# Towards a New Synthesis of Reasoning and Learning

Guy Van den Broeck

University of Toronto CS

Mar 25, 2019

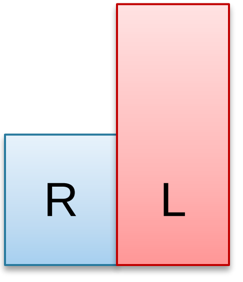
UCLA

STAR AI
RESEARCH LAB
UCLA

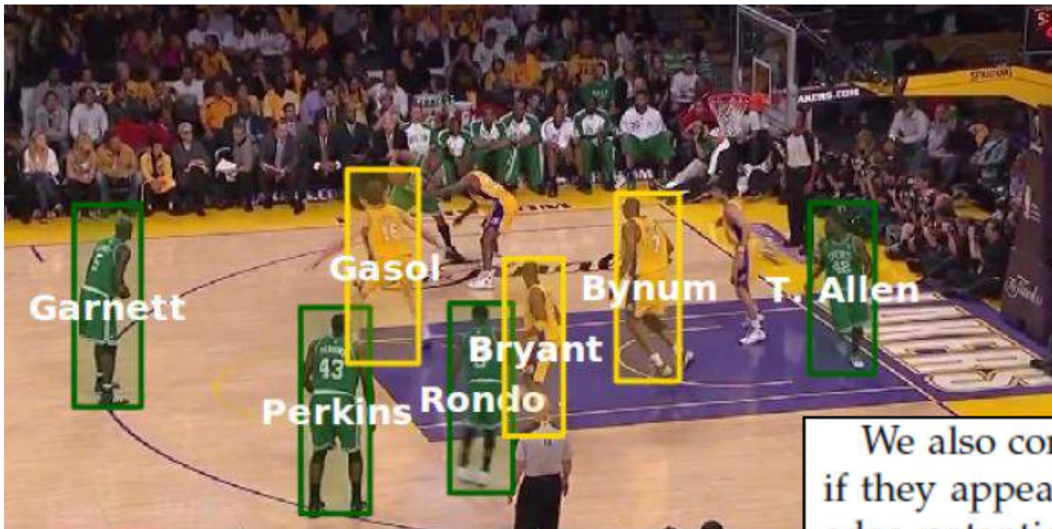
# Outline: Reasoning ∩ Learning

1. Deep Learning with Symbolic Knowledge
2. Efficient Reasoning During Learning
3. Probabilistic and Logistic Circuits
4. High-Level Probabilistic Reasoning

# *Deep Learning with Symbolic Knowledge*



# Motivation: Vision



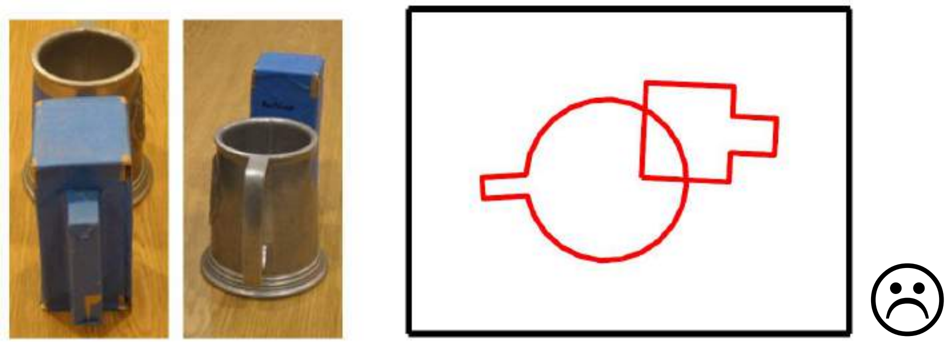

We also connect all pairs of identity nodes $y_{t,i}$ and $y_{t,j}$ if they appear in the same time $t$. We then introduce an edge potential that enforces mutual exclusion:

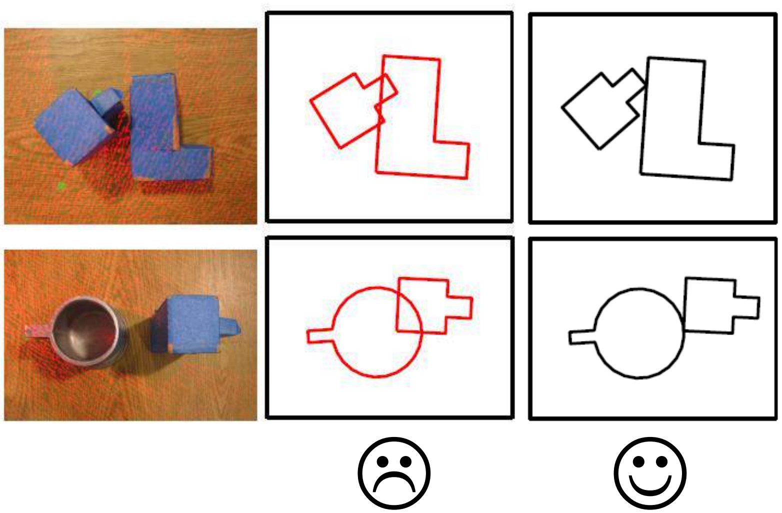
$$\psi_{\text{mutex}}(y_{t,i}, y_{t,j}) = \begin{cases} 1 & \text{if } y_{t,i} \neq y_{t,j} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

This potential specifies the constraint that a player can be appear only *once* in a frame. For example, if the $i$-th detection $y_{t,i}$ has been assign to Bryant, $y_{t,j}$ cannot have the same identity because Bryant is impossible to appear twice in a frame.

[Lu, W. L., Ting, J. A., Little, J. J., & Murphy, K. P. (2013). Learning to track and identify players from broadcast sports videos.]

# Motivation: Robotics





The method developed in this paper can be used in a broad variety of semantic mapping and object manipulation tasks, providing an efficient and effective way to incorporate collision constraints into a recursive state estimator, obtaining optimal or near-optimal solutions.

[Wong, L. L., Kaelbling, L. P., & Lozano-Perez, T., Collision-free state estimation. ICRA 2012]

# Motivation: Language

- Non-local dependencies:

  *"At least one verb in each sentence"*

- Sentence compression

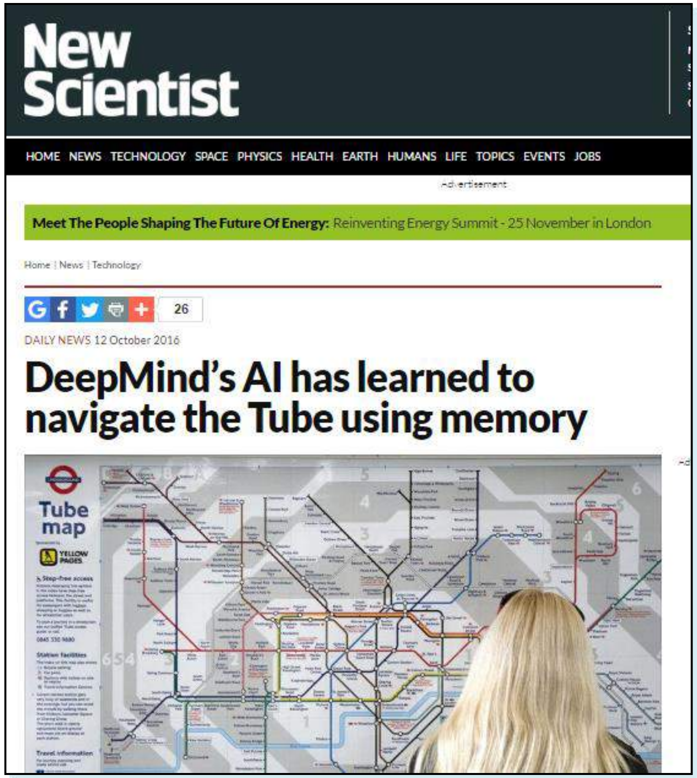
  *"If a modifier is kept, its subject is also kept"*

… and many more!

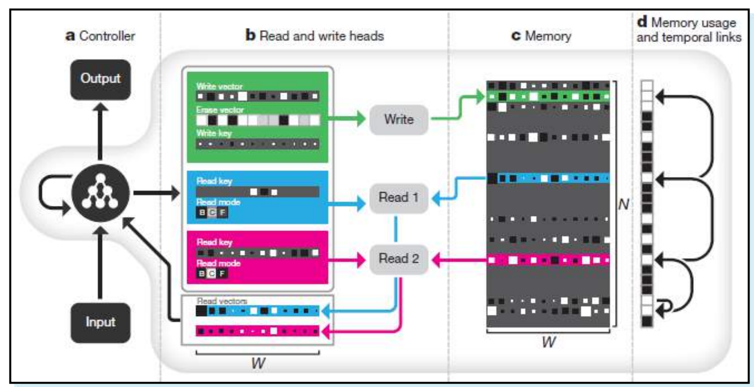
| Citations | |
|---|---|
| Start | The citation must start with author or editor. |
| AppearsOnce | Each field must be a consecutive list of words, and can appear at most once in a citation. |
| Punctuation | State transitions must occur on punctuation marks. |
| BookJournal | The words *proc*, *journal*, *proceedings*, *ACM* are *JOURNAL* or *BOOKTITLE*. |
| … | … |
| TechReport | The words *tech*, *technical* are *TECH_REPORT*. |
| Title | Quotations can appear only in titles. |
| Location | The words *CA*, *Australia*, *NY* are *LOCATION*. |

[Chang, M., Ratinov, L., & Roth, D. (2008). Constraints as prior knowledge],
[Ganchev, K., Gillenwater, J., & Taskar, B. (2010). Posterior regularization for structured latent variable models]

# Motivation: Deep Learning





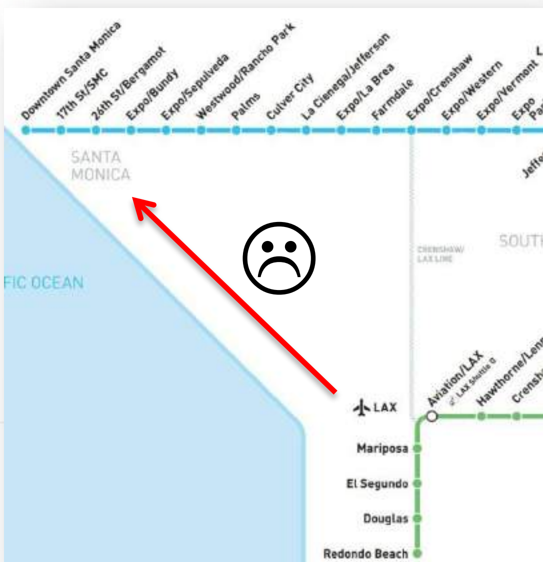



[Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., et al.. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*(7626), 471-476.]

# Motivation: Deep Learning

DeepMind's latest technique uses external memory to **solve tasks that require logic and reasoning — a step toward more human-like AI.**

… but …




☹

optimal planner recalculating a shortest path to the end node. **To ensure that the network always moved to a valid node, the output distribution was renormalized over the set of possible triples outgoing from the current node.** The performance

**it also received input triples during the answer phase, indicating the actions chosen on the previous time-step.** This makes the problem a 'structured prediction'

[Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., et al.. (2016). Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*(7626), 471-476.]

# Learning with Symbolic Knowledge

| L | K | P | A | Students |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 6 |
| 0 | 0 | 1 | 1 | 54 |
| 0 | 1 | 1 | 1 | 10 |
| 1 | 0 | 0 | 0 | 5 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 17 |
| 1 | 1 | 1 | 0 | 4 |
| 1 | 1 | 1 | 1 | 3 |

**Data** + **Constraints**

**(Background Knowledge)**
**(Physics)**

$$P \vee L$$
$$A \Rightarrow P$$
$$K \Rightarrow (P \vee L)$$

1. Must take at least one of Probability (**P**) or Logic (**L**).
2. Probability (**P**) is a prerequisite for AI (**A**).
3. The prerequisites for KR (**K**) is either AI (**A**) or Logic (**L**).

# Learning with Symbolic Knowledge

| L | K | P | A | Students |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 6 |
| 0 | 0 | 1 | 1 | 54 |
| 0 | 1 | 1 | 1 | 10 |
| 1 | 0 | 0 | 0 | 5 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 17 |
| 1 | 1 | 1 | 0 | 4 |
| 1 | 1 | 1 | 1 | 3 |

**Data** + **Constraints**
**(Background Knowledge)**
**(Physics)**

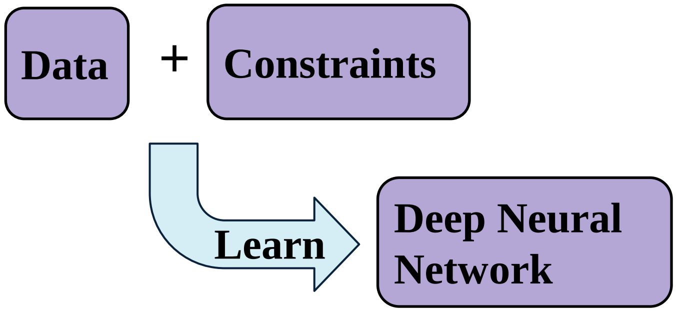
$$P \vee L$$
$$A \Rightarrow P$$
$$K \Rightarrow (P \vee L)$$

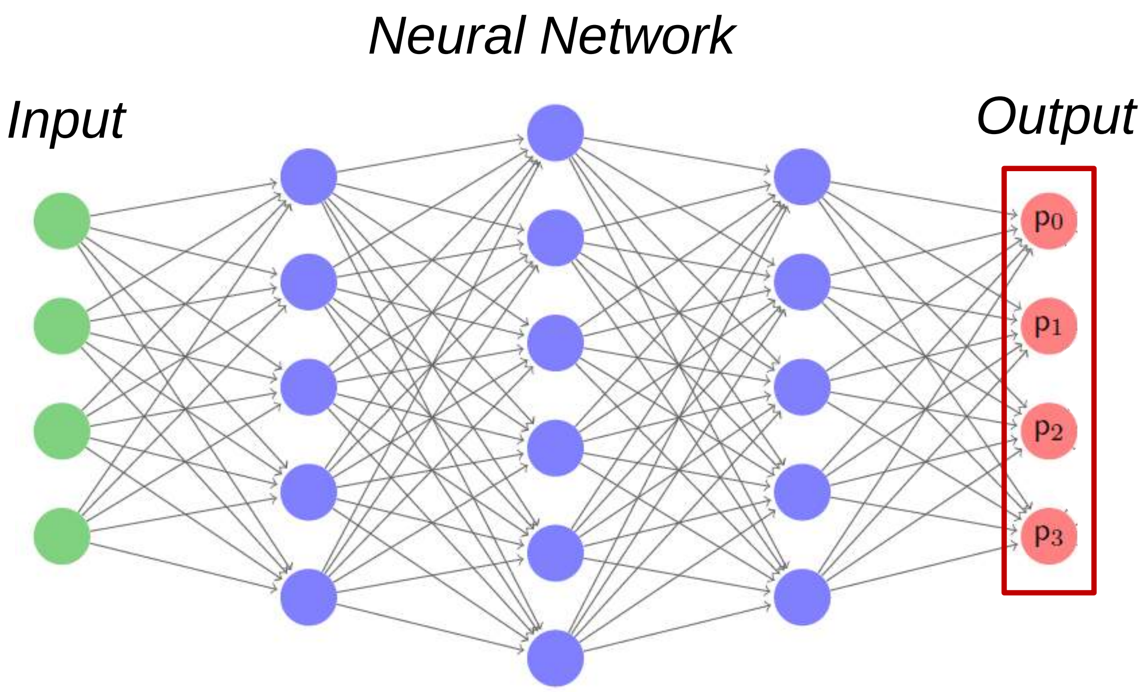
**Learn**

**ML Model**

Today's machine learning tools don't take knowledge as input! ☹

# Deep Learning with Symbolic Knowledge




Output is probability vector **p**, not Boolean logic!

