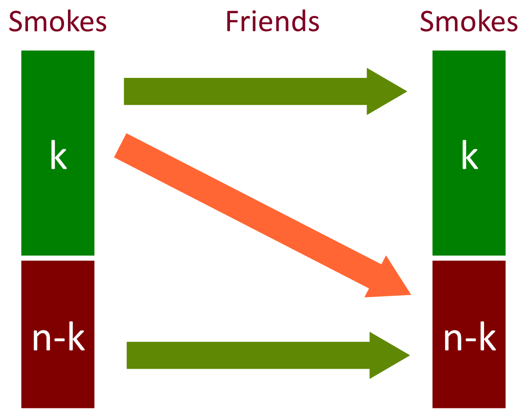
$\Delta = \forall x,y, (\text{Smokes}(x) \land \text{Friends}(x,y) \Rightarrow \text{Smokes}(y))$

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

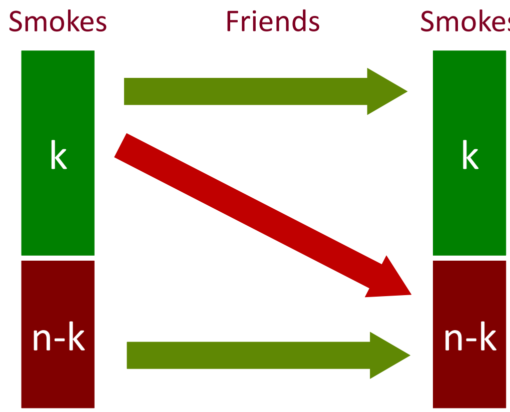
Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

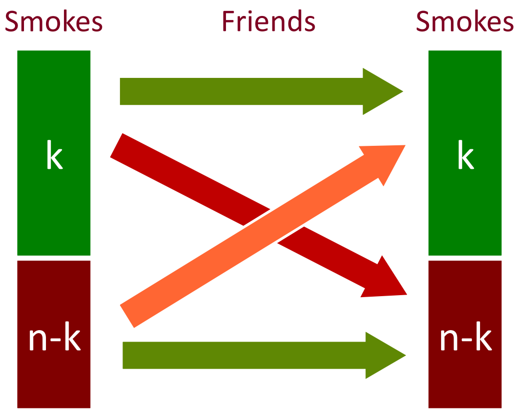
$\Delta = \forall x,y, (\text{Smokes}(x) \wedge \text{Friends}(x,y) \Rightarrow \text{Smokes}(y))$

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

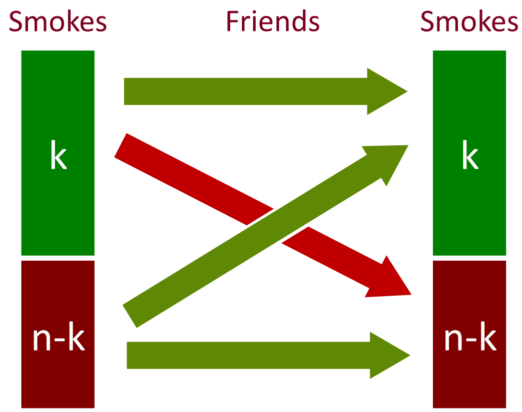
Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...



Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

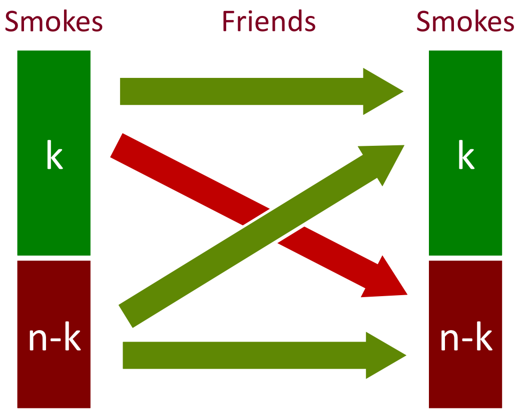
Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

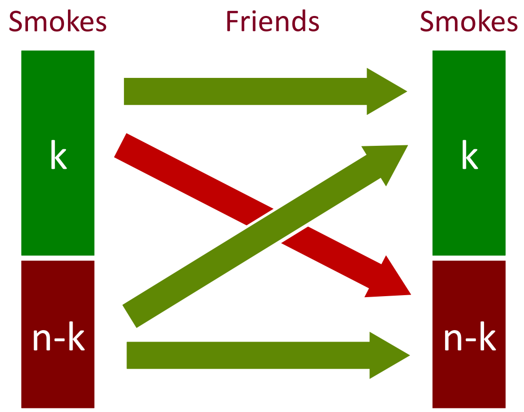
Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2-k(n-k)}$ models



- If we know that there are *k* smokers?

Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

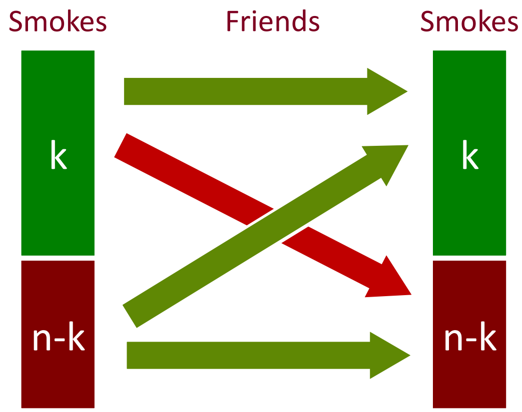
Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

→ $2^{n^2 - k(n-k)}$ models



- If we know that there are *k* smokers?

→ $\binom{n}{k} 2^{n^2 - k(n-k)}$ models

Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

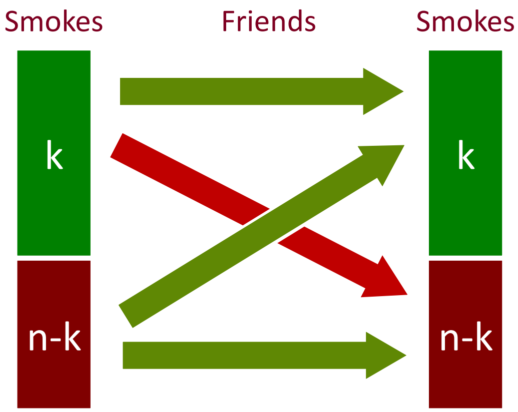
Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

$\rightarrow 2^{n^2-k(n-k)}$ models



- If we know that there are *k* smokers? $\rightarrow \binom{n}{k} 2^{n^2-k(n-k)}$ models

- In total…

Atom Counting: Example

Δ = ∀x,y, (Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y))

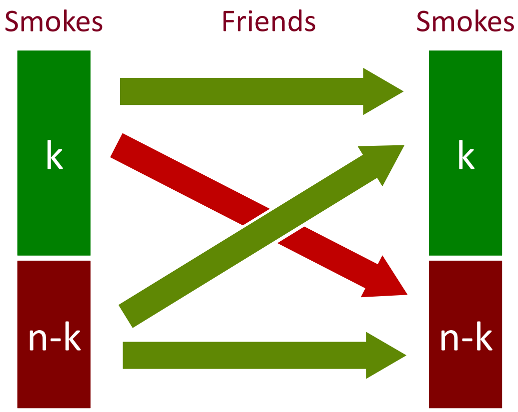
Domain = {n people}

- If we know precisely who smokes, and there are *k* smokers?

Database:

Smokes(Alice) = 1
Smokes(Bob) = 0
Smokes(Charlie) = 0
Smokes(Dave) = 1
Smokes(Eve) = 0
...

