technische universität dortmund

GIF German-Israeli Foundation for Scientific Research and Development

DFG

SEVENTH FRAMEWORK PROGRAMME

SFB 876 Verfügbarkeit von Information durch Analyse unter Ressourcenbeschränkung



# Declarative Data Science Machines




Kristian Kersting



Martin Mladenov TUD, Google



Sriraam Natarajan U. Indiana



Amir Globerson HUJI



Babak Ahmadi PicoEgo



Christopher Re Stanford



Martin Grohe RWTH Aachen



Christian Bauckhage U. Bonn



Pavel Tokmakov INRIA Grenoble

and many more ...

technische universität dortmund

GIF German-Israeli Foundation for Scientific Research and Development

DFG

SEVENTH FRAMEWORK PROGRAMME

SFB 876 Verfügbarkeit von Information durch Analyse unter Ressourcenbeschränkung



# What about "-O" flags for Data Science Machines



Kristian Kersting


Martin Mladenov
TUD, Google


Sriraam Natarajan
U. Indiana


Amir Globerson
HUJI


Babak Ahmadi
PicoEgo


Christopher Re
Stanford


Martin Grohe
RWTH Aachen


Christian Bauckhage
U. Bonn


Pavel Tokmakov
INRIA Grenoble

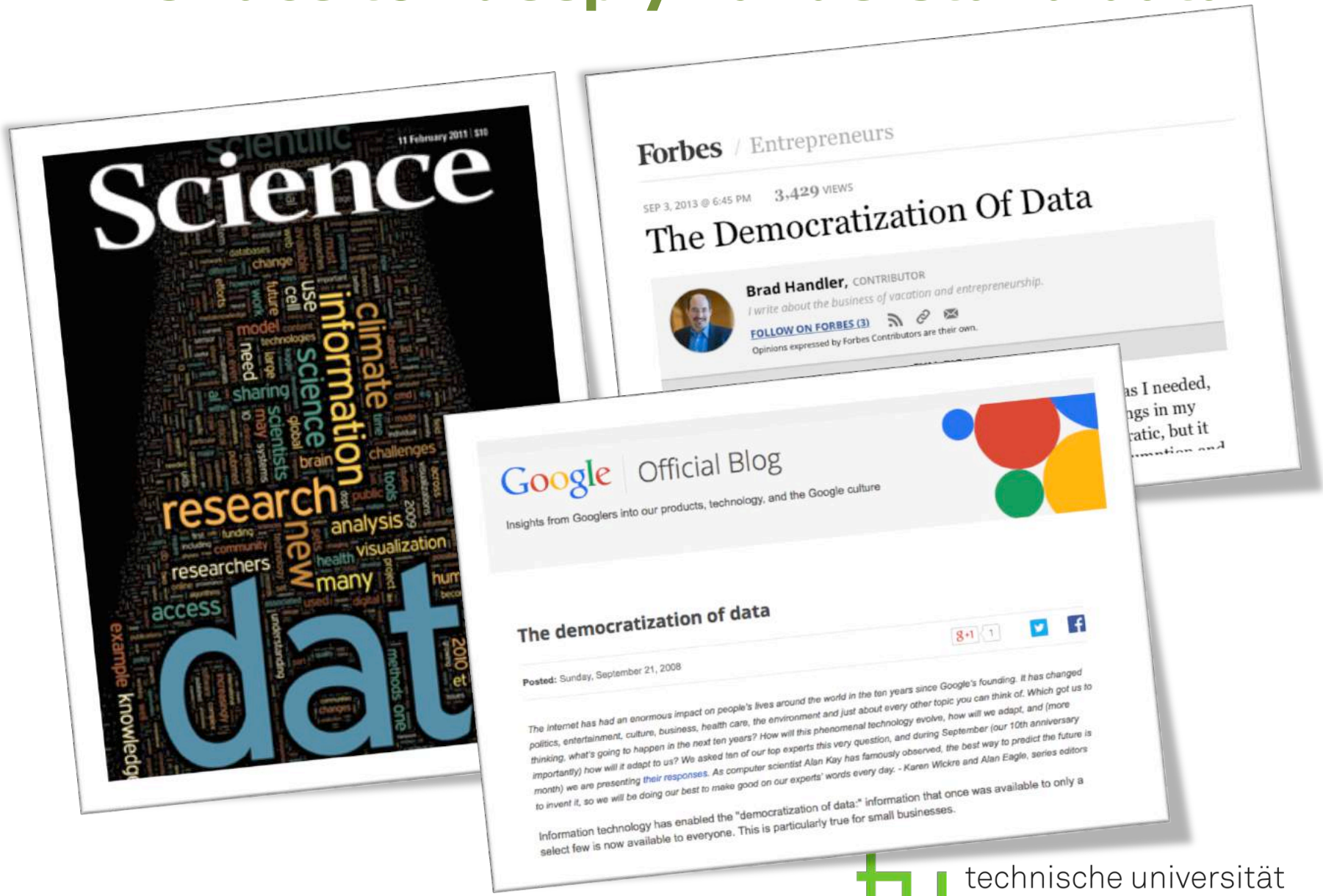
and many more ...