# Semantic Loss

*<u>Q</u>: How close is output **p** to satisfying constraint α?*

*<u>Answer</u>: Semantic loss function L(α,**p**)*

- Axioms, for example:
  - If **p** is Boolean then L(**p**,**p**) = 0
  - If α implies β then L(α,**p**) ≥ L(β,**p**)     (*α more strict*)

- Implied Properties:
  - If α is equivalent to β then L(α,**p**) = L(β,**p**)
  - If **p** is Boolean and satisfies α then L(α,**p**) = 0

***SEMANTIC***
Loss!

# Semantic Loss: Definition

Theorem: Axioms imply unique semantic loss:

$$\mathrm{L}^{\mathrm{s}}(\alpha, \mathsf{p}) \propto -\log \underbrace{\sum_{\mathbf{x} \models \alpha} \underbrace{\prod_{i : \mathbf{x} \models X_i} \mathsf{p}_i \prod_{i : \mathbf{x} \models \neg X_i} (1 - \mathsf{p}_i)}_{\text{Probability of getting state } \mathbf{x} \text{ after flipping coins with probabilities } \mathbf{p}}}_{\text{Probability of satisfying } \alpha \text{ after flipping coins with probabilities } \mathbf{p}}$$

# Simple Example: Exactly-One

- Data must have some label

  *We agree this must be one of the 10 digits:*

- Exactly-one constraint

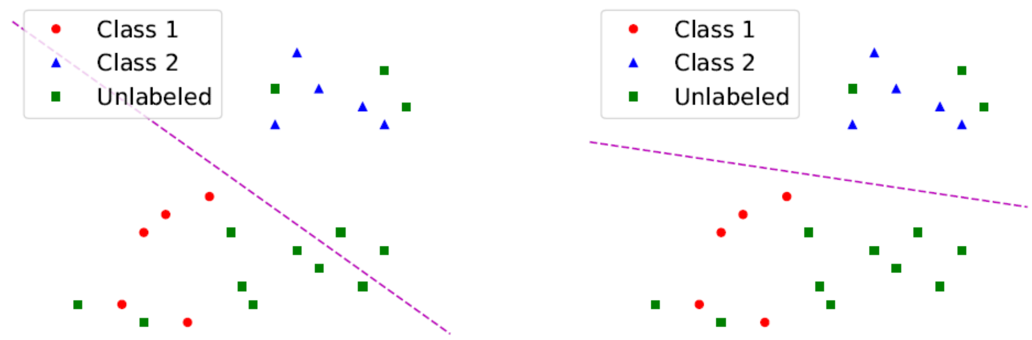
  → For 3 classes: $\begin{cases} \boldsymbol{x_1} \lor \boldsymbol{x_2} \lor \boldsymbol{x_3} \\ \neg \boldsymbol{x_1} \lor \neg \boldsymbol{x_2} \\ \neg \boldsymbol{x_2} \lor \neg \boldsymbol{x_3} \\ \neg \boldsymbol{x_1} \lor \neg \boldsymbol{x_3} \end{cases}$

- Semantic loss:

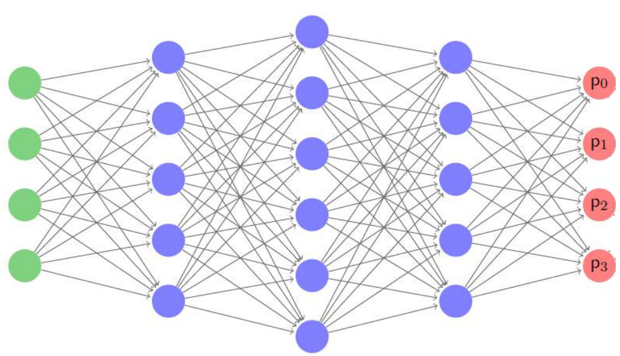
$$\mathrm{L^s}(\text{exactly-one}, \mathsf{p}) \propto -\log \sum_{i=1}^{n} \underbrace{\mathsf{p}_i \underbrace{\prod_{j=1, j \neq i}^{n} (1 - \mathsf{p}_j)}_{\text{Only } \boldsymbol{x_i} = \mathbf{1} \text{ after flipping coins}}}_{\text{Exactly one true } \boldsymbol{x} \text{ after flipping coins}}$$

# Semi-Supervised Learning

- Intuition: Unlabeled data must have some label
  Cf. entropy minimization, manifold learning




- Minimize exactly-one semantic loss on unlabeled data




Train with

$$existing\ loss + w \cdot semantic\ loss$$

# Experimental Evaluation

| Accuracy % with # of used labels | 100 | 1000 | ALL |
|---|---|---|---|
| AtlasRBF (Pitelis et al., 2014) | 91.9 (±0.95) | 96.32 (±0.12) | 98.69 |
| Deep Generative (Kingma et al., 2014) | 96.67(±0.14) | 97.60 (±0.02) | 99.04 |
| Virtual Adversarial (Miyato et al., 2016) | 97.67 | 98.64 | 99.36 |
| Ladder Net (Rasmus et al., 2015) | **98.94** (±0.37 ) | **99.16** (±0.08) | 99.43 (±0.02) |
| Baseline: MLP, Gaussian Noise | 78.46 (±1.94) | 94.26 (±0.31) | 99.34 (±0.08) |
| Baseline: Self-Training | 72.55 (±4.21) | 87.43 (±3.07) | |
| Baseline: MLP with Entropy Regularizer | 96.27 (±0.64) | 98.32 (±0.34) | 99.37 (±0.12) |
| MLP with Semantic Loss | 98.38 (±0.51) | 98.78 (±0.17) | 99.36 (±0.02) |

Competitive with state of the art in semi-supervised deep learning

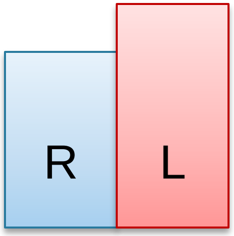
| Accuracy % with # of used labels | 100 | 500 | 1000 | ALL |
|---|---|---|---|---|
| Ladder Net (Rasmus et al., 2015) | 81.46 (±0.64 ) | 85.18 (±0.27) | 86.48 (±0.15) | 90.46 |
| Baseline: MLP, Gaussian Noise | 69.45 (±2.03) | 78.12 (±1.41) | 80.94 (±0.84) | 89.87 |
| MLP with Semantic Loss | **86.74** (±0.71) | **89.49** (±0.24) | **89.67** (±0.09) | 89.81 |

Outperforms SoA!

Same conclusion on CIFAR10

| Accuracy % with # of used labels | 4000 | ALL |
|---|---|---|
| CNN Baseline in Ladder Net | 76.67 (± 0.61) | 90.73 |
| Ladder Net (Rasmus et al., 2015) | 79.60 (±0.47) | |
| Baseline: CNN, Whitening, Cropping | 77.13 | 90.96 |
| CNN with Semantic Loss | **81.79** | 90.92 |

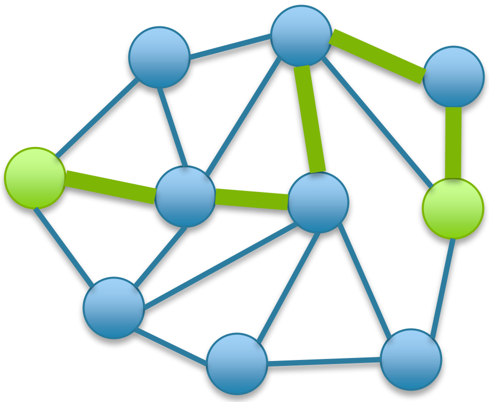
# *Efficient Reasoning During Learning*



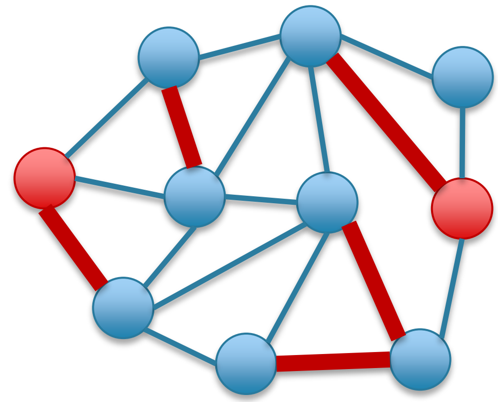
# But what about *real* constraints?

- Path constraint


cf. Nature paper



vs.



- Example: 4x4 grids

  $2^{24}$ = 184 paths + 16,777,032 non-paths

- Easily encoded as logical constraints ☺

[Nishino et al., Choi et al.]

# How to Compute Semantic Loss?

- In general: #P-hard ☹

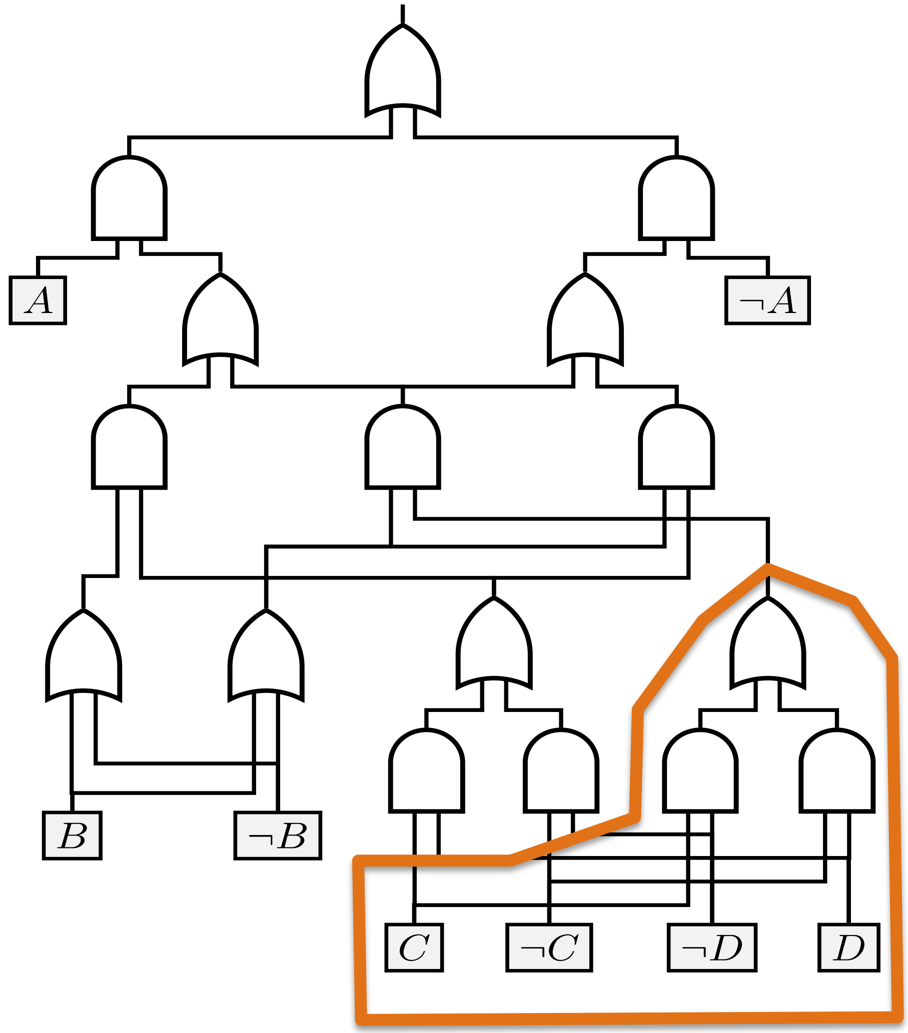
$$\mathrm{L}^{\mathrm{s}}(\alpha, \mathsf{p}) \propto -\log \sum_{\mathbf{x} \models \alpha} \prod_{i : \mathbf{x} \models X_i} \mathsf{p}_i \prod_{i : \mathbf{x} \models \neg X_i} (1 - \mathsf{p}_i)$$

# Reasoning Tool: Logical Circuits

Representation of logical sentences:

$(C \wedge \neg D) \vee (\neg C \wedge D)$

C XOR D



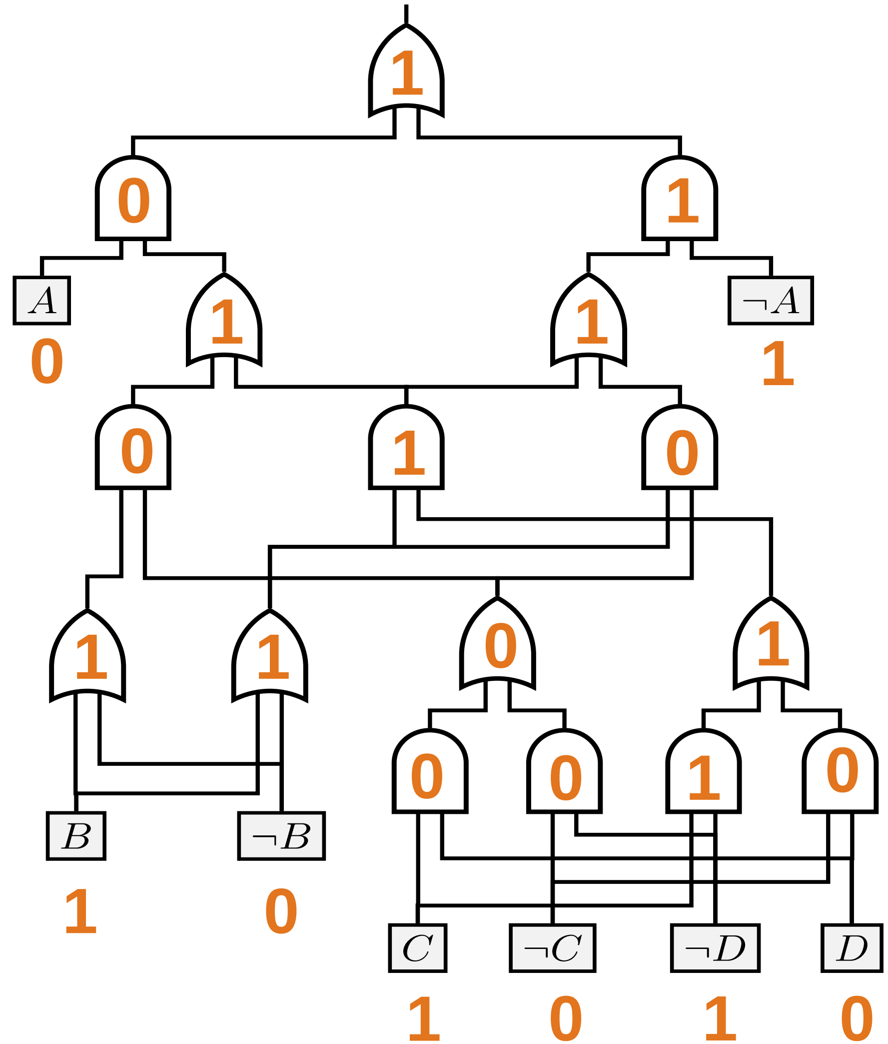
# Reasoning Tool: Logical Circuits

Representation of logical sentences:

Input:

| $A$ | $B$ | $C$ | $D$ |
|---|---|---|---|
| 0 | 1 | 1 | 0 |

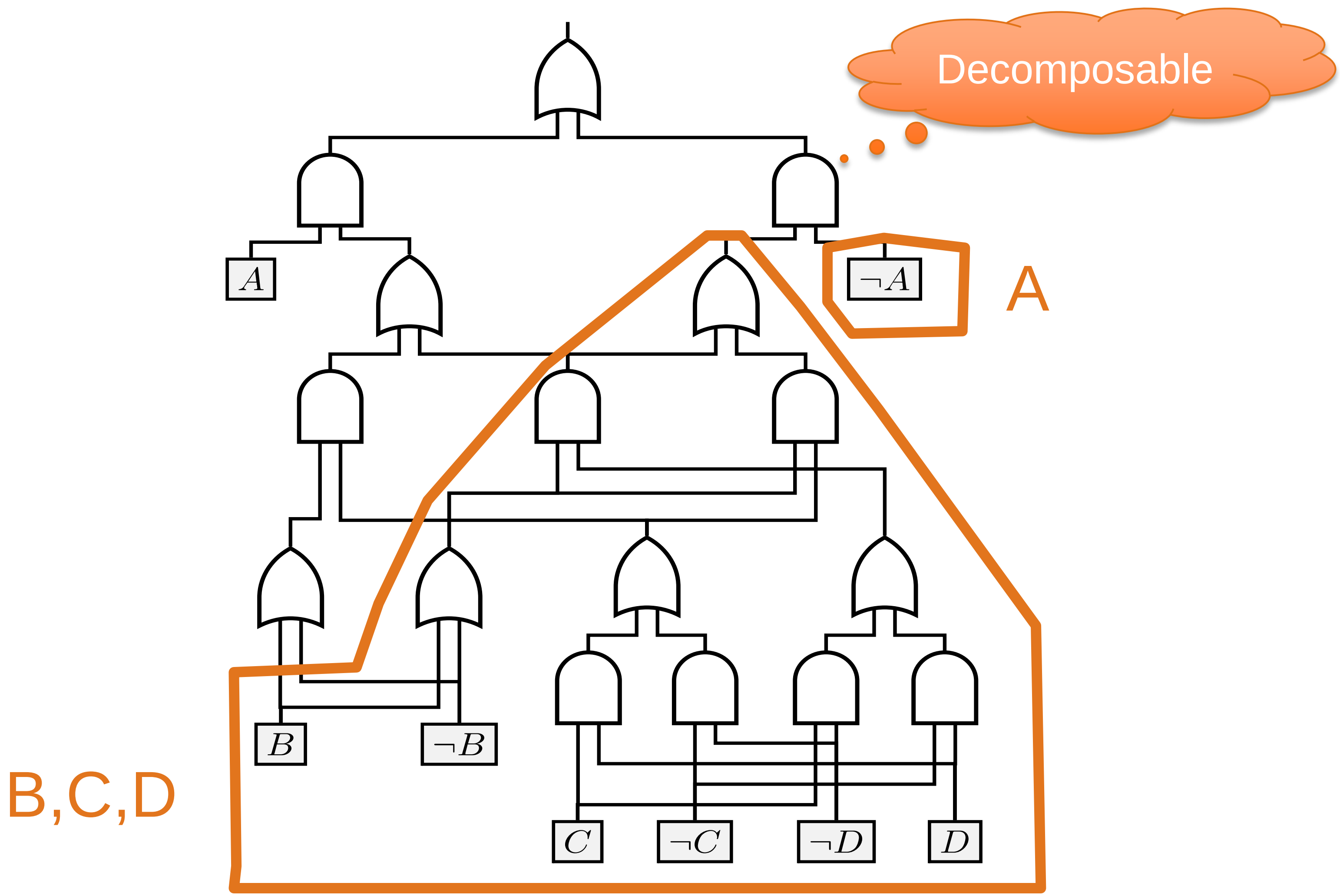
Bottom-up Evaluation



# Tractable for Logical Inference

- Is there a solution? (SAT)
  - SAT($\alpha \lor \beta$) iff SAT($\alpha$) or SAT($\beta$) (*always*)
  - SAT($\alpha \land \beta$) iff ***???***

# Decomposable Circuits

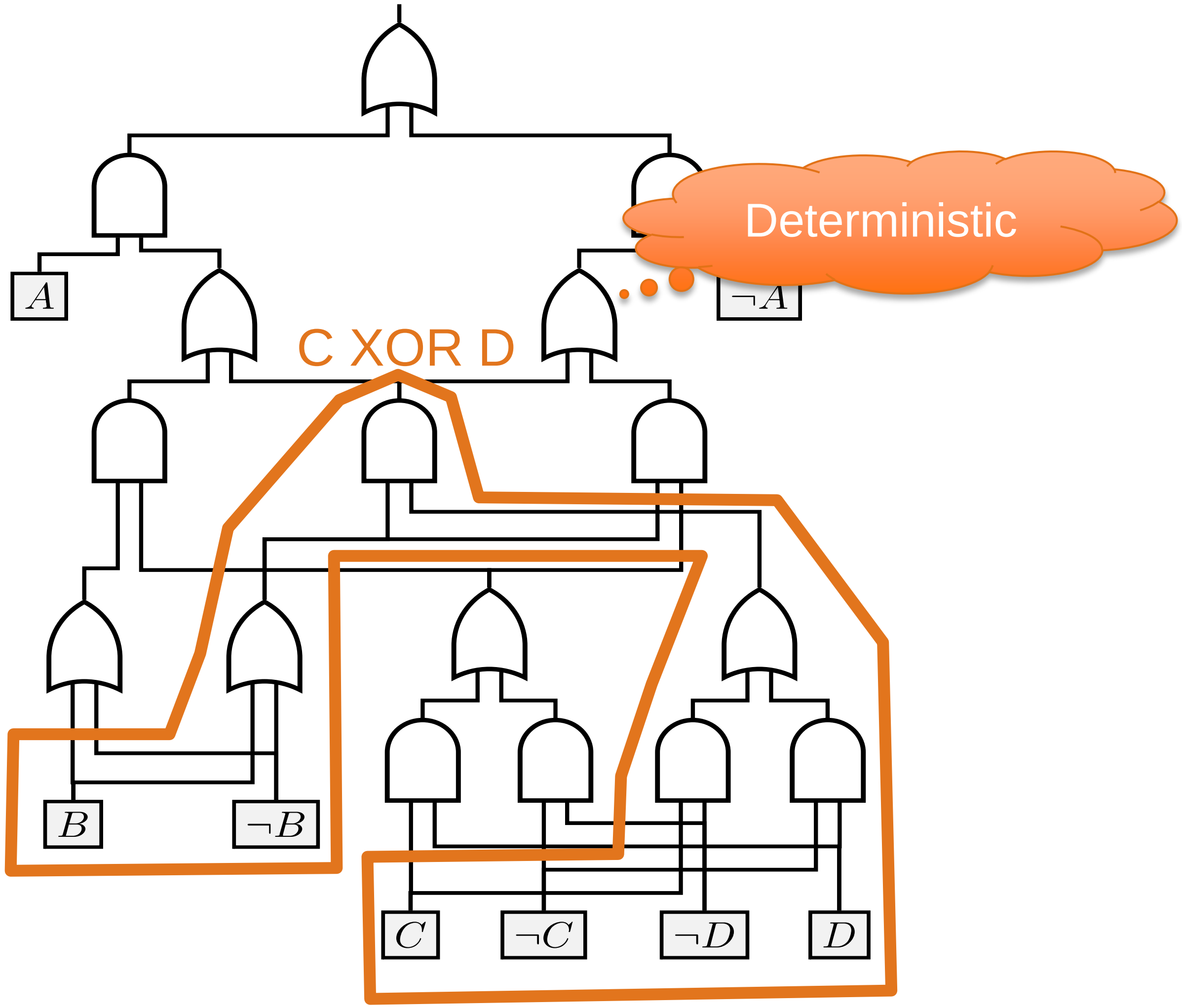


# Tractable for Logical Inference

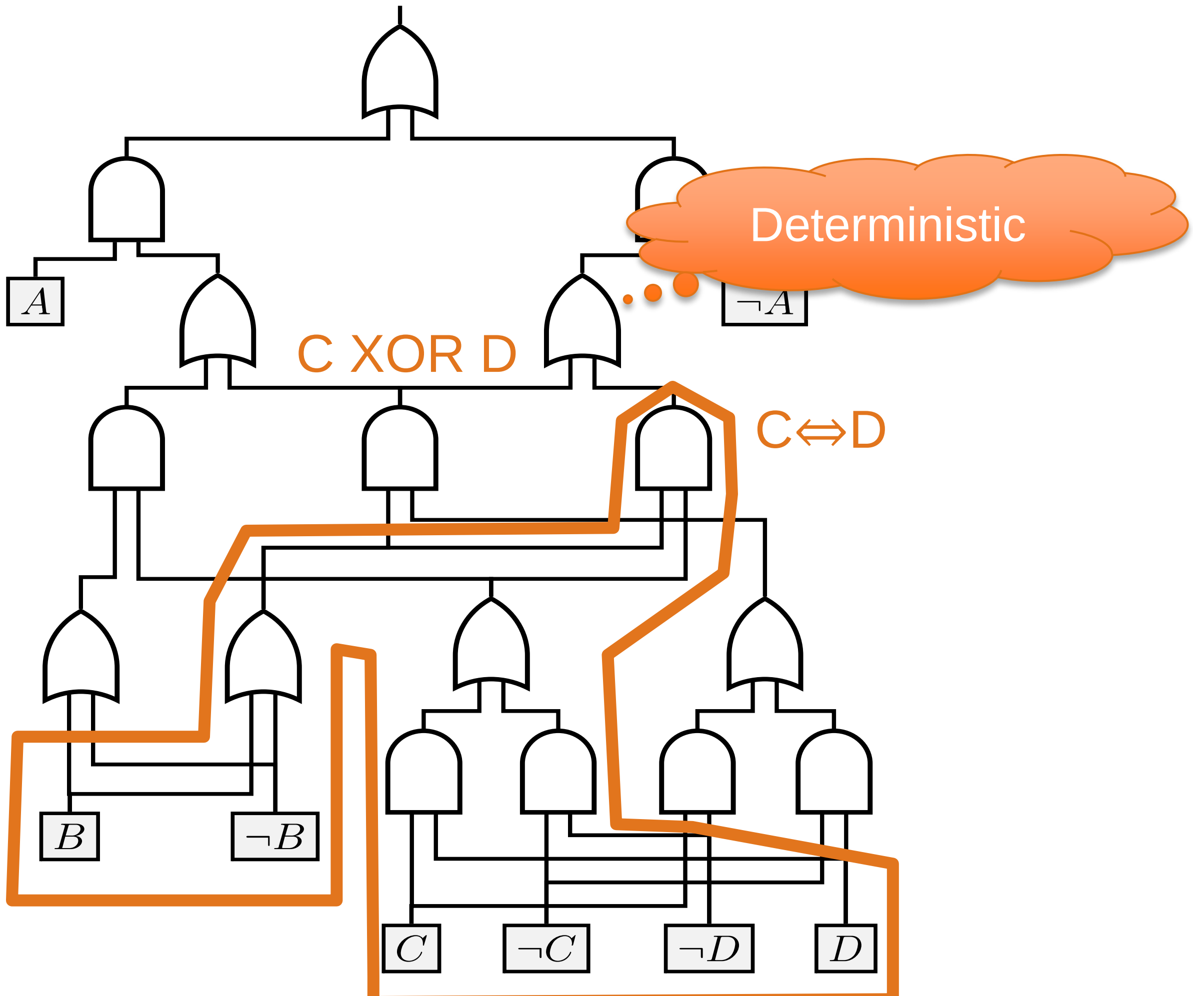
- Is there a solution? (SAT) ✓
  - SAT($\alpha \vee \beta$) iff SAT($\alpha$) or SAT($\beta$) (*always*)
  - SAT($\alpha \wedge \beta$) iff SAT($\alpha$) and SAT($\beta$) (*decomposable*)
- How many solutions are there? (#SAT)

- Complexity linear in circuit size ☺

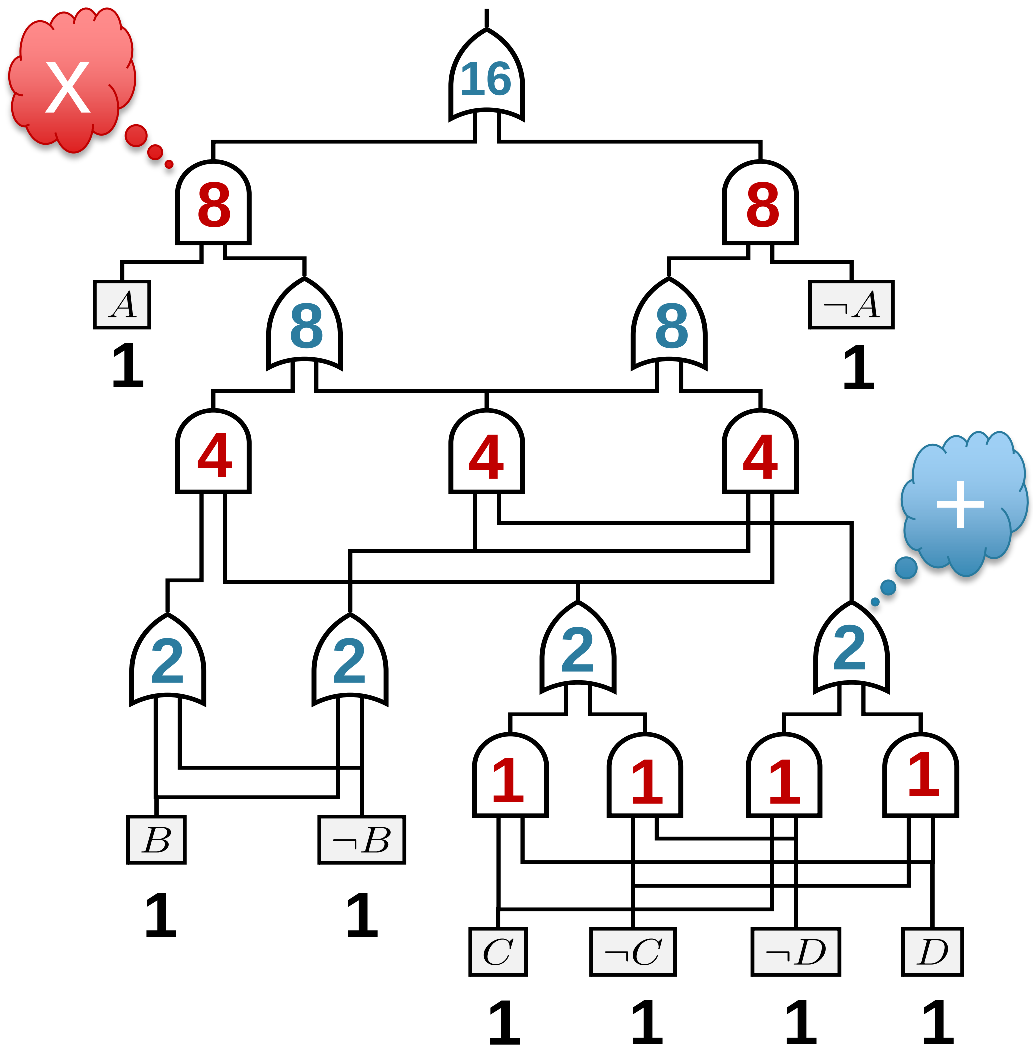
# Deterministic Circuits



# Deterministic Circuits



# How many solutions are there? (#SAT)



# Tractable for Logical Inference

- Is there a solution? (SAT) ✓
- How many solutions are there? (#SAT) ✓

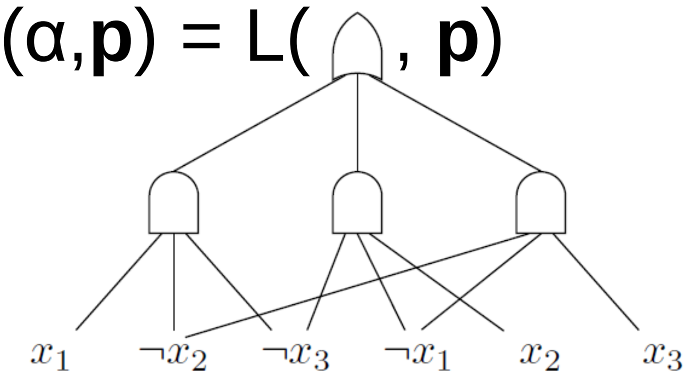
- Complexity linear in circuit size ☺

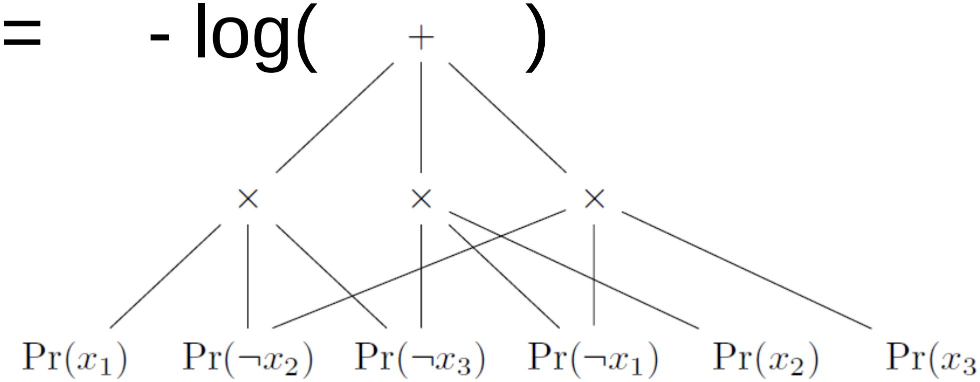
- Compilation into circuit by
  - ↓ exhaustive SAT solver
  - ↑ conjoin/disjoin/negate