$\rightarrow$ $2^{n^2 - k(n-k)}$ models



- If we know that there are *k* smokers? $\rightarrow$ $\binom{n}{k} 2^{n^2 - k(n-k)}$ models

- In total... $\rightarrow$ $\sum_{k=0}^{n} \binom{n}{k} 2^{n^2 - k(n-k)}$ models

First-Order Knowledge Compilation

Markov Logic

3.14	Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

[Van den Broeck et al.; IJCAI'11, NIPS'11, PhD'13, KR'14]

First-Order Knowledge Compilation

Markov Logic: 3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

[Van den Broeck et al.; IJCAI'11, NIPS'11, PhD'13, KR'14]

First-Order Knowledge Compilation

Markov Logic: **3.14** Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

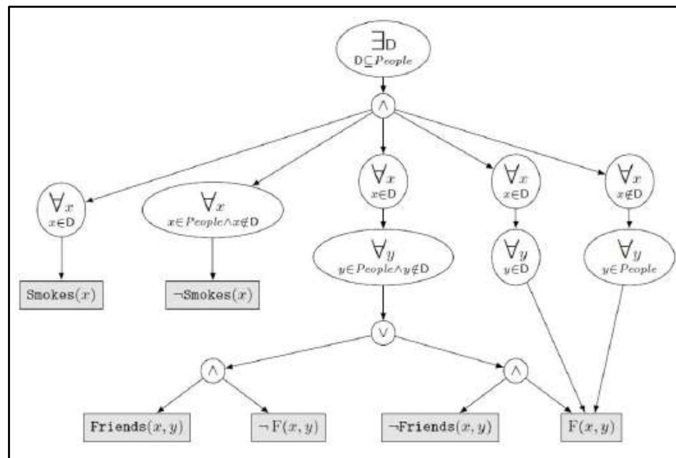
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile

First-Order d-DNNF Circuit



[Van den Broeck et al.; IJCAI'11, NIPS'11, PhD'13, KR'14]

First-Order Knowledge Compilation

Markov Logic: 3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

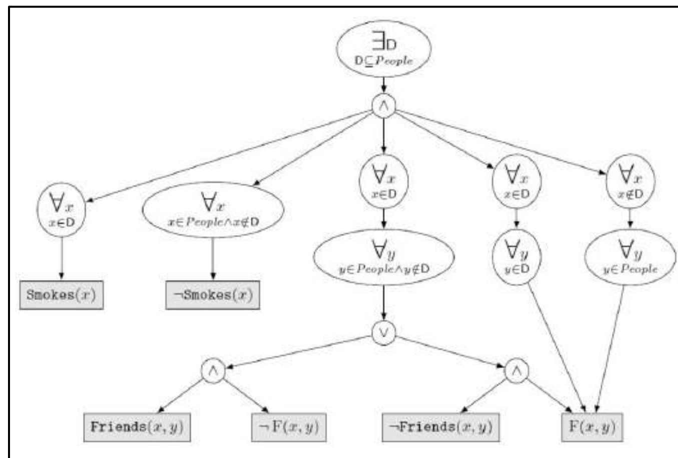
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

[Van den Broeck et al.; IJCAI'11, NIPS'11, PhD'13, KR'14]

First-Order Knowledge Compilation

Markov Logic | 3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

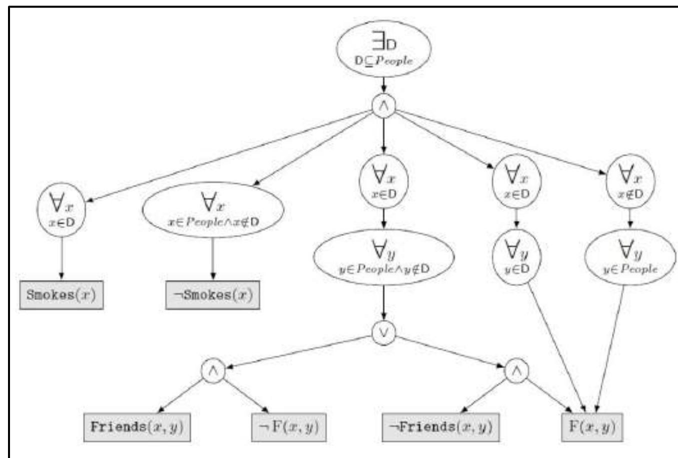
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile

First-Order d-DNNF Circuit

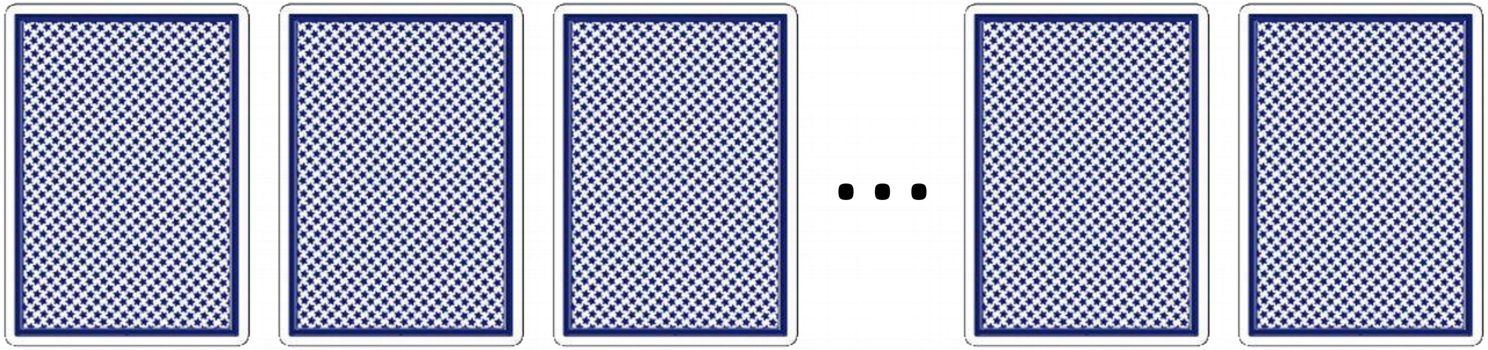


Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

[Van den Broeck et al.; IJCAI’11, NIPS’11, PhD’13, KR’14]



Let us automate this:

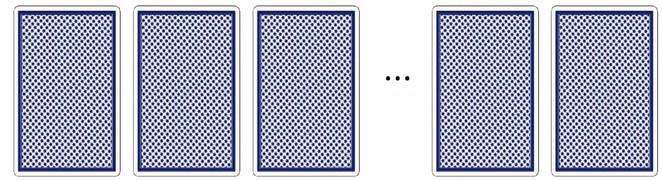
– **Relational** model

$$\forall p, \exists c, \mathrm{Card}(p,c)$$
$$\forall c, \exists p, \mathrm{Card}(p,c)$$
$$\forall p, \forall c, \forall c', \mathrm{Card}(p,c) \wedge \mathrm{Card}(p,c') \Rightarrow c = c'$$

– **Lifted** probabilistic inference algorithm

Playing Cards Revisited

Let us automate this:

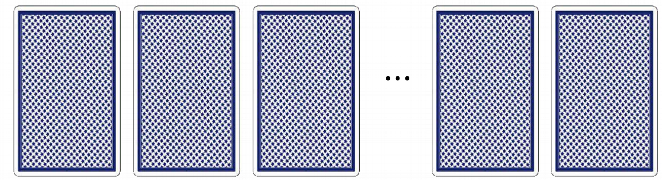


$$\forall p, \exists c, Card(p,c)$$
$$\forall c, \exists p, Card(p,c)$$
$$\forall p, \forall c, \forall c', Card(p,c) \land Card(p,c') \Rightarrow c = c'$$

[Van den Broeck.; AAAI-KR'15]

Playing Cards Revisited

Let us automate this:



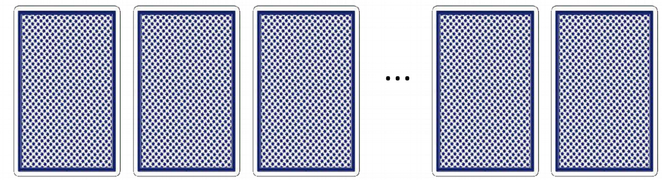
∀p, ∃c, Card(p,c)
∀c, ∃p, Card(p,c)
∀p, ∀c, ∀c’, Card(p,c) ∧ Card(p,c’) ⇒ c = c’

$$\#SAT = \sum_{k=0}^{n} \binom{n}{k} \sum_{l=0}^{n} \binom{n}{l} (l+1)^k (-1)^{2n-k-l} = n!$$