technische universität
dortmund

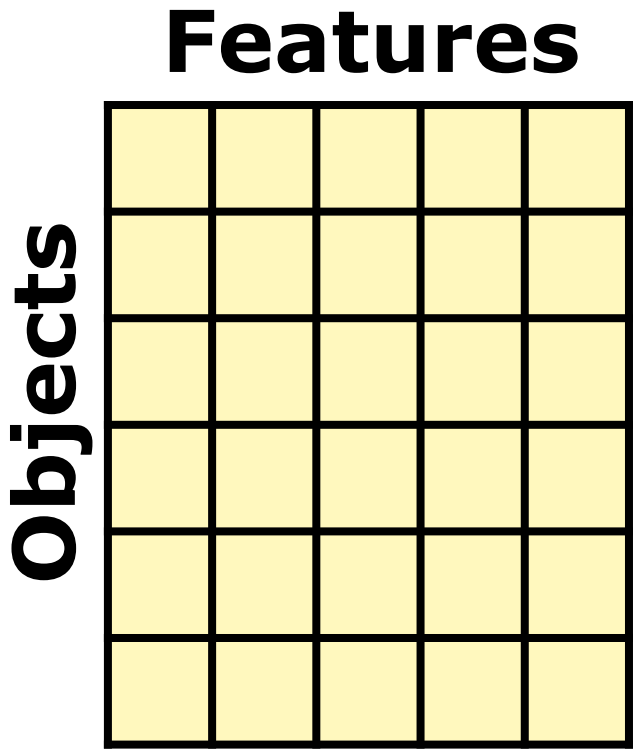
# Arms race to "deeply" understand data



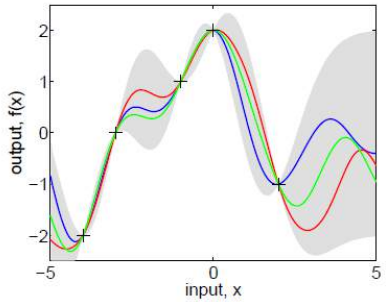



technische universität dortmund

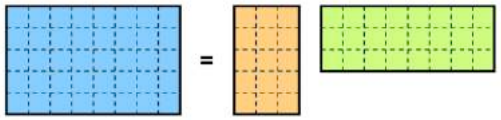
# Take your spreadsheet ...