[Darwiche and Marquis, JAIR 2002]

# How to Compute Semantic Loss?

- In general: #P-hard ☹
- With a logical circuit for α: Linear ☺
- Example: exactly-one constraint:

$L(\alpha,\mathbf{p}) = L($ 

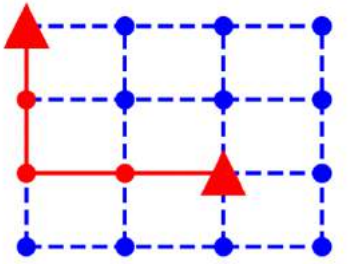
$, \mathbf{p}) = -\log($ 

$)$

- *Why?* Decomposability and determinism!

# Predict Shortest Paths

Add semantic loss
for path constraint



| Test accuracy % | Coherent | Incoherent | Constraint |
|---|---|---|---|
| 5-layer MLP | 5.62 | **85.91** | 6.99 |
| Semantic loss | **28.51** | 83.14 | **69.89** |

*Is prediction*
*the shortest path?*
**This is the real task!**

*Are individual*
*edge predictions*
*correct?*

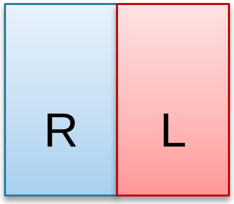
*Is output*
*a path?*

(same conclusion for predicting sushi preferences, see paper)

# Conclusions 1

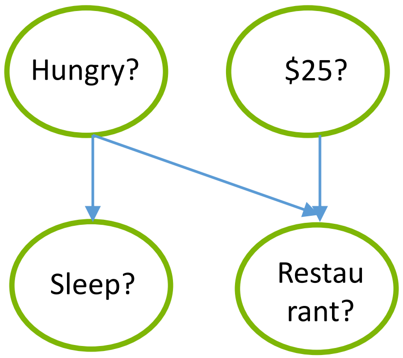
- Knowledge is (hidden) everywhere in ML
- Semantic loss makes logic differentiable
- Performs well semi-supervised
- Requires hard reasoning in general
  - Reasoning can be encapsulated in a circuit
  - No overhead during learning
- Performs well on structured prediction
- A little bit of reasoning goes a long way!

# *Probabilistic and Logistic Circuits*

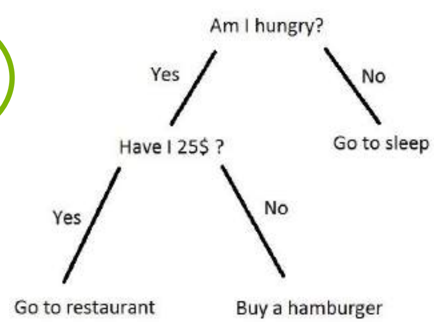


# A False Dilemma?

## Classical AI Methods

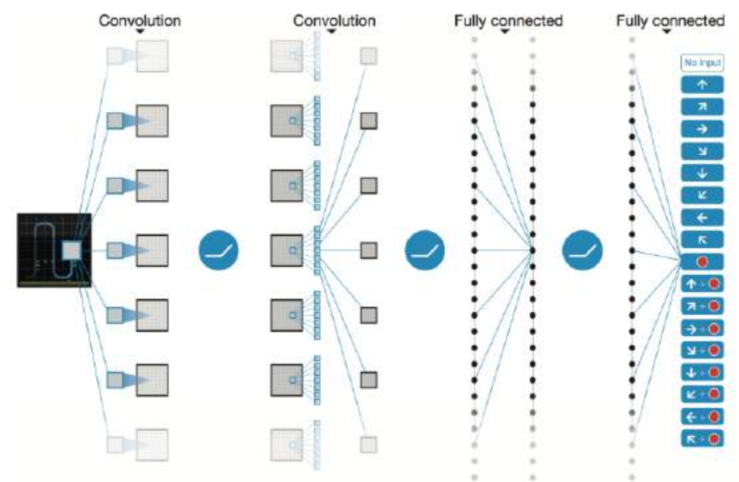



...




Clear Modeling Assumption
Well-understood

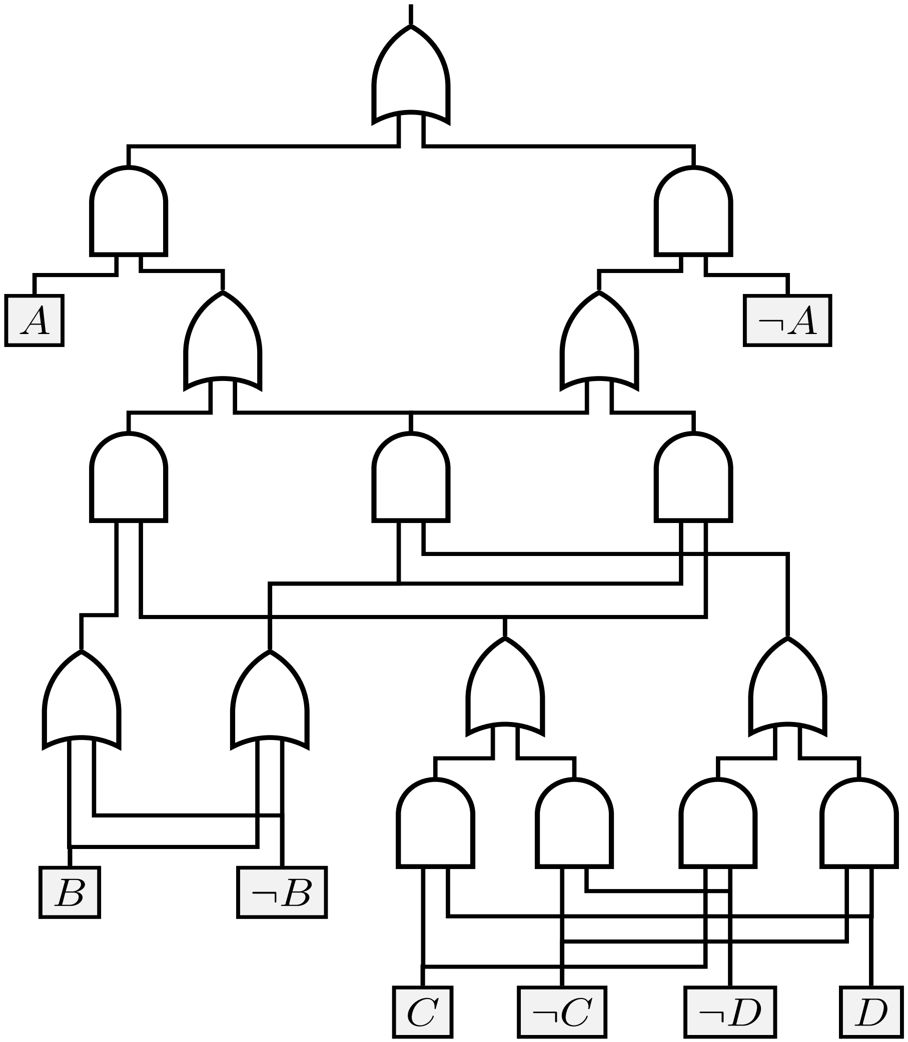
## Neural Networks




“Black Box”
Empirical performance

# Inspiration: Probabilistic Circuits

Can we turn **logic circuits** into a **statistical model?**



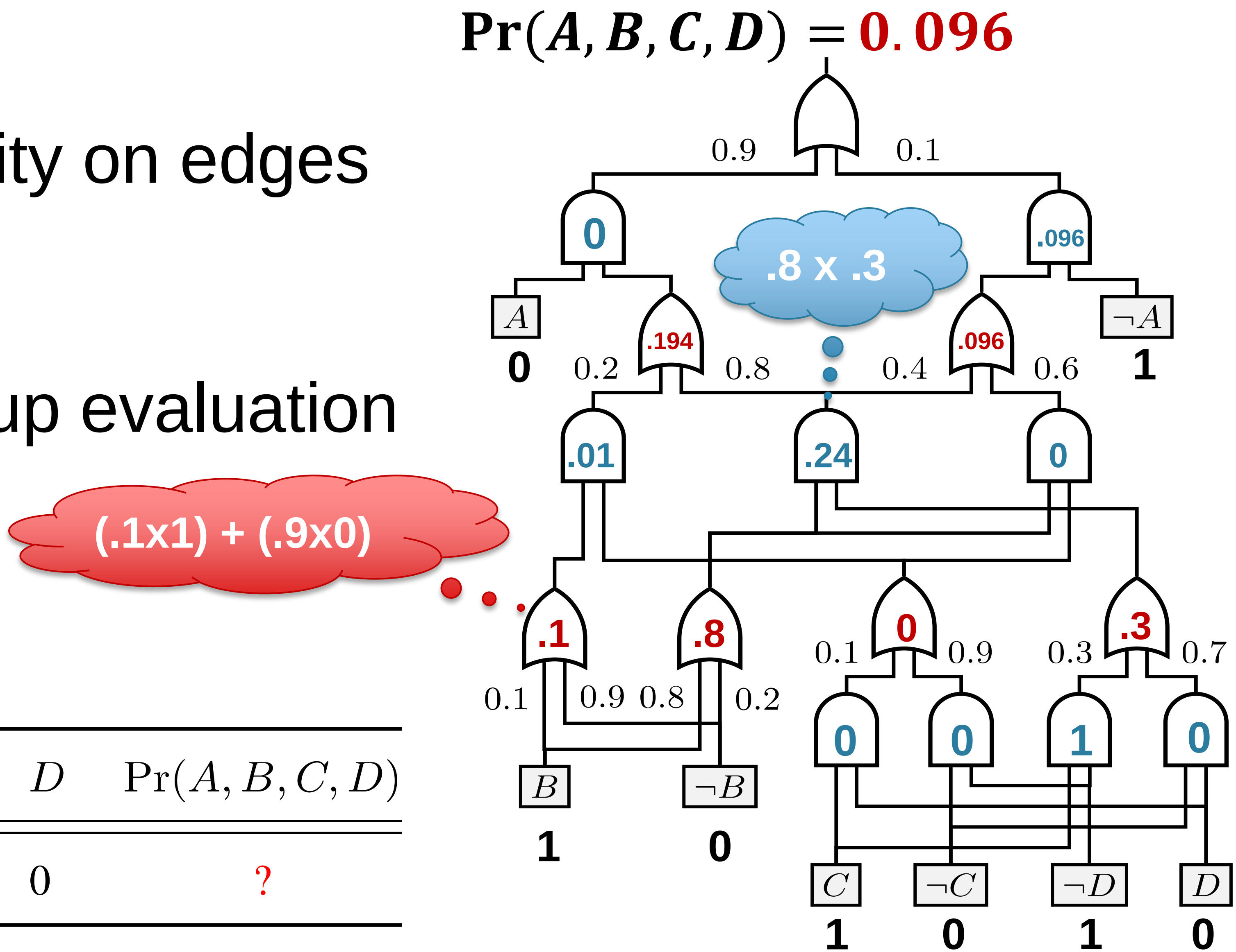
# Probabilistic Circuits

Probability on edges

Bottom-up evaluation

Input:

| $A$ | $B$ | $C$ | $D$ | $\Pr(A, B, C, D)$ |
|---|---|---|---|---|
| 0 | 1 | 1 | 0 | ? |



# Properties, Properties, Properties!

- Read conditional independencies from structure
- Interpretable parameters (XAI)
  (conditional probabilities of logical sentences)
- Closed-form parameter learning
- Efficient reasoning

  – **MAP inference**: most-likely assignment to x given y
    (otherwise NP-hard)
  – Computing **conditional probabilities** Pr(x|y)
    (otherwise #P-hard)
  – Algorithms linear in circuit size ☺

# Discrete Density Estimation

| Datasets | \|Var\| | LearnPSDD Ensemble | Best-to-Date |
|---|---|---|---|
| NLTCS | 16 | $-5.99^\dagger$ | $-6.00$ |
| MSNBC | 17 | $-6.04^\dagger$ | $-6.04^\dagger$ |
| KDD | 64 | $-2.11^\dagger$ | $-2.12$ |
| Plants | 69 | $-13.02$ | $-11.99^\dagger$ |
| Audio | 100 | $-39.94$ | $-39.49^\dagger$ |
| Jester | 100 | $-51.29$ | $-41.11^\dagger$ |
| Netflix | 100 | $-55.71^\dagger$ | $-55.84$ |
| Accidents | 111 | $-30.16$ | $-24.87^\dagger$ |
| Retail | 135 | $-10.72^\dagger$ | $-10.78$ |
| Pumsb-Star | 163 | $-26.12$ | $-22.40^\dagger$ |
| DNA | 180 | $-88.01$ | $-80.03^\dagger$ |
| Kosarek | 190 | $-10.52^\dagger$ | $-10.54$ |
| MSWeb | 294 | $-9.89$ | $-9.22^\dagger$ |
| Book | 500 | $-34.97$ | $-30.18^\dagger$ |
| EachMovie | 500 | $-58.01$ | $-51.14^\dagger$ |
| WebKB | 839 | $-161.09$ | $-150.10^\dagger$ |
| Reuters-52 | 889 | $-89.61$ | $-80.66^\dagger$ |
| 20NewsGrp. | 910 | $-155.97$ | $-150.88^\dagger$ |
| BBC | 1058 | $-253.19$ | $-233.26^\dagger$ |
| AD | 1556 | $-31.78$ | $-14.36^\dagger$ |

LearnPSDD state of the art on 6 datasets!

*Q: “Help! I need to learn a discrete probability distribution…”*
A: Learn probabilistic circuits!