[Van den Broeck.; AAAI-KR’15]

Playing Cards Revisited

Let us automate this:



$\forall p, \exists c, \text{Card}(p,c)$

$\forall c, \exists p, \text{Card}(p,c)$

$\forall p, \forall c, \forall c', \text{Card}(p,c) \wedge \text{Card}(p,c') \Rightarrow c = c'$

$$\#SAT = \sum_{k=0}^{n} \binom{n}{k} \sum_{l=0}^{n} \binom{n}{l} (l+1)^k (-1)^{2n-k-l} = n!$$

Computed in time polynomial in n

[Van den Broeck.; AAAI-KR'15]

Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- Intuition: Inference rules
- **Liftability theory: Strengths and limitations**
- Lifting in practice
  - Approximate symmetries
  - Lifted learning

Theory of Inference

Goal:
Understand complexity of probabilistic reasoning

- Low-level graph-based concepts (treewidth)
  - ⇒ inadequate to describe high-level reasoning
- Need to develop "liftability theory"
- Deep connections to
  - – database theory, finite model theory, 0-1 laws,
  - – counting complexity

[Van den Broeck.; NIPS'11], [Van den Broeck, Jaeger.; StarAI'12]

Lifted Inference: Definition

- Informal [Poole'03, etc.]

  Exploit symmetries, Reason at first-order level, Reason about groups of objects, Scalable inference, High-level probabilistic reasoning, etc.

- A formal definition: **Domain-lifted inference**

  Inference runs in time **polynomial**
  in the number of entities in the **domain**.

[Van den Broeck.; NIPS'11]

Lifted Inference: Definition

- Informal [Poole'03, etc.]

  Exploit symmetries, Reason at first-order level, Reason about groups of objects, Scalable inference, High-level probabilistic reasoning, etc.

- A formal definition: **Domain-lifted inference**

  > Inference runs in time **polynomial**
  > in the number of entities in the **domain**.

  - Polynomial in #rows, #entities, #people, #webpages, #cards
  - ~ data complexity in databases

[Van den Broeck.; NIPS'11]

Lifted Inference: Definition

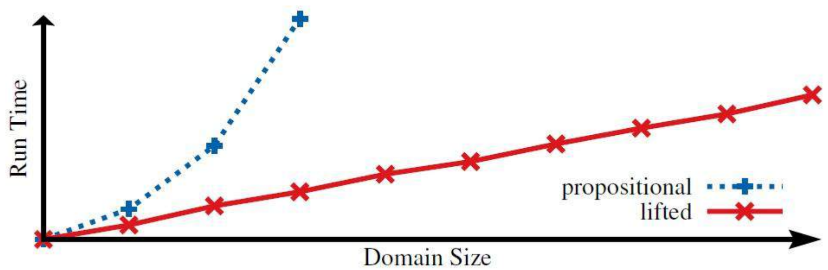
- Informal [Poole'03, etc.]

  Exploit symmetries, Reason at first-order level, Reason about groups of objects, Scalable inference, High-level probabilistic reasoning, etc.

- A formal definition: **Domain-lifted inference**

  Inference runs in time **polynomial** in the number of entities in the **domain.**

  – Polynomial in #rows, #entities, #people, #webpages, #cards
  – ~ data complexity in databases



[Van den Broeck.; NIPS'11]

Lifted Inference: Definition

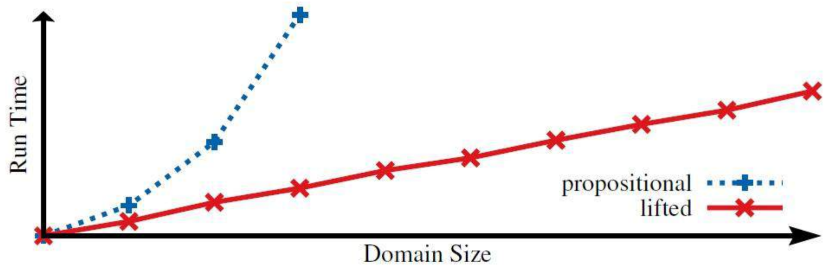
- Informal [Poole'03, etc.]

  Exploit symmetries, Reason at first-order level, Reason about groups of objects, Scalable inference, High-level probabilistic reasoning, etc.

- A formal definition: **Domain-lifted inference**

  Inference runs in time **polynomial** in the number of entities in the **domain.**

  – Polynomial in #rows, #entities, #people, #webpages, #cards
  – ~ data complexity in databases



Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0

↓ Big data

[Van den Broeck.; NIPS'11]

First-Order Knowledge Compilation

Markov Logic: 3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

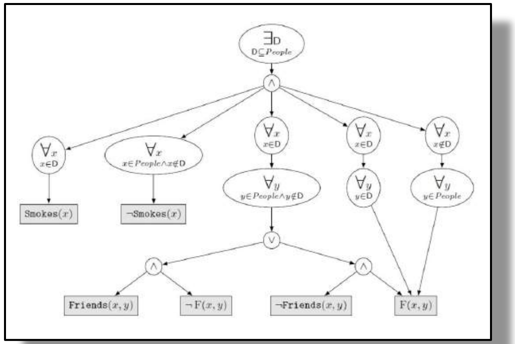
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

[Van den Broeck.; NIPS'11]

First-Order Knowledge Compilation

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

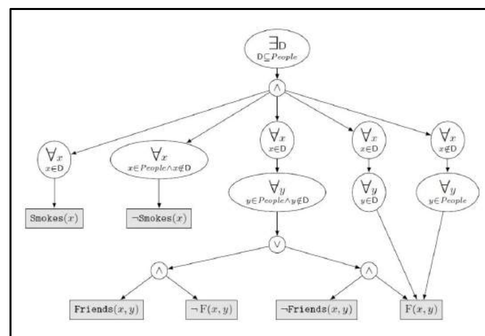
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

Evaluation in time polynomial in domain size

[Van den Broeck.; NIPS'11]

First-Order Knowledge Compilation

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

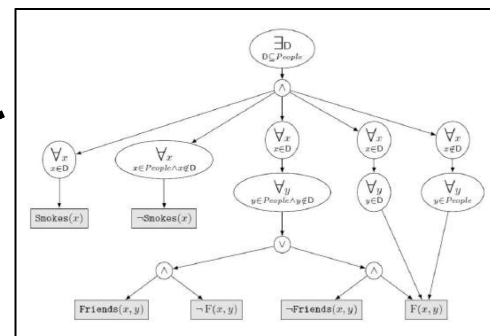
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

Evaluation in time polynomial in domain size

Domain-lifted!

[Van den Broeck.; NIPS’11]

First-Order Knowledge Compilation

Markov Logic

3.14 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)

Weight Function

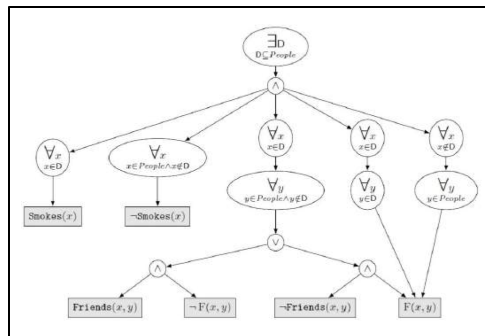
w(Smokes)=1
w(¬Smokes)=1
w(Friends)=1
w(¬Friends)=1
w(F)=3.14
w(¬F)=1

FOL Sentence

∀x,y, F(x,y) ⇔ [Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)]

Compile?