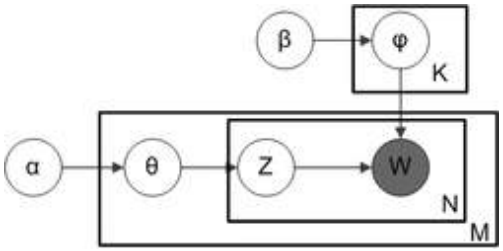
# … and apply Machine Learning



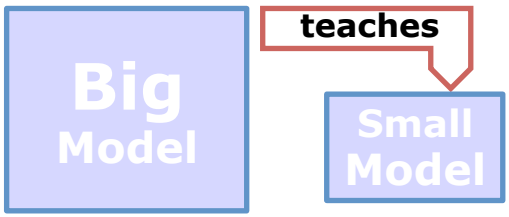

**Gaussian Processes**



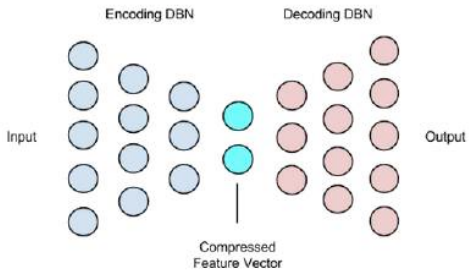
**Big Data Matrix Factorization**



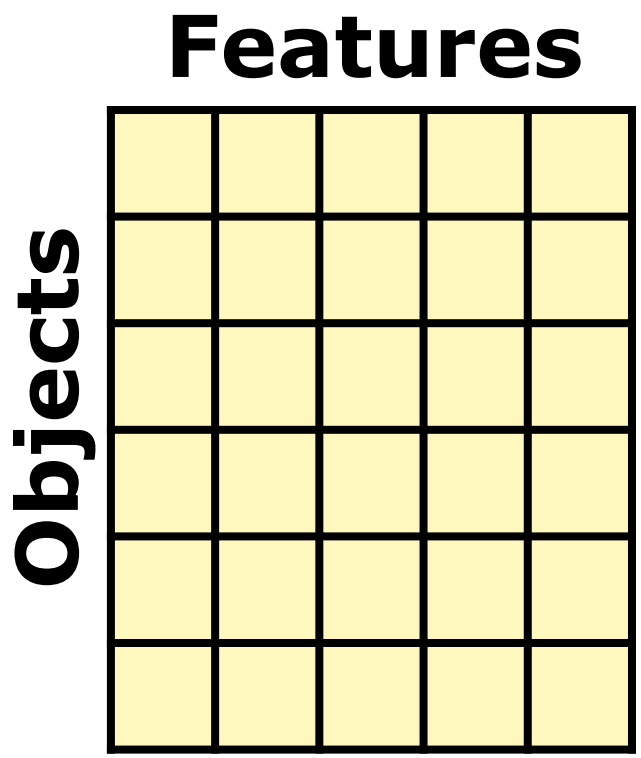

**Latent Dirichlet Allocation**



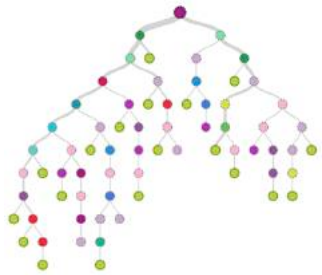

**Distillation/LUPI**



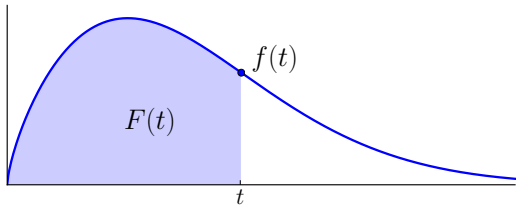