Strongly outperforms

- Bayesian network learners
- Markov network learners

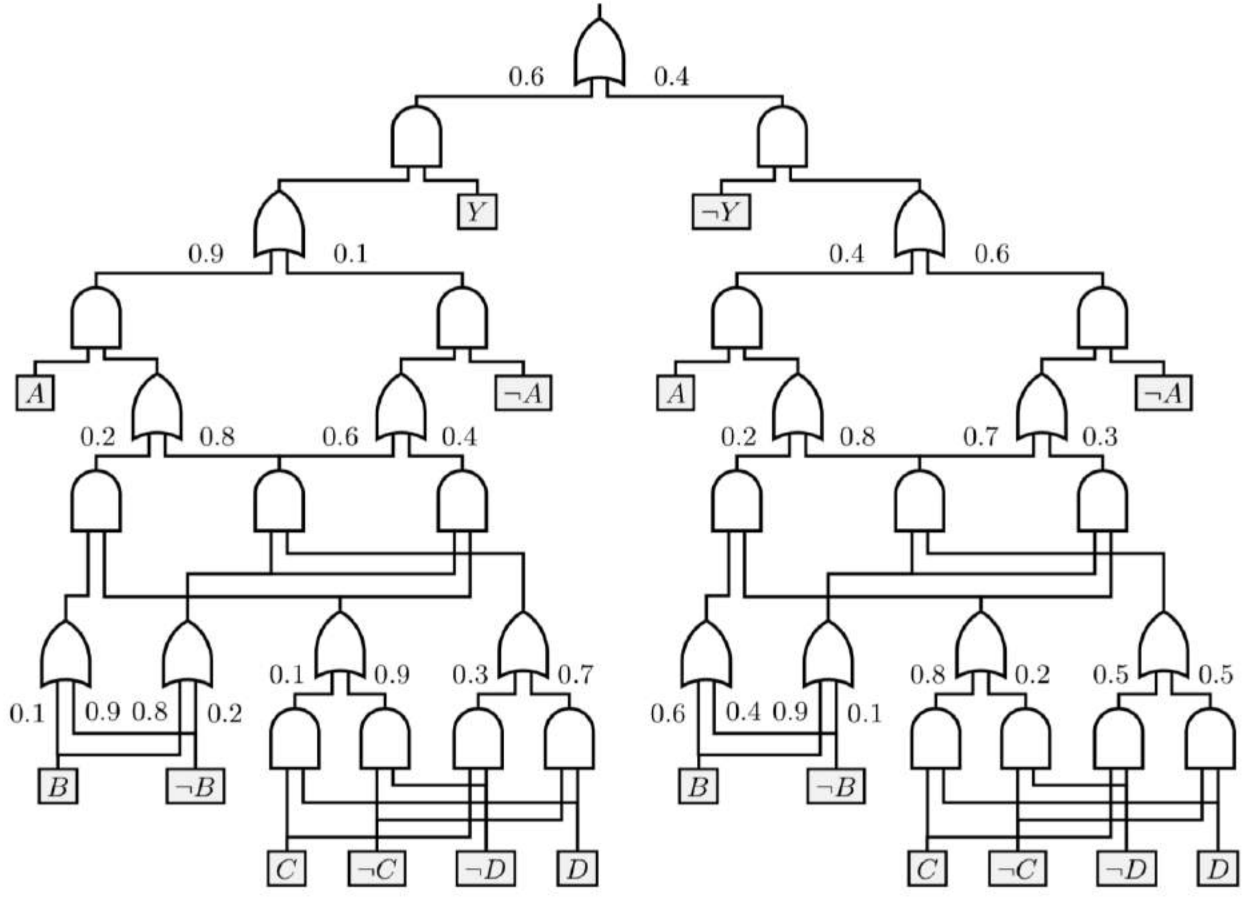
Competitive with SPN learners

*(State of the art for approximate inference in discrete factor graphs)*

# *But what if I only want to classify Y?*

$$\Pr(Y|A,B,C,D)$$

$$\cancel{\Pr(Y,A,B,C,D)}$$



# Logistic Circuits

$$\Pr(Y = 1 \mid A, B, C, D) = \frac{1}{1 + exp(-1.9)} = 0.869$$

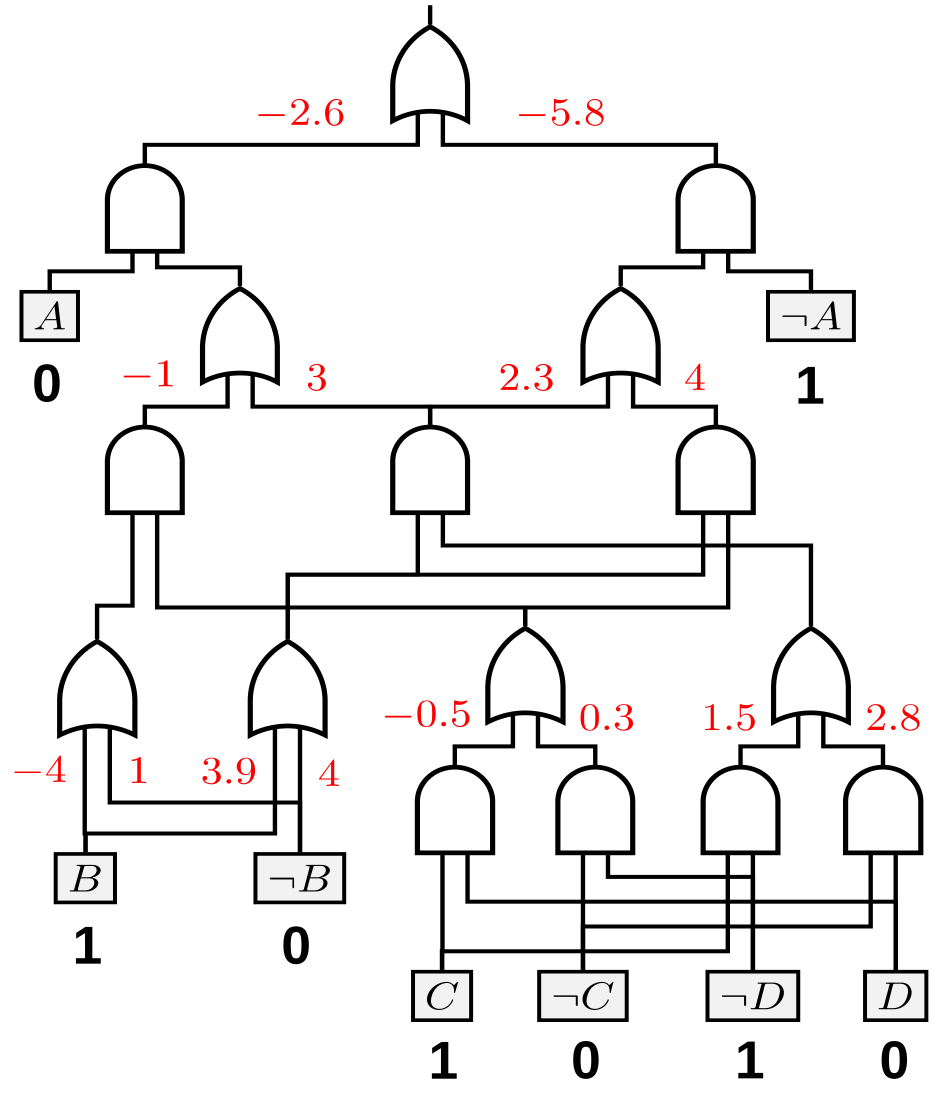
Weights on edges

Logistic function on output weight

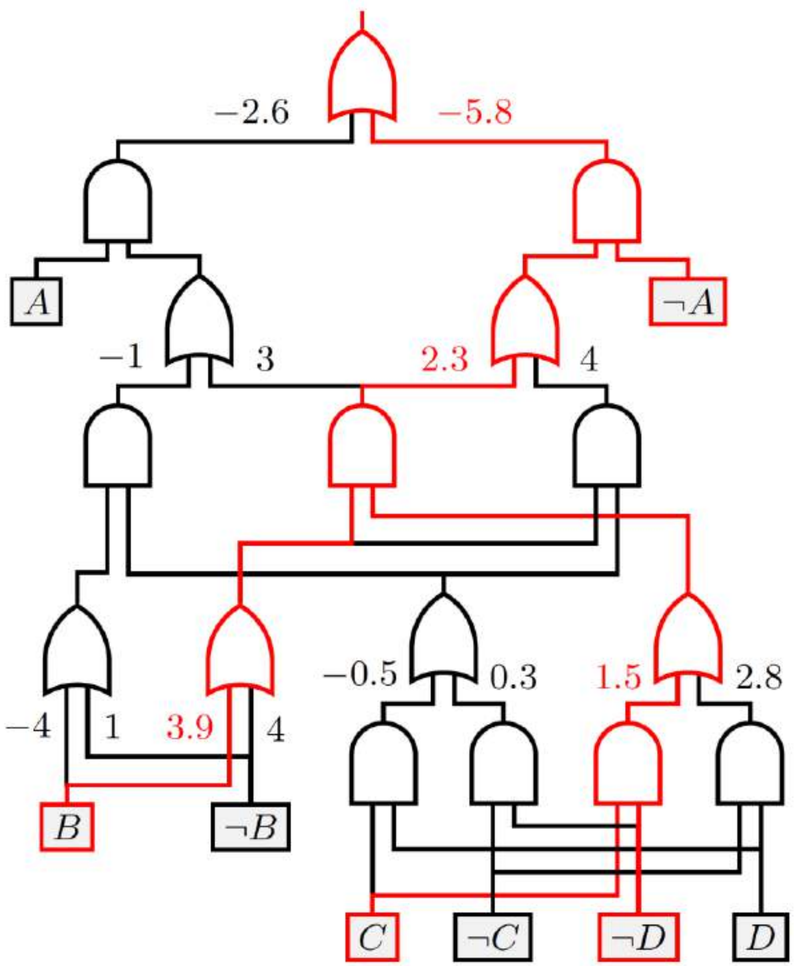
Bottom-up evaluation

Input:

| $A$ | $B$ | $C$ | $D$ | $\Pr(Y \mid A, B, C, D)$ |
|---|---|---|---|---|
| 0 | 1 | 1 | 0 | ? |



# Alternative Semantics

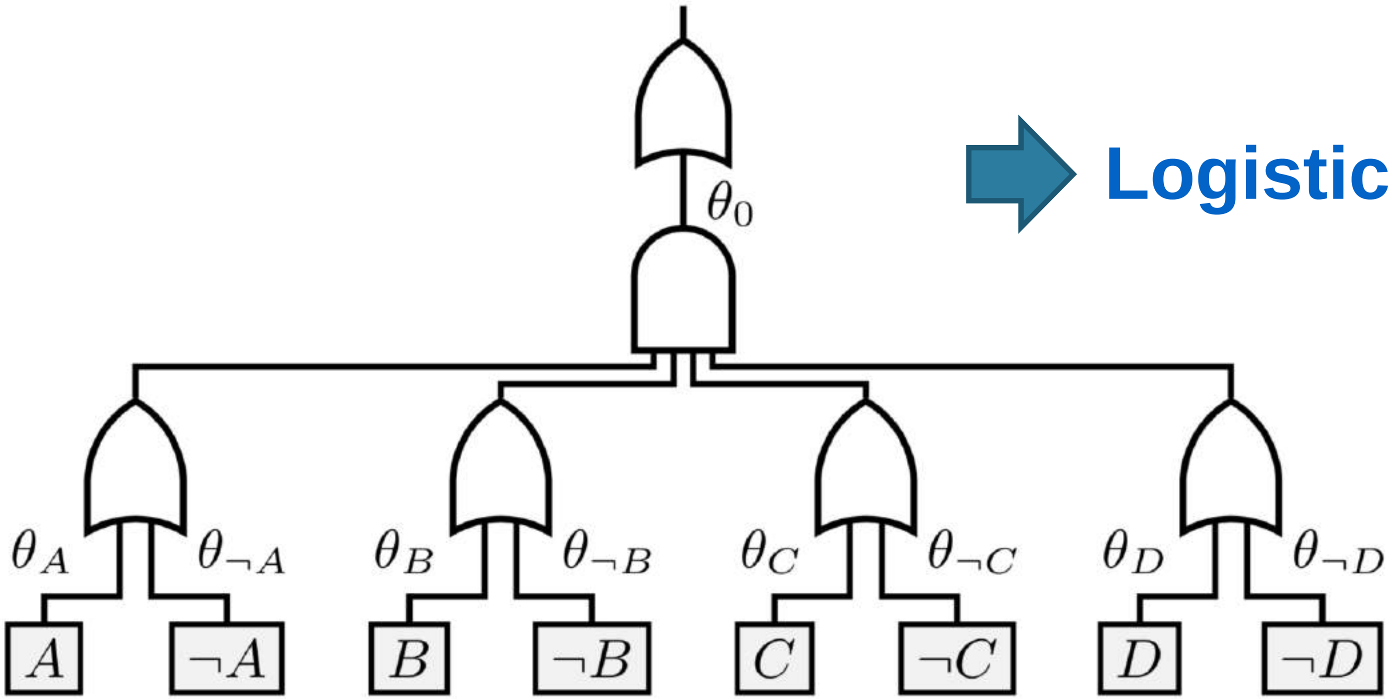



Represents $\Pr(Y \mid A, B, C, D)$

- Take all 'hot' wires
- Sum their weights
- Push through logistic function

| $A$ | $B$ | $C$ | $D$ | $g_r(ABCD)$ | $\Pr(Y = 1 \mid ABCD)$ |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | -3.1 | 4.31% |
| 0 | 1 | 1 | 0 | 1.9 | 86.99% |
| 1 | 1 | 1 | 0 | 5.8 | 99.70% |

# Special Case: Logistic Regression




➡ **Logistic Regression**

$$\Pr(Y = 1 | A, B, C, D) = \frac{1}{1 + \exp(-A * \theta_A - \neg A * \theta_{\neg A} - B * \theta_B - \cdots)}$$

Is this a coincidence?

What about more general circuits?