First-Order d-DNNF Circuit



Domain

Alice
Bob
Charlie

Z = WFOMC = 1479.85

Evaluation in time polynomial in domain size

Domain-lifted!

[Van den Broeck.; NIPS'11]

What Can Be Lifted?

Theorem: WFOMC for FO^2 is liftable

[Van den Broeck.; NIPS’11], [Van den Broeck et al.; KR’14]

What Can Be Lifted?

Theorem: WFOMC for FO^2 is liftable

Corollary: Markov logic with two logical variables per formula is liftable.

[Van den Broeck.; NIPS'11], [Van den Broeck et al.; KR'14]

What Can Be Lifted?

Theorem: WFOMC for FO^2 is liftable

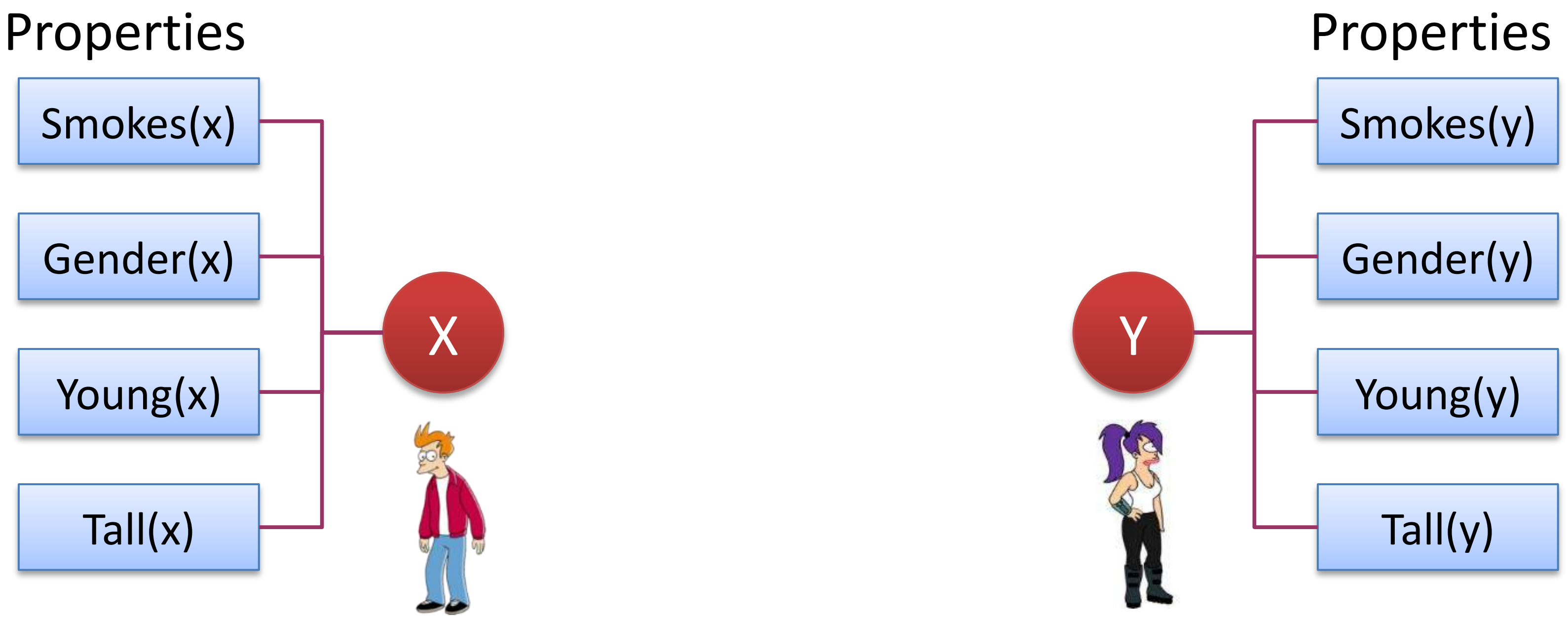
Corollary: Markov logic with two logical variables per formula is liftable.

Corollary: Tight probabilistic logic programs with two logical variables are liftable.

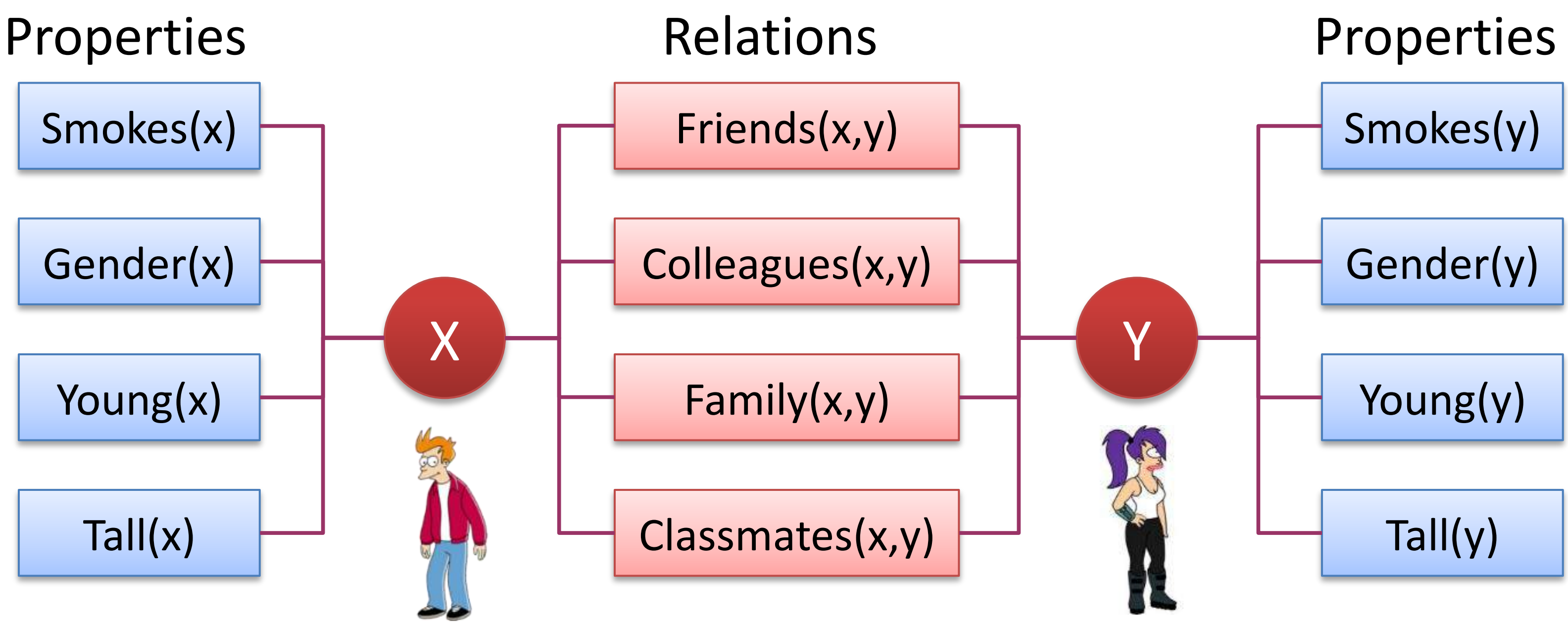
…

[Van den Broeck.; NIPS'11], [Van den Broeck et al.; KR'14]

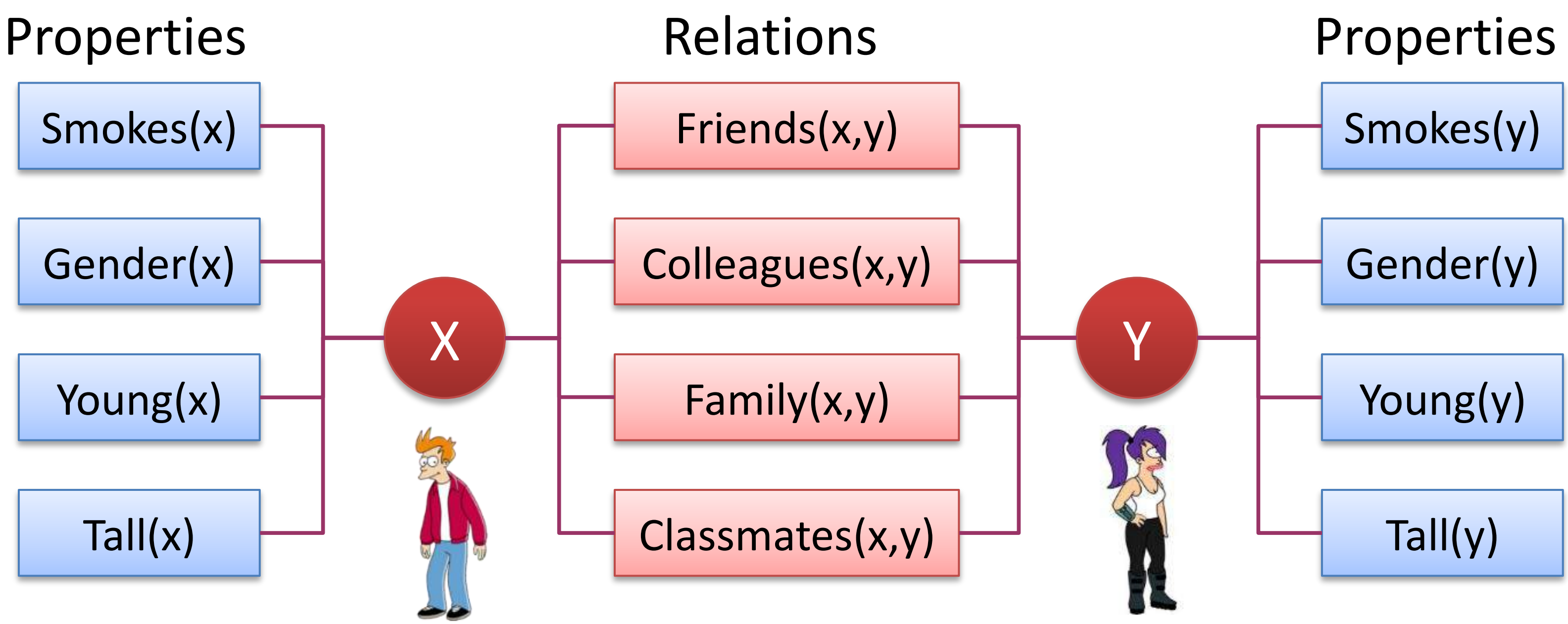
FO^2 is liftable!



FO^2 is liftable!



FO^2 is liftable!



"Smokers are more likely to be friends with other smokers."
"Colleagues of the same age are more likely to be friends."
"People are either family or friends, but never both."
"If X is family of Y, then Y is also family of X."
"If X is a parent of Y, then Y cannot be a parent of X."

FO^2 is liftable!

Medical Records

Statistical Relational Model in FO^2

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Frank	1	?	?

Friends

Brothers

$2.1 \quad \text{Asthma}(x) \Rightarrow \text{Cough}(x)$

$3.5 \quad \text{Smokes}(x) \Rightarrow \text{Cough}(x)$

$1.9 \quad \text{Smokes}(x) \wedge \text{Friends}(x,y) \Rightarrow \text{Smokes}(y)$

$1.5 \quad \text{Asthma}(x) \wedge \text{Family}(x,y) \Rightarrow \text{Asthma}(y)$

Frank	1	0.2	0.6

[Van den Broeck.; NIPS'11], [Van den Broeck et al.; KR'14]

FO^2 is liftable!

Medical Records

Statistical Relational Model in FO^2

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0
Dave	1	0	1
Eve	1	0	0

Friends

Brothers

Frank	1	?	?

Big data

2.1 Asthma(x) ⇒ Cough(x)
3.5 Smokes(x) ⇒ Cough(x)
1.9 Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)
1.5 Asthma (x) ∧ Family(x,y) ⇒ Asthma (y)

Frank	1	0.2	0.6

[Van den Broeck.; NIPS'11], [Van den Broeck et al.; KR'14]

Can Everything Be Lifted?

[Beame, Van den Broeck, Gribkoff, Suciu; PODS'15]

Can Everything Be Lifted?

Theorem: There exists an FO^3 sentence Θ_1 for which first-order model counting is $\#P_1$-complete in the domain size.

[Beame, Van den Broeck, Gribkoff, Suciu; PODS'15]

Can Everything Be Lifted?

Theorem: There exists an FO^3 sentence Θ_1 for which first-order model counting is $\#P_1$-complete in the domain size.

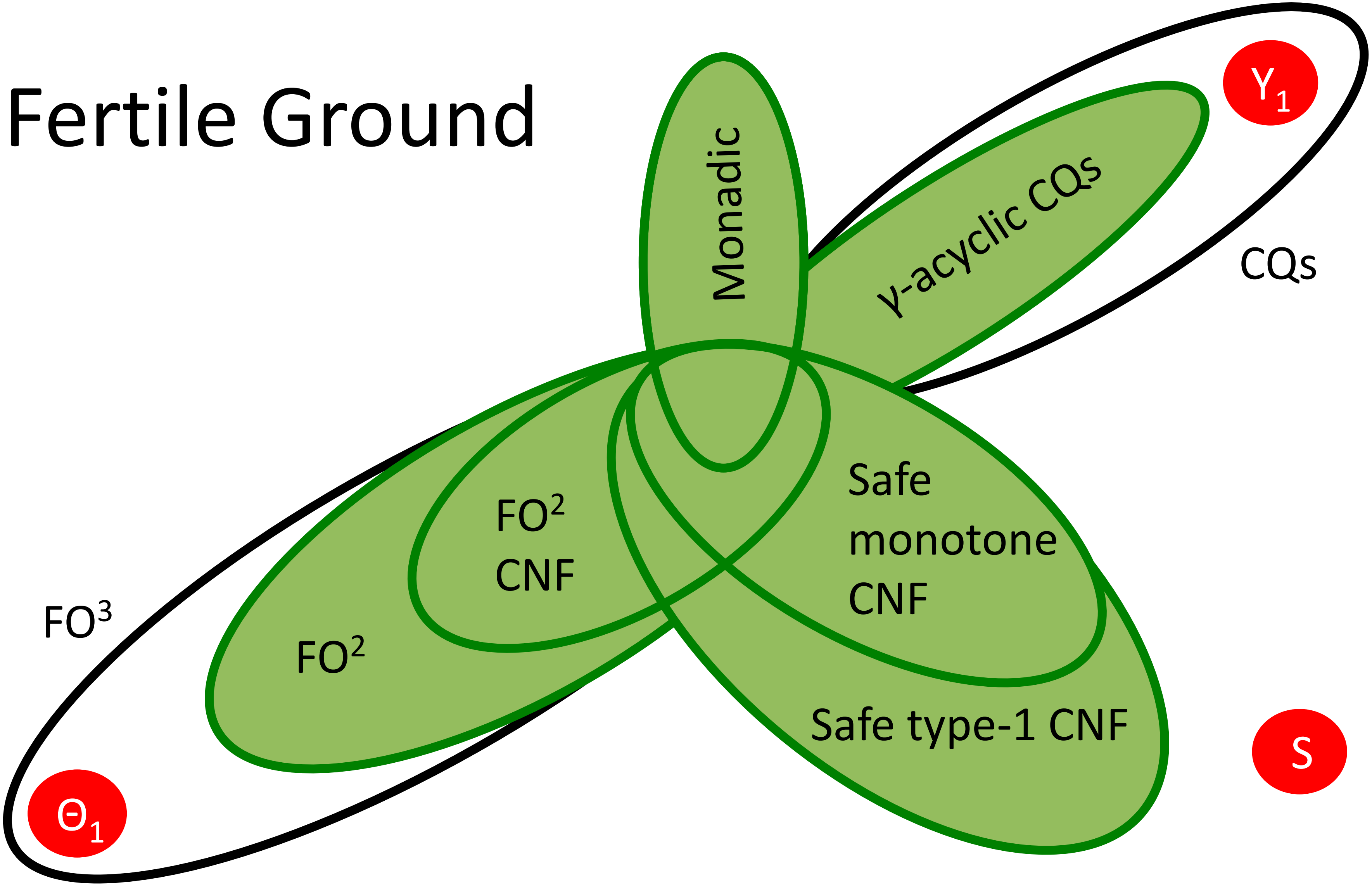
A counting Turing machine is a nondeterministic TM that prints the number of its accepting computations.