# Parameter Learning

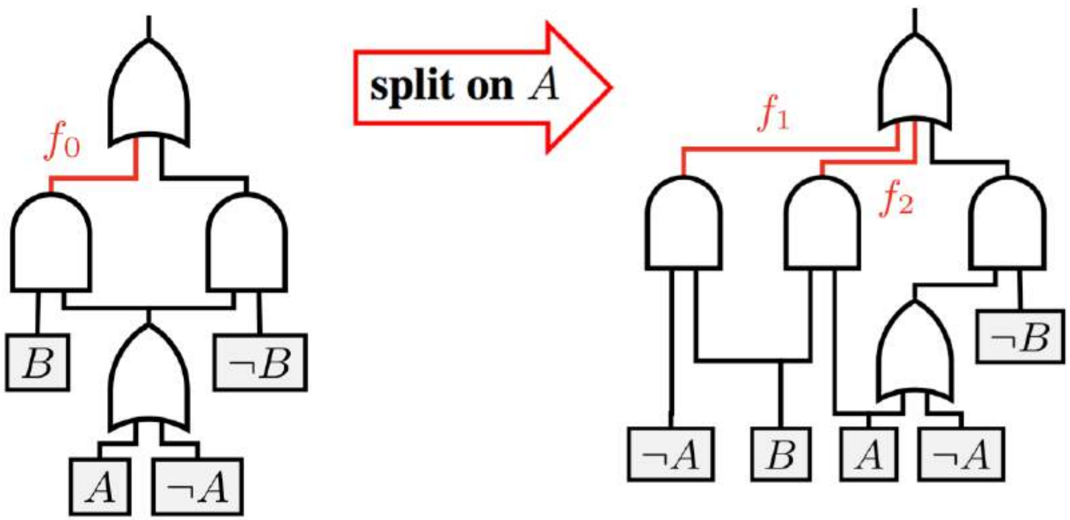
Reduce to logistic regression:

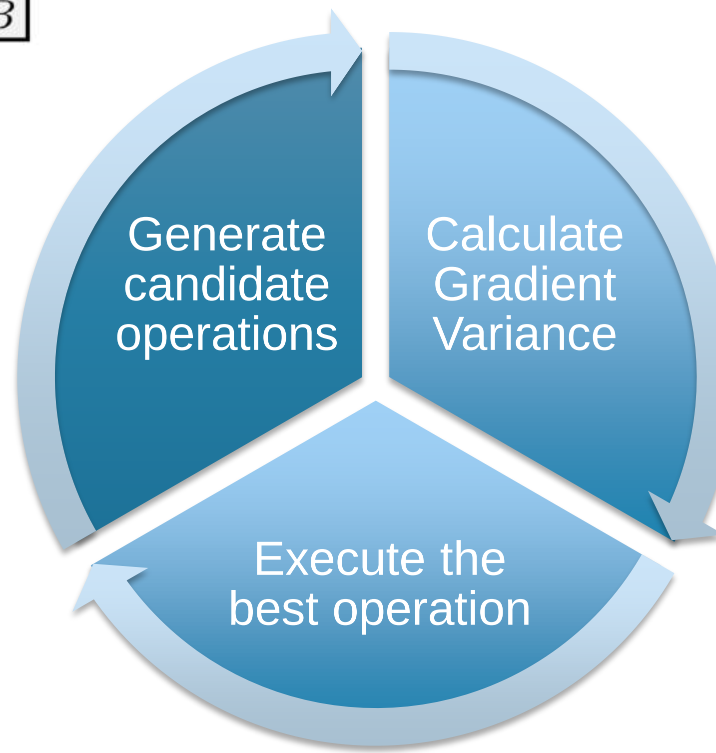
$$\Pr(Y = 1 \mid \mathbf{x}) = \frac{1}{1 + \exp(-\mathbf{x} \cdot \boldsymbol{\theta})}$$

Features associated with each wire
"Global Circuit Flow" features

**Learning parameters θ is convex optimization!**

# Logistic Circuit Structure Learning



# Comparable Accuracy with Neural Nets

| Accuracy % on Dataset | Mnist | Fashion |
|---|---|---|
| Baseline: Logistic Regression | 85.3 | 79.3 |
| Baseline: Kernel Logistic Regression | 97.7 | 88.3 |
| Random Forest | 97.3 | 81.6 |
| 3-layer MLP | 97.5 | 84.8 |
| RAT-SPN (Peharz et al. 2018) | 98.1 | 89.5 |
| SVM with RBF Kernel | 98.5 | 87.8 |
| 5-Layer MLP | 99.3 | 89.8 |
| Logistic Circuit (binary) | 97.4 | 87.6 |
| Logistic Circuit (real-valued) | 99.4 | 91.3 |
| CNN with 3 conv layers | 99.1 | 90.7 |
| Resnet (He et al. 2016) | 99.5 | 93.6 |

# Significantly Smaller in Size

| Number of Parameters | Mnist | Fashion |
|---|---|---|
| Baseline: Logistic Regression | <1K | <1K |
| Baseline: Kernel Logistic Regression | 1,521 K | 3,930K |
| Logistic Circuit (real-valued) | 182K | 467K |
| Logistic Circuit (binary) | 268K | 614K |
| 3-layer MLP | 1,411K | 1,411K |
| RAT-SPN (Peharz et al. 2018) | 8,500K | 650K |
| CNN with 3 conv layers | 2,196K | 2,196K |
| 5-layer MLP | 2,411K | 2,411K |
| Resnet (He et al. 2016) | 4,838K | 4,838K |

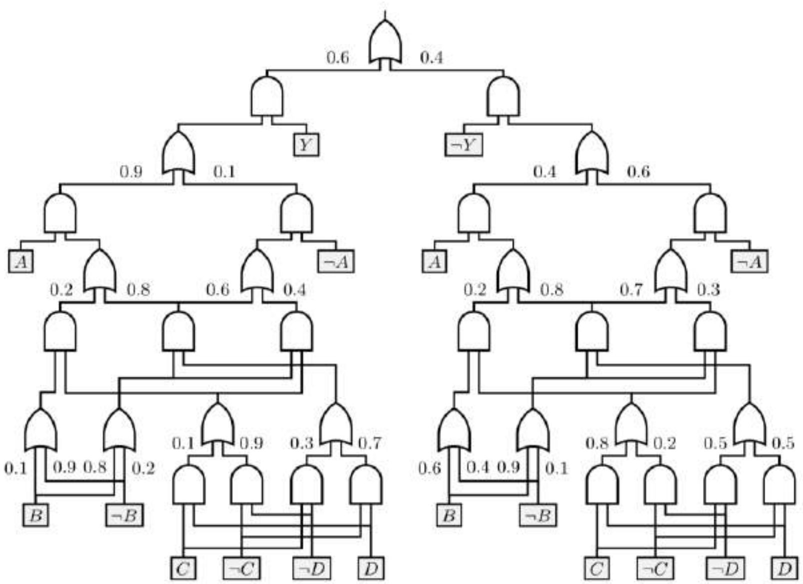
# Better Data Efficiency

| Accuracy % with % of Training Data | MNIST | | | Fashion | | |
|---|---|---|---|---|---|---|
| | 100% | 10% | 2% | 100% | 10% | 2% |
| 5-Layer MLP | 99.3 | **98.2** | 94.3 | 89.8 | 86.5 | 80.9 |
| CNN with 3 Conv Layers | 99.1 | 98.1 | 95.3 | 90.7 | 87.6 | 83.8 |
| Logistic Circuit (Binary) | 97.4 | 96.9 | 94.1 | 87.6 | 86.7 | 83.2 |
| Logistic Circuit (Real-Valued) | **99.4** | 97.6 | **96.1** | **91.3** | **87.8** | **86.0** |