The class $\#P_1$ consists of all functions computed by a polynomial-time counting TM with unary input alphabet.

<u>Proof:</u> Encode a universal $\#P_1$-TM in FO^3

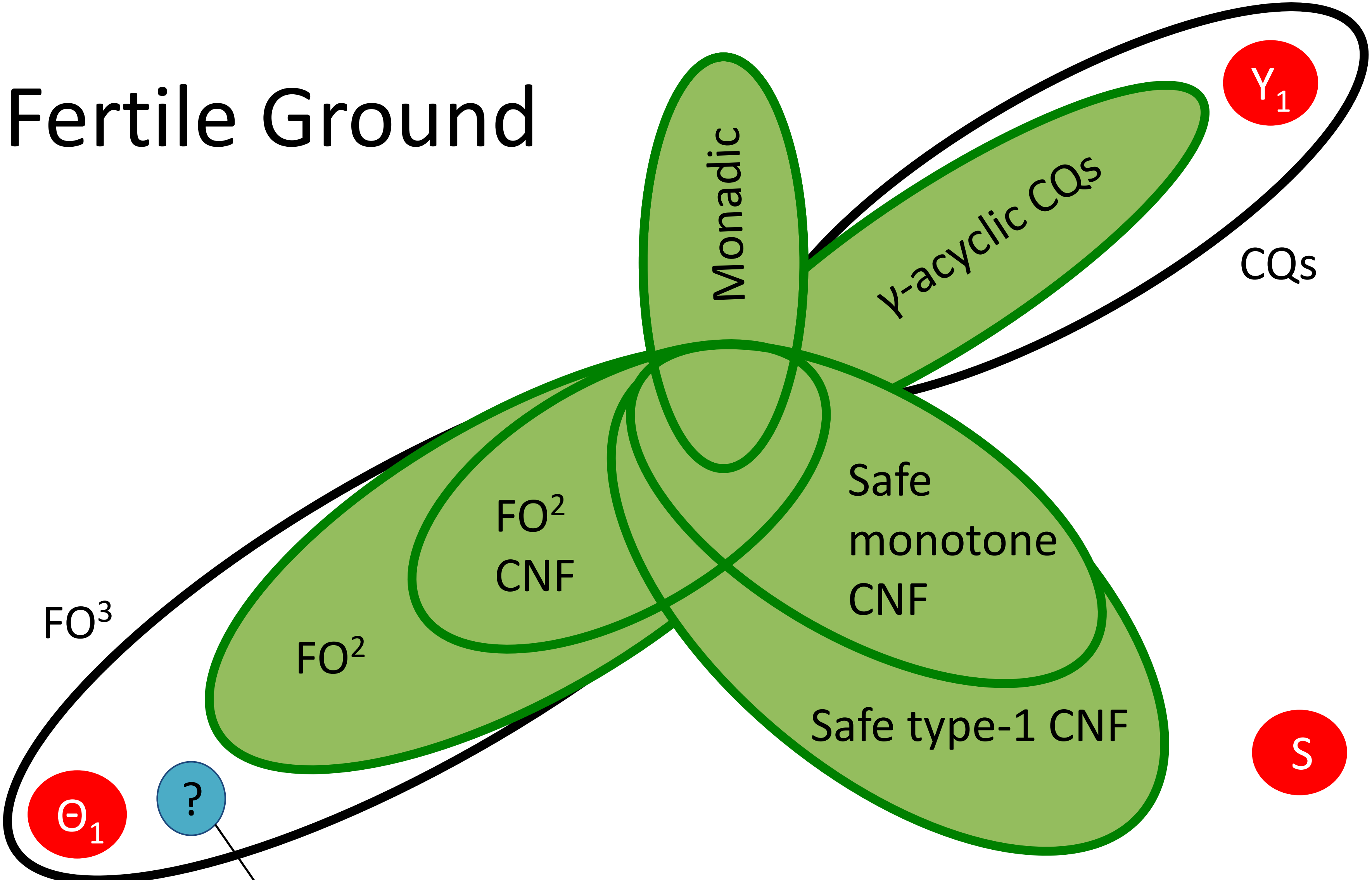
[Beame, Van den Broeck, Gribkoff, Suciu; PODS'15]

Fertile Ground



[VdB; NIPS'11], [VdB et al.; KR'14], [Gribkoff, VdB, Suciu; UAI'15], [Beame, VdB, Gribkoff, Suciu; PODS'15], etc.

Fertile Ground



$\Delta = \forall x,y,z,\ \text{Friends}(x,y) \wedge \text{Friends}(y,z) \Rightarrow \text{Friends}(x,z)$

[VdB; NIPS'11], [VdB et al.; KR'14], [Gribkoff, VdB, Suciu; UAI'15], [Beame, VdB, Gribkoff, Suciu; PODS'15], etc.

Statistical Properties

1. Independence

P(

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0

) = P(

Alice	1	1	0

) x P(

Bob	0	0	0

) x P(

Charlie	0	1	0

)

2. Partial Exchangeability

P(

Name	Cough	Asthma	Smokes
Alice	1	1	0
Bob	0	0	0
Charlie	0	1	0

) = P(

Name	Cough	Asthma	Smokes
Charlie	1	1	0
Alice	0	0	0
Bob	0	1	0

)

3. Independent and identically distributed (i.i.d.)

   = Independence + Partial Exchangeability

Statistical Properties for Tractability

- Tractable classes independent of representation
- Traditionally:
  - Tractable learning from **i.i.d. data**
  - Tractable inference when **cond. independence**
- New understanding:
  - Tractable learning from **exchangeable data**
  - Tractable inference when
    - **Conditional independence**
    - **Conditional exchangeability**
    - **A combination**

[Niepert, Van den Broeck; AAAI'14]

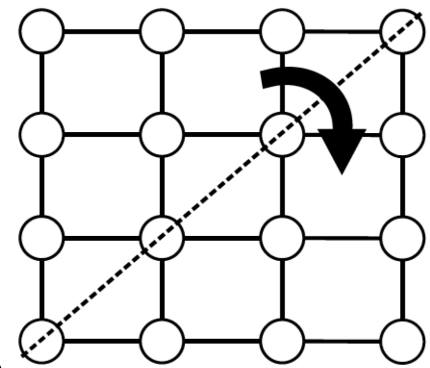
Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - **Approximate symmetries**
  - Lifted learning

Approximate Symmetries

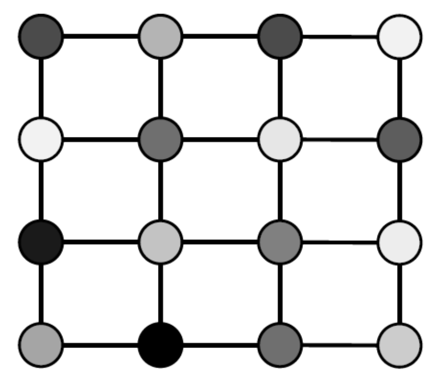
- What if not liftable? Asymmetric graph?
- Exploit approximate symmetries:
  - Exact symmetry g: Pr(**x**) = Pr(**x**^{g})

    E.g. Ising model
    without external field



  - Approximate symmetry g: Pr(**x**) ≈ Pr(**x**^{g})

    E.g. Ising model with external field



P  ≈ P 

[Van den Broeck, Darwiche; NIPS'13], [Van den Broeck, Niepert; AAAI'15]

Example: Statistical Relational Model

- WebKB: Classify pages given links and words
- Very large Markov logic network

$$1.3 \quad \texttt{Page}(\mathsf{x}, \mathsf{Faculty}) \Rightarrow \texttt{HasWord}(\mathsf{x}, \mathsf{Hours})$$

$$1.5 \quad \texttt{Page}(\mathsf{x}, \mathsf{Faculty}) \wedge \texttt{Link}(\mathsf{x}, \mathsf{y}) \Rightarrow \texttt{Page}(\mathsf{y}, \mathsf{Course})$$

and 5000 more …

- No symmetries with evidence on Link or Word
- Where do approx. symmetries come from?

[Van den Broeck, Darwiche; NIPS'13], [Van den Broeck, Niepert; AAAI'15]

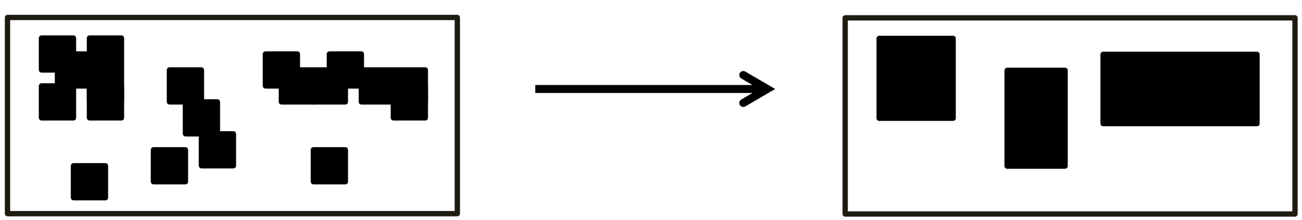
Over-Symmetric Approximations

- OSA makes model more symmetric
- E.g., low-rank Boolean matrix factorization

Link (“aaai.org”, “google.com”)
Link (“google.com”, “aaai.org”)
Link (“google.com”, “gmail.com”)
Link (“ibm.com”, “aaai.org”)

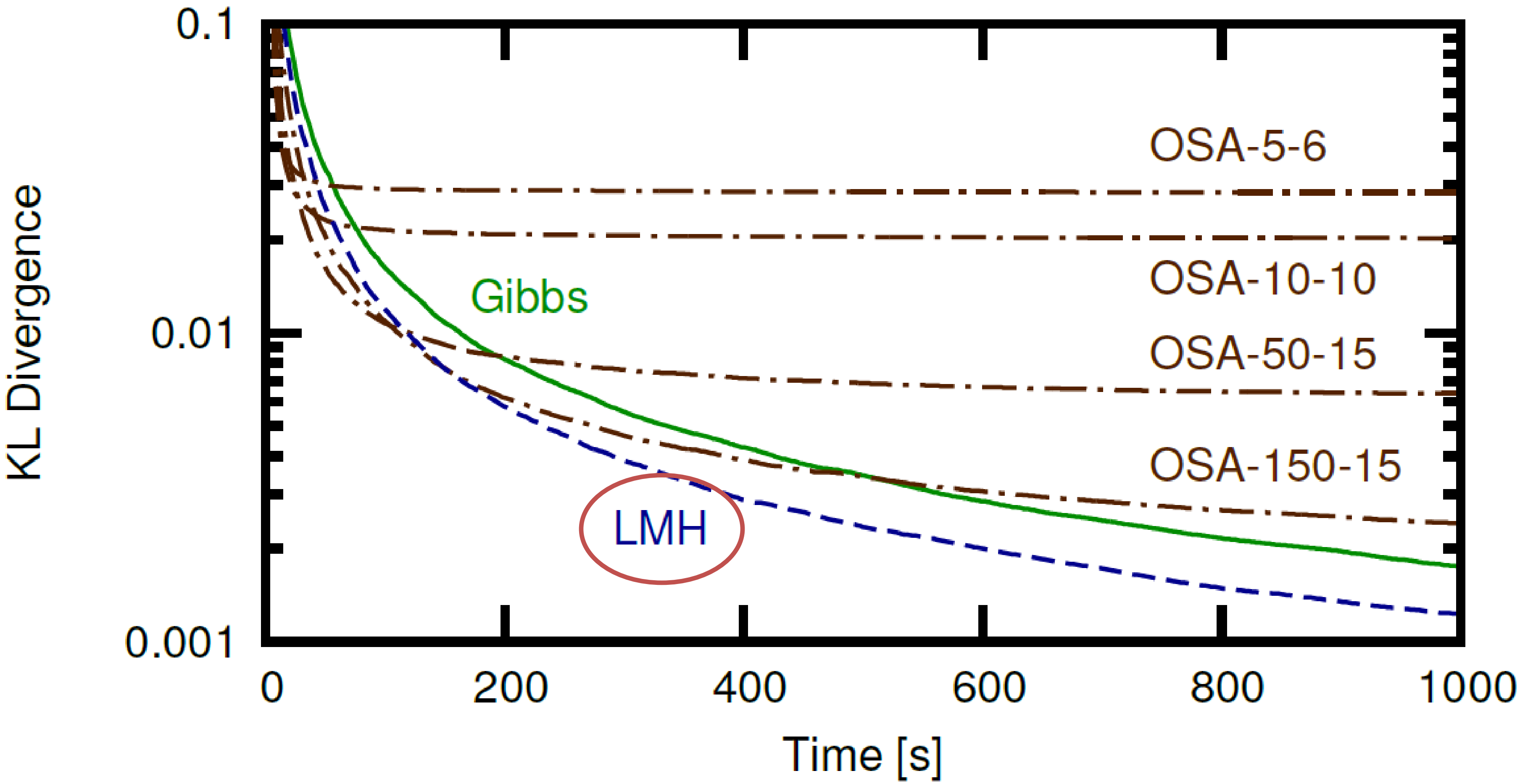
Link (“aaai.org”, “google.com”)
Link (“google.com”, “aaai.org”)
- ~~Link (“google.com”, “gmail.com”)~~
+ Link (“aaai.org”, “ibm.com”)
Link (“ibm.com”, “aaai.org”)

google.com and ibm.com become symmetric!



[Van den Broeck, Darwiche; NIPS’13]

Experiments: WebKB



[Van den Broeck, Niepert; AAAI’15]

Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Approximate symmetries
  - **Lifted learning**

Lifted Weight Learning

- **Given:** A set of first-order logic **formulas**

  $\boxed{w \ \text{FacultyPage}(x) \wedge \text{Linked}(x,y) \Rightarrow \text{CoursePage}(y)}$