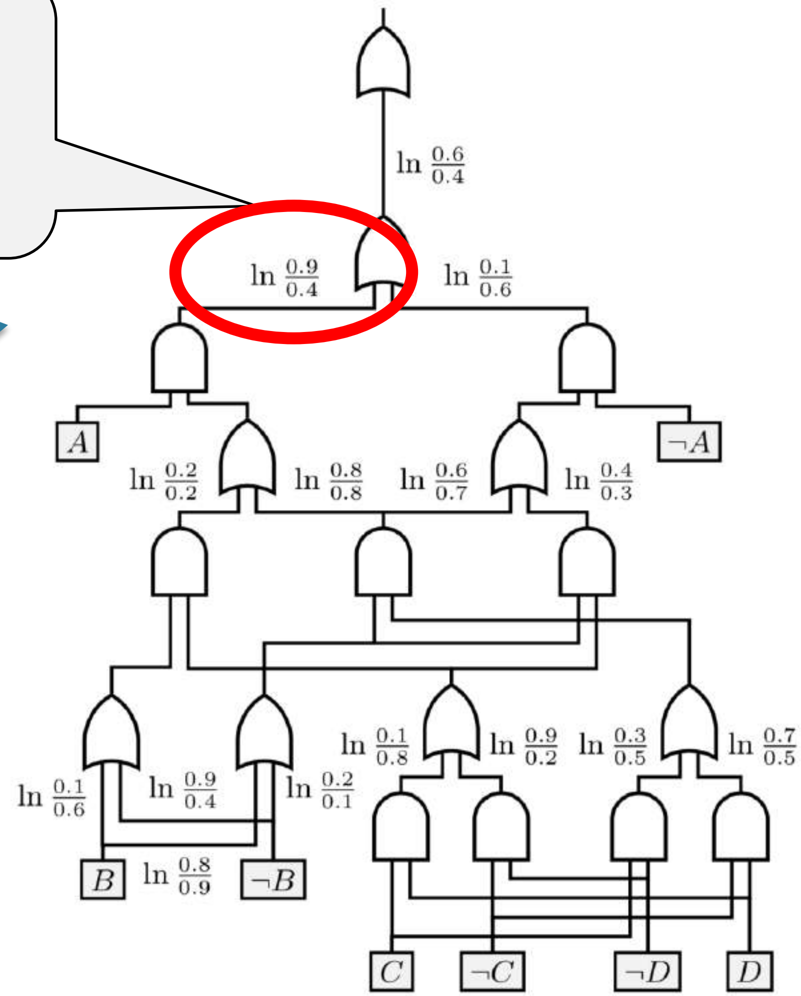
# Logistic vs. Probabilistic Circuits

Probabilities
become
log-odds

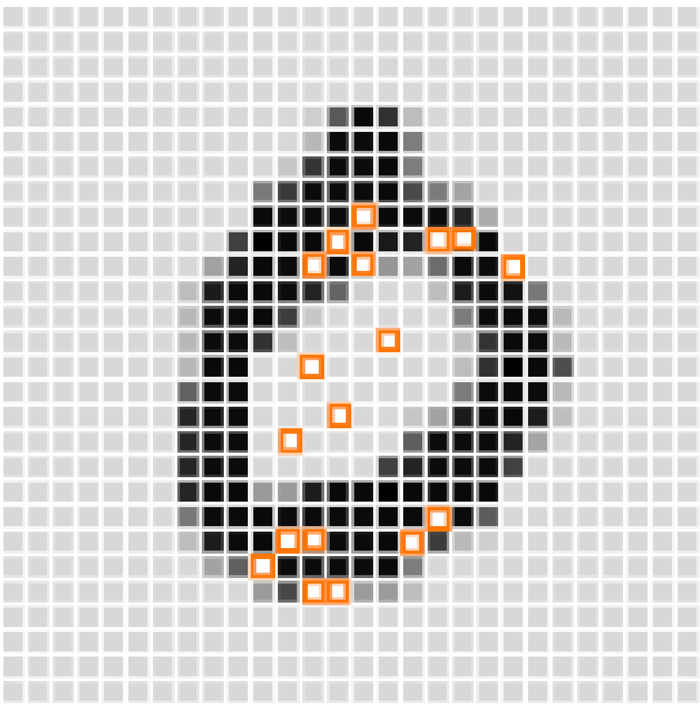
$\Pr(Y, A, B, C, D)$

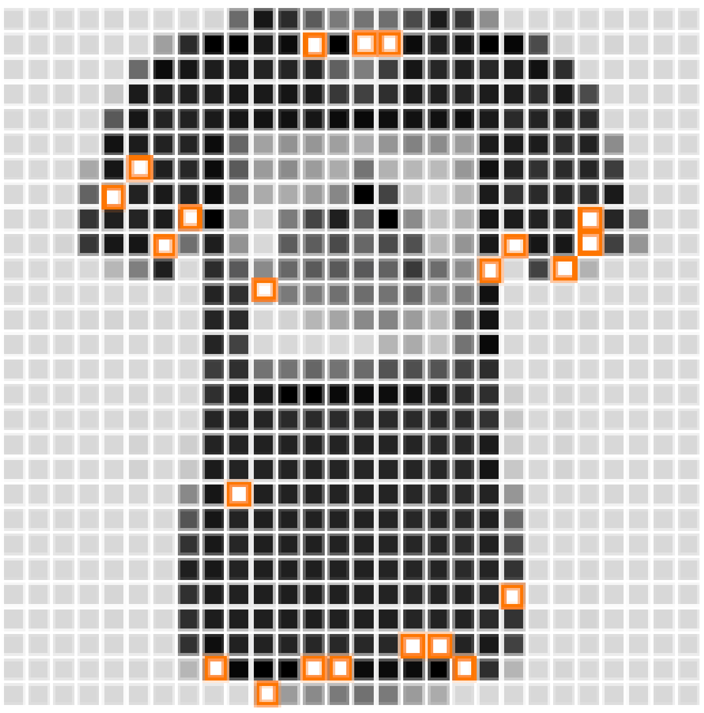


$\Pr(\, Y \mid A, B, C, D \,)$

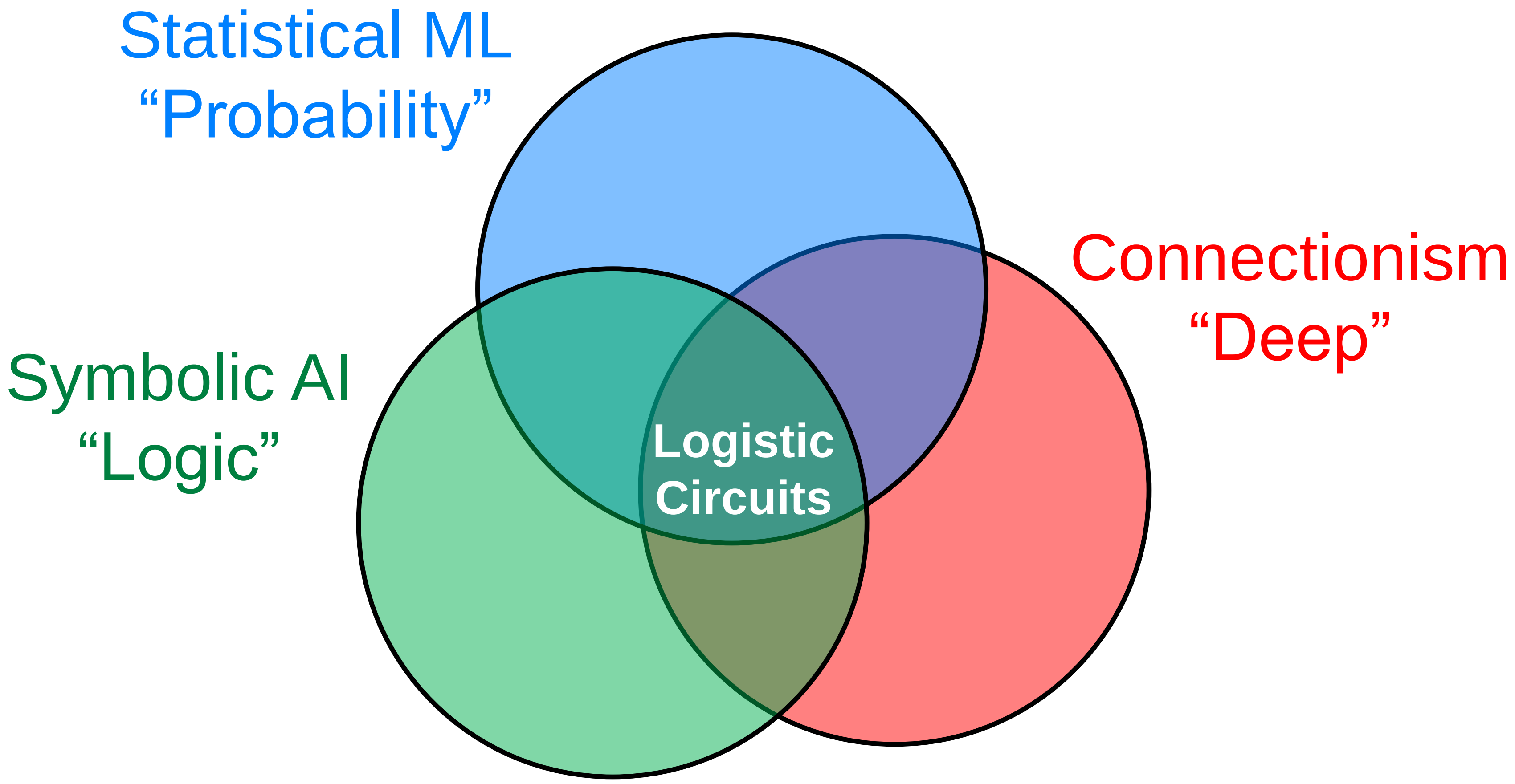


# Interpretable?

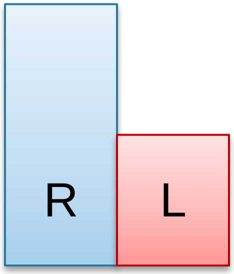




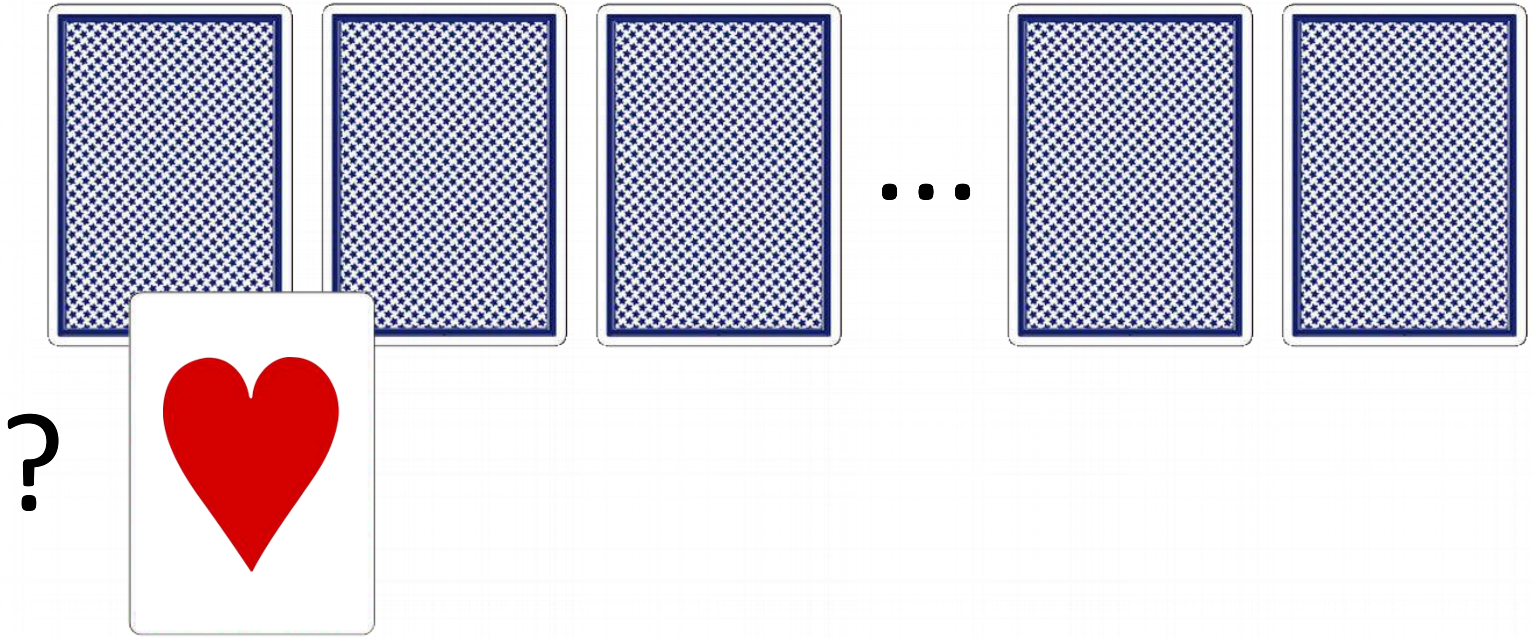
# Conclusions 2



# *High-Level Probabilistic Inference*



# Simple Reasoning Problem



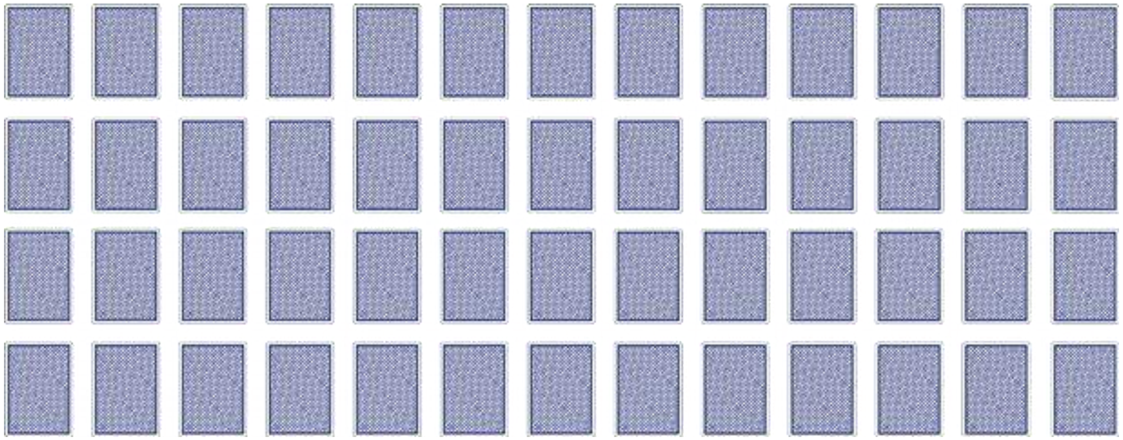

*Probability that Card1 is Hearts?* 1/4

[Van den Broeck; AAAI-KRR’15]

# Automated Reasoning

Let us automate this:

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



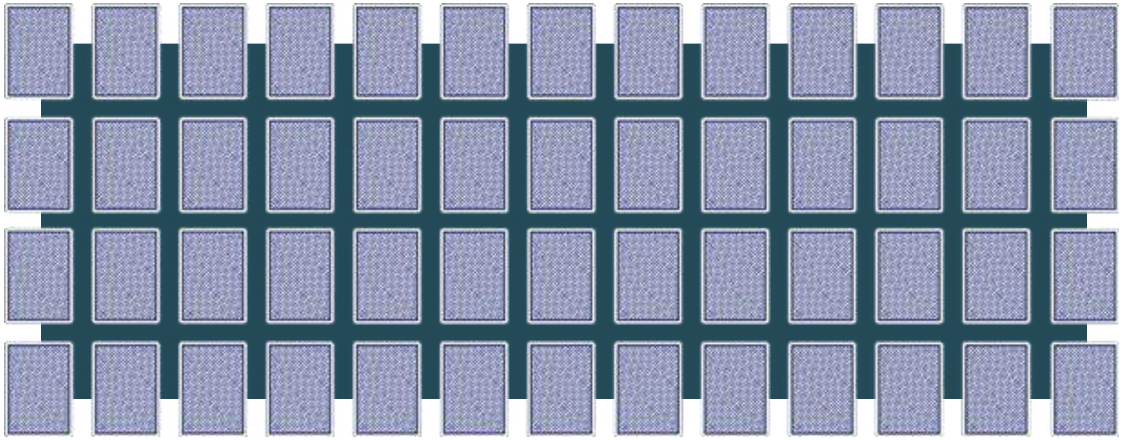
2. Probabilistic inference algorithm
   (e.g., variable elimination or junction tree)

[Van den Broeck; AAAI-KRR'15]

# Automated Reasoning

Let us automate this:

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

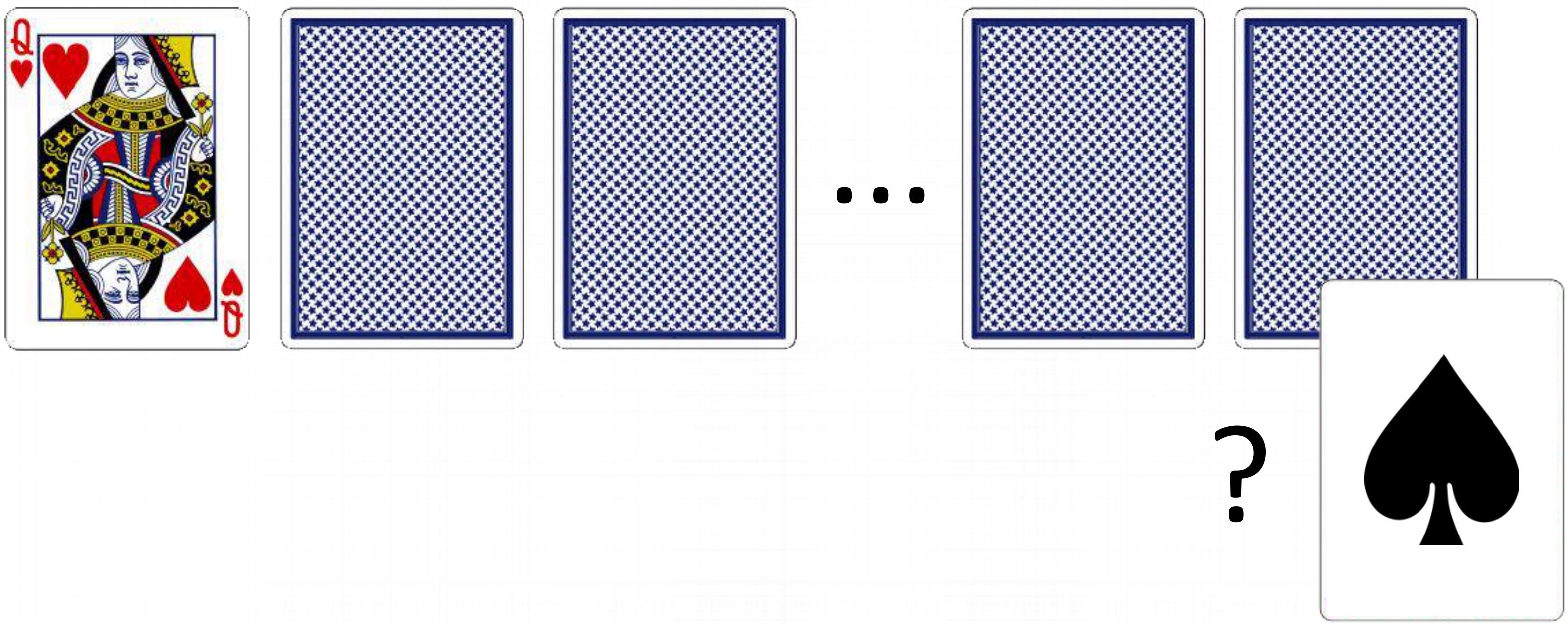


(artist's impression)

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

[Van den Broeck; AAAI-KRR'15]

# What's Going On Here?

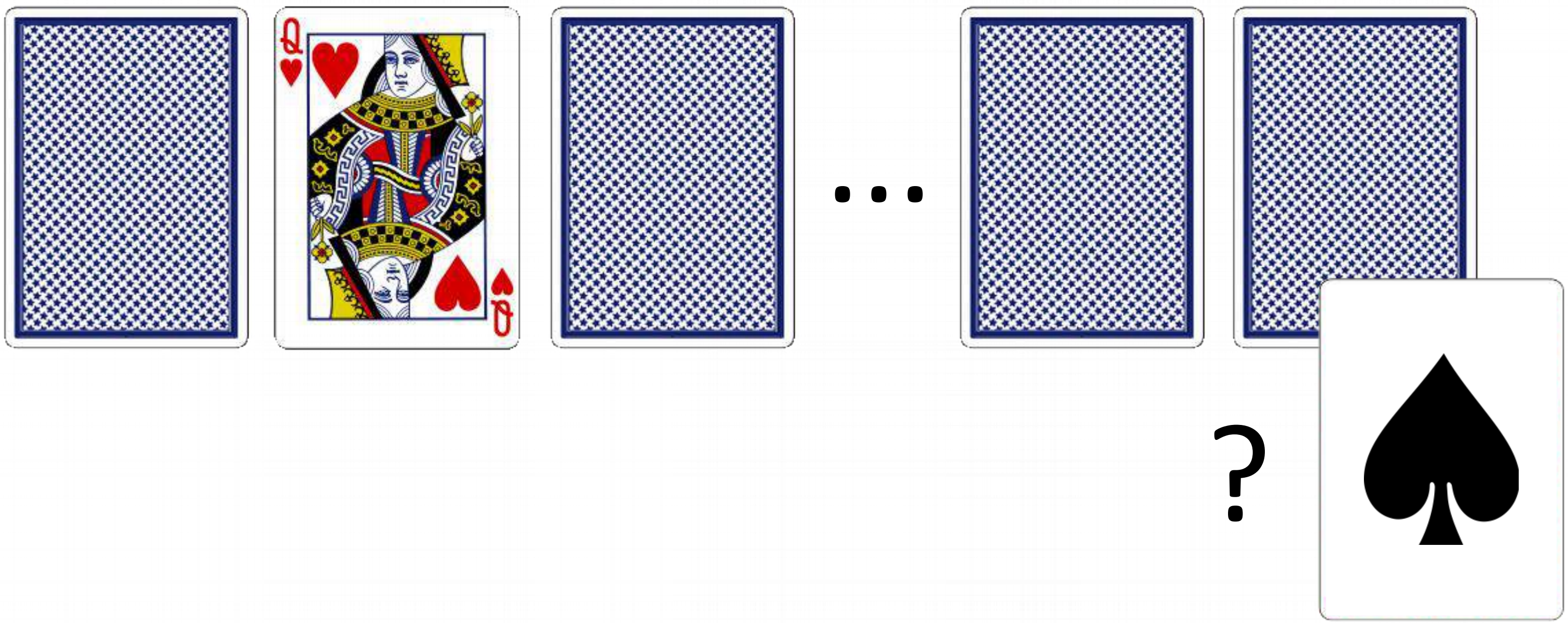



*Probability that Card52 is Spades given that Card1 is QH?*

13/51

[Van den Broeck; AAAI-KRR’15]

# What's Going On Here?

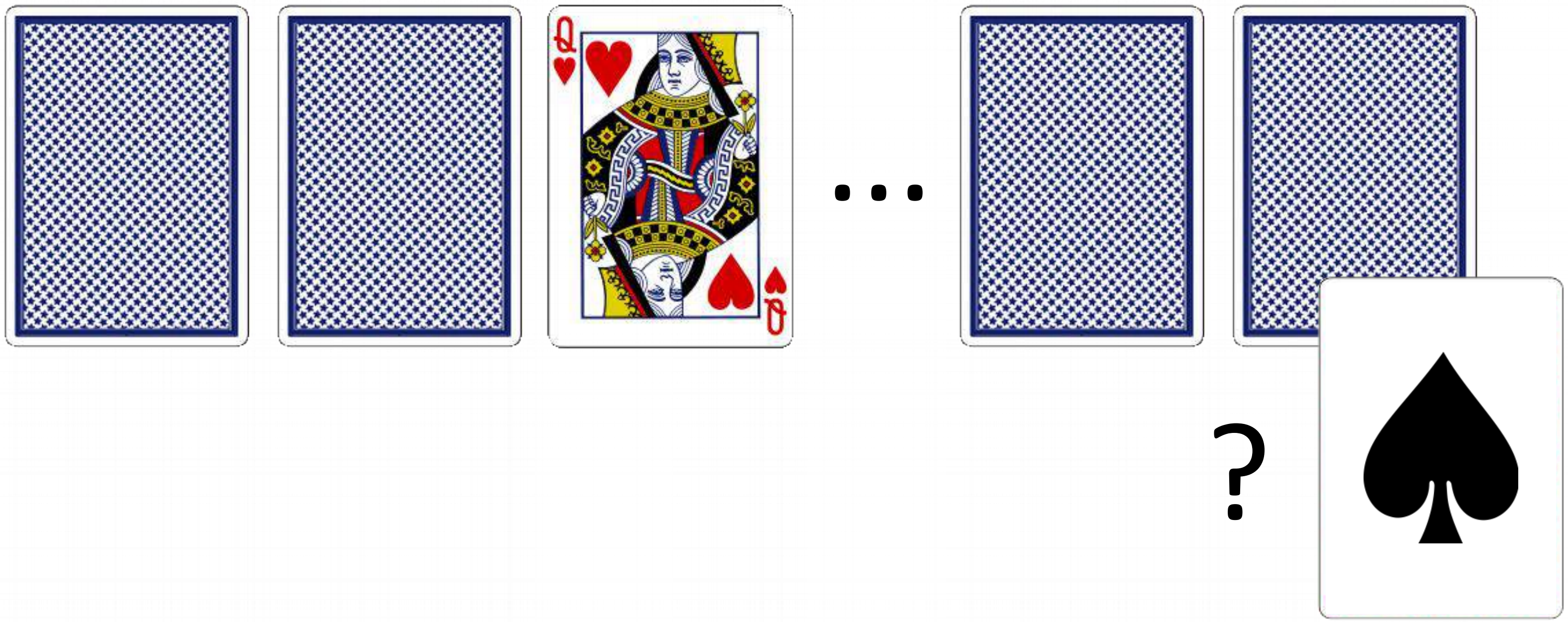



*Probability that Card52 is Spades given that Card2 is QH?*

13/51

[Van den Broeck; AAAI-KRR’15]