  A set of training **databases**

- **Learn:** The associated maximum-likelihood **weights**

$$\frac{\partial}{\partial w_j} \log \Pr_w(db) = n_j(db) - \mathbb{E}_w[n_j]$$

Count in databases
Efficient

Expected counts
Requires inference

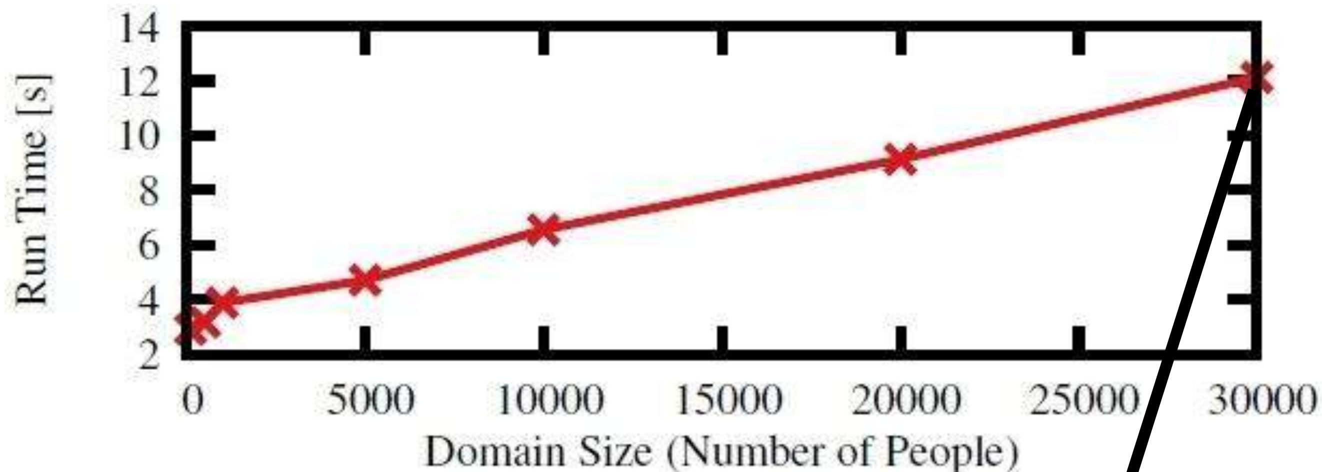
$$\mathbb{E}_w[n_F] = \Pr(F\theta_1) + \cdots + \Pr(F\theta_m)$$

- **Idea:** Lift the computation of $\mathbb{E}_w[n_j]$

[Van den Broeck et al.; StarAI'13]

Learning Time

w Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)



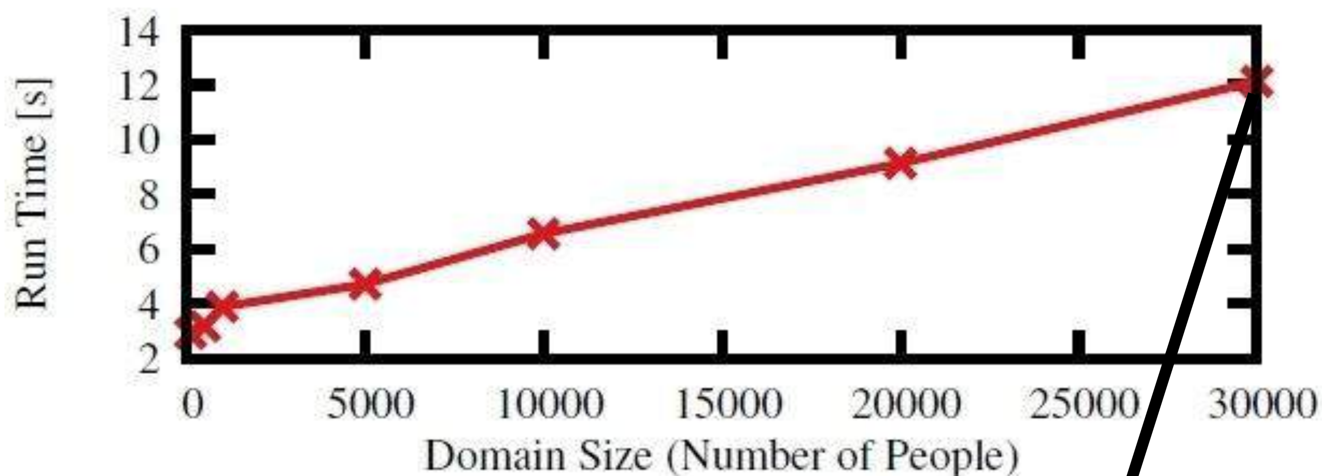
Big data

Learns a model over
900,030,000 random variables

[Van den Broeck et al.; StarAI'13]

Learning Time

w Smokes(x) ∧ Friends(x,y) ⇒ Smokes(y)



~~Big data~~

Big models

Learns a model over
900,030,000 random variables

[Van den Broeck et al.; StarAI'13]

Lifted Structure Learning

- **Given:** A set of training **databases**
- **Learn:** A set of first-order logic **formulas**
  The associated maximum likelihood **weights**
- **Idea:** Learn liftable models (regularize with symmetry)

	IMDb			*UWCSE*		
	Baseline	Lifted Weight Learning	Lifted Structure Learning	Baseline	Lifted Weight Learning	Lifted Structure Learning
Fold 1	-548	-378	**-306**	-1,860	-1,524	**-1,477**
Fold 2	-689	-390	**-309**	-594	-535	**-511**
Fold 3	-1,157	-851	**-733**	-1,462	-1,245	**-1,167**
Fold 4	-415	-285	**-224**	-2,820	-2,510	**-2,442**
Fold 5	-413	-267	**-216**	-2,763	-2,357	**-2,227**

[VHaaren, Van den Broeck, et al.;'15]

Outline

- Motivation
  - Why high-level representations?
  - Why high-level reasoning?
- Intuition: Inference rules
- Liftability theory: Strengths and limitations
- Lifting in practice
  - Lifted learning
  - Approximate symmetries

Conclusions

- A radically new reasoning paradigm
- Lifted inference is **frontier** and **integration** of AI, KR, ML, DBs, theory, etc.
- We need
  - relational databases and logic
  - probabilistic models and statistical learning
  - algorithms that scale
- Many theoretical open problems
- It works in practice

Long-Term Outlook

Probabilistic inference and learning exploit

~ 1988: conditional independence

~ 2000: contextual independence (local structure)

Long-Term Outlook

Probabilistic inference and learning exploit

- ~ 1988: conditional independence
- ~ 2000: contextual independence (local structure)
- ~ 201?: **symmetry & exchangeability**

Collaborators

KU Leuven	
Luc De Raedt	Siegfried Nijssen
Wannes Meert	Jessa Bekker
Jesse Davis	Ingo Thon
Hendrik Blockeel	Bernd Gutmann
Daan Fierens	Vaishak Belle
Angelika Kimmig	Joris Renkens
Nima Taghipour	Davide Nitti
Kurt Driessens	Bart Bogaerts
Jan Ramon	Jonas Vlasselaer
Maurice Bruynooghe	Jan Van Haaren

UCLA
Adnan Darwiche
Arthur Choi
Doga Kisa
Karthika Mohan
Judea Pearl

Univ. Washington
Mathias Niepert
Dan Suciu
Eric Gribkoff
Paul Beame

Indiana Univ.
Sriraam Natarajan

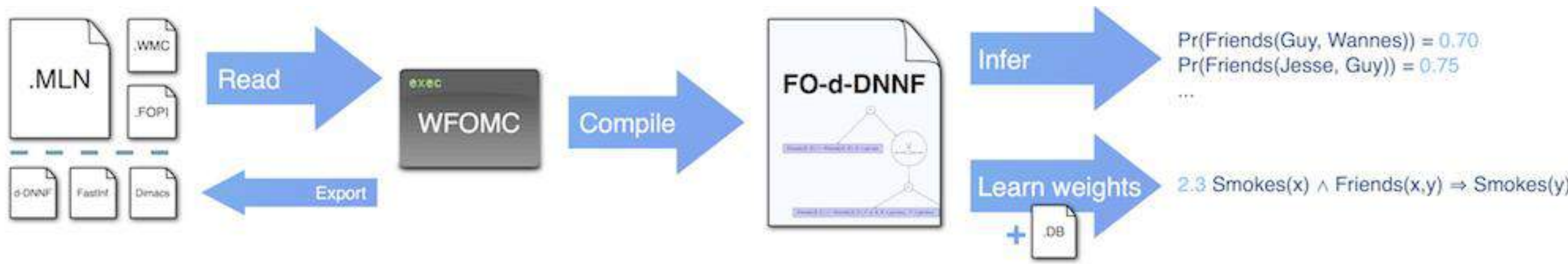
UBC
David Poole

Univ. Dortmund
Kristian Kersting

Aalborg Univ.
Manfred Jaeger

Trento Univ.
Andrea Passerini

Prototype Implementation



http://dtai.cs.kuleuven.be/wfomc

Thanks