# What's Going On Here?

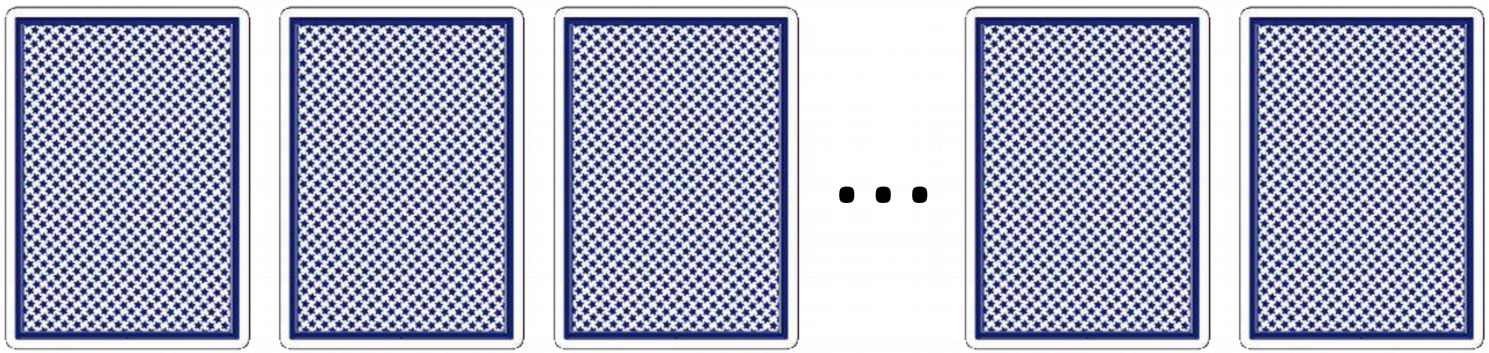



*Probability that Card52 is Spades given that Card3 is QH?*

13/51

[Van den Broeck; AAAI-KRR'15]

# Tractable Reasoning



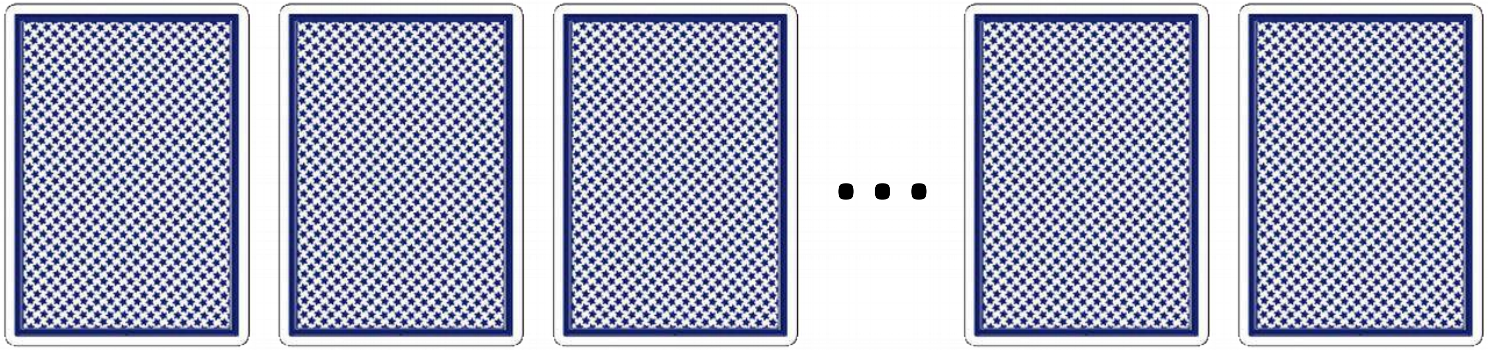

What's going on here?

Which property makes reasoning tractable?

- High-level (first-order) reasoning
- Symmetry
- Exchangeability

⇒ **Lifted Inference**

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



Model distribution at first-order level:

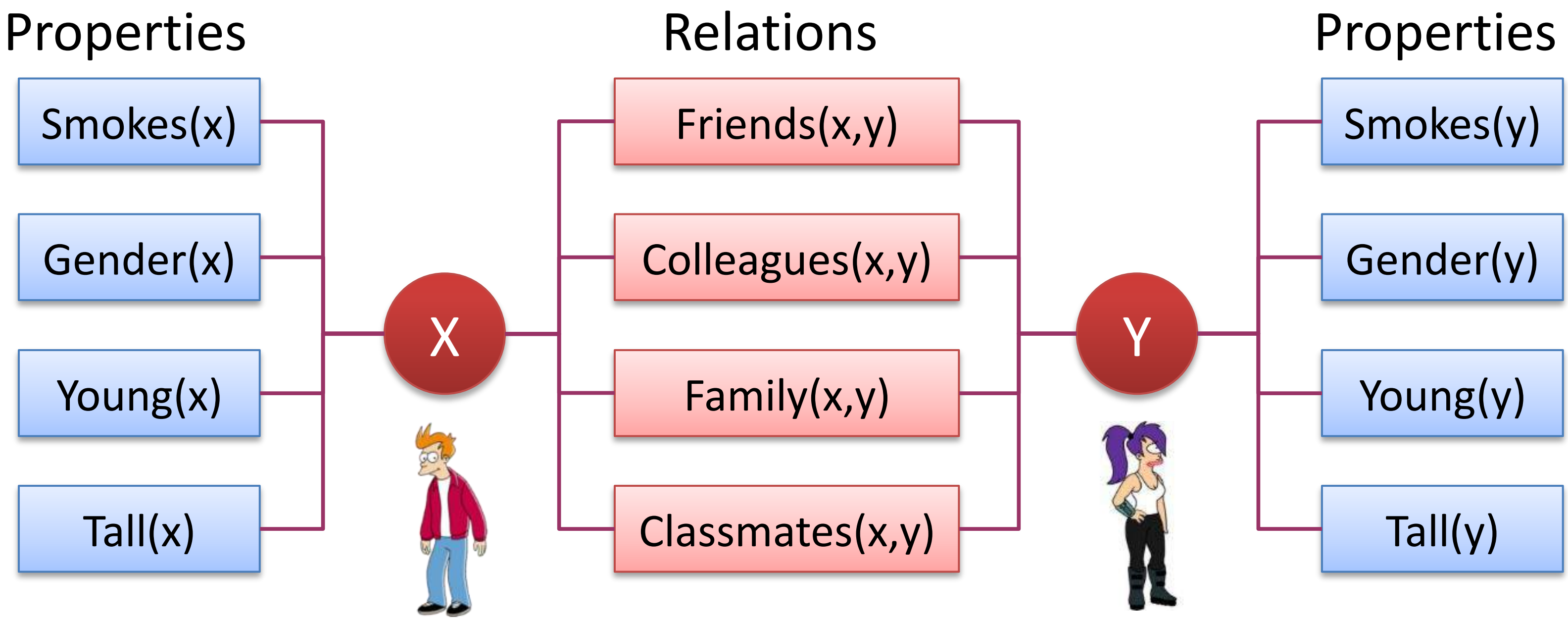
$\Delta =$

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

Can we now be efficient
in the size of our domain?

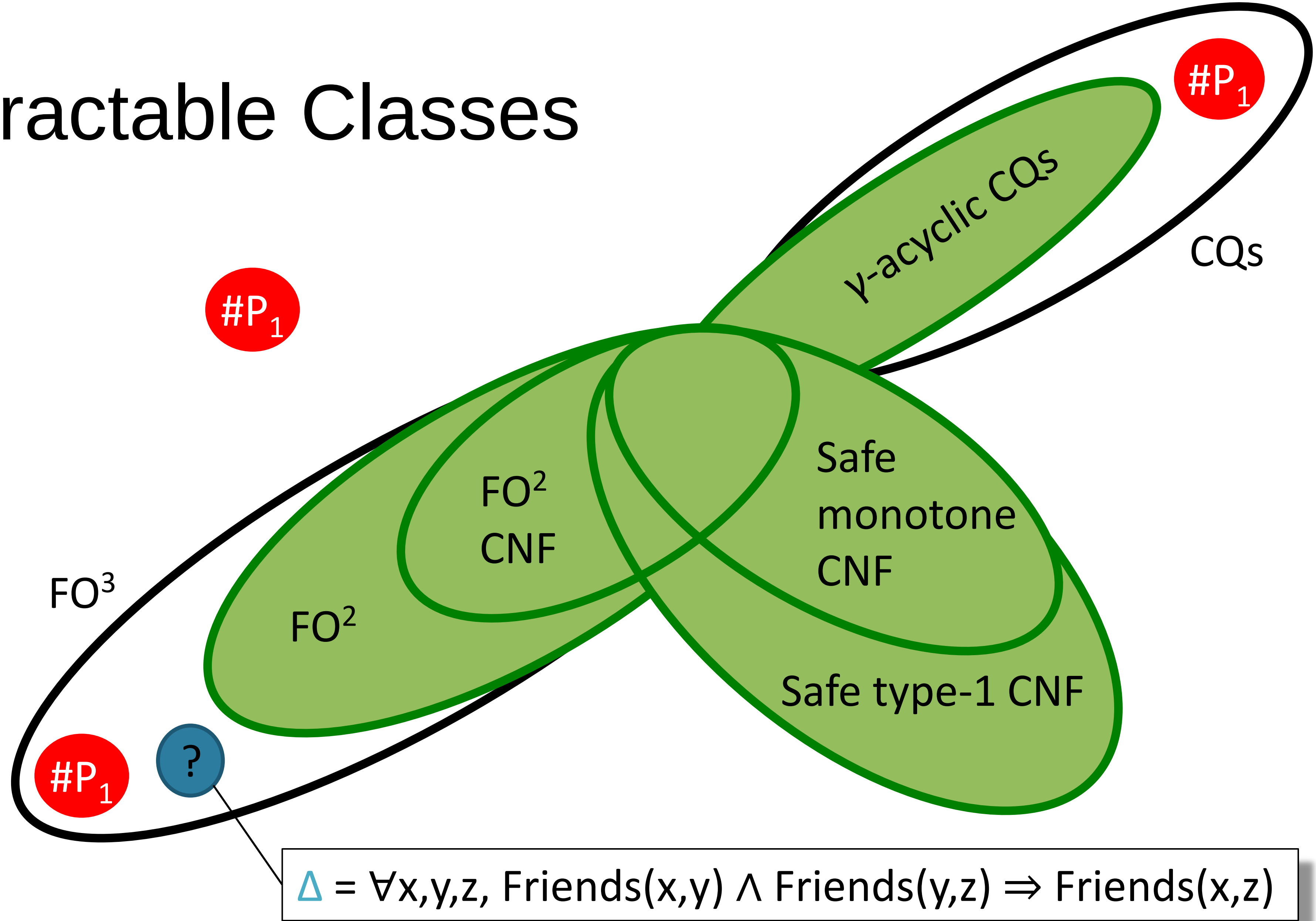
[Van den Broeck 2015]

# FO$^2$ is liftable!



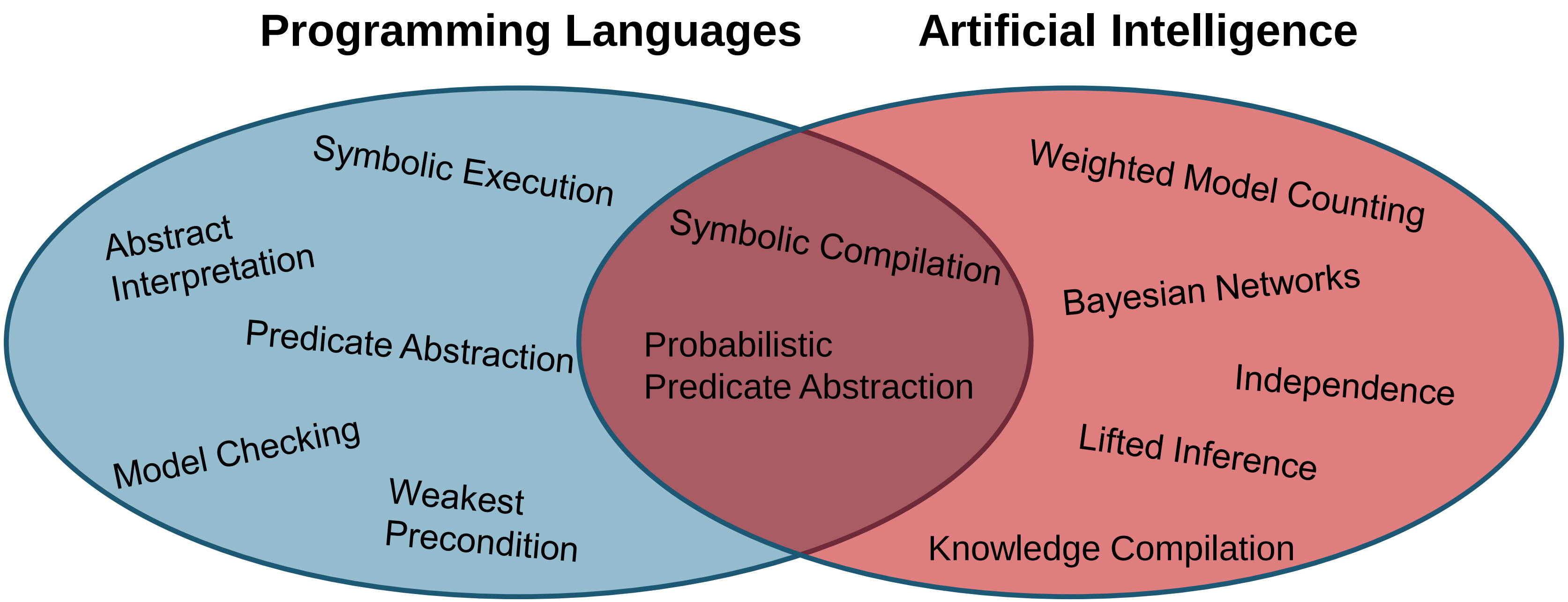

"Smokers are more likely to be friends with other smokers."
"Colleagues of the same age are more likely to be friends."
"People are either family or friends, but never both."
"If X is family of Y, then Y is also family of X."
"Universities in the Bay Area are more likely to be rivals."

# Tractable Classes




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

# Probabilistic Programming




Similar picture for *probabilistic databases*

# Conclusions 3

- Challenge is even greater at first-order level
- Existing reasoning algorithms cannot cut it!

- Integration of first-order logic and probability is long-standing goal of AI
- First-order probabilistic reasoning is **frontier** and **integration** of
  AI, KR, ML, DBs, theory, PL, etc.

# Final Conclusions

- Knowledge is everywhere in learning
- Some concepts not easily learned from data
- Make knowledge first-class citizen in ML

- Logical circuits turned statistical models
- Strong properties produce strong learners
- There is no dilemma between understanding and accuracy?

- A wealth of high-level reasoning approaches are still absent from ML discussion

# Acknowledgements

Thanks to my students and collaborators!

Thanks for your attention!

*Questions?*