**Autoencoder/Deep Learning**




**and many more …**



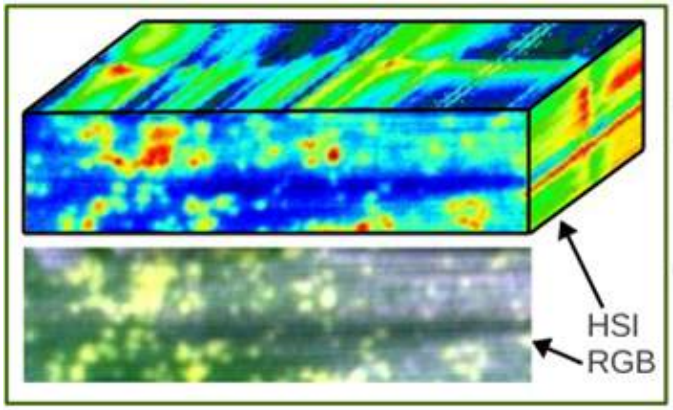
**Decision Trees/Boosting**

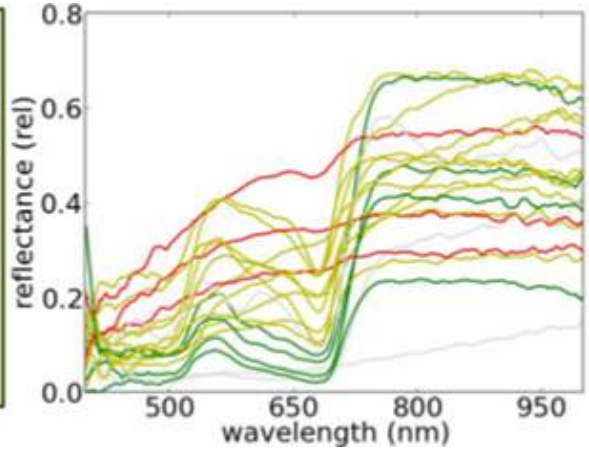



**Diffusion Models**



technische universität dortmund




Not only Big on data but also on interpretability

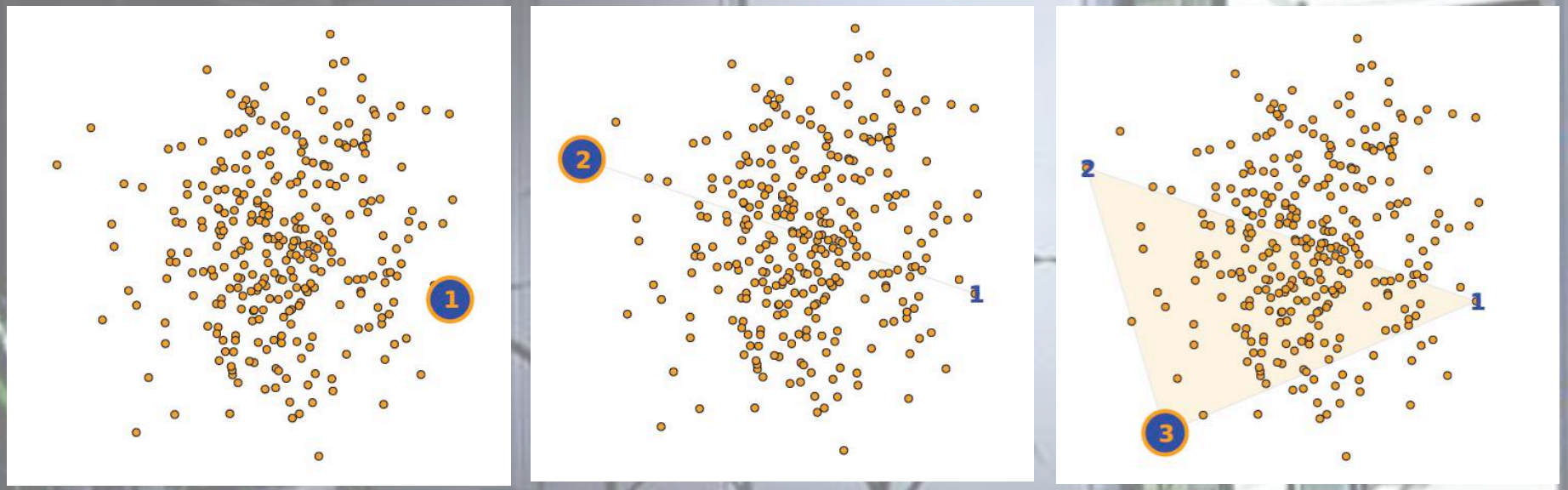
# Plant Phenotyping

# IS IT REALLY THAT SIMPLE?


What is the biological meaning of an eigenvector?

[Thurau, Kersting, Bauckhage, DAMI 2012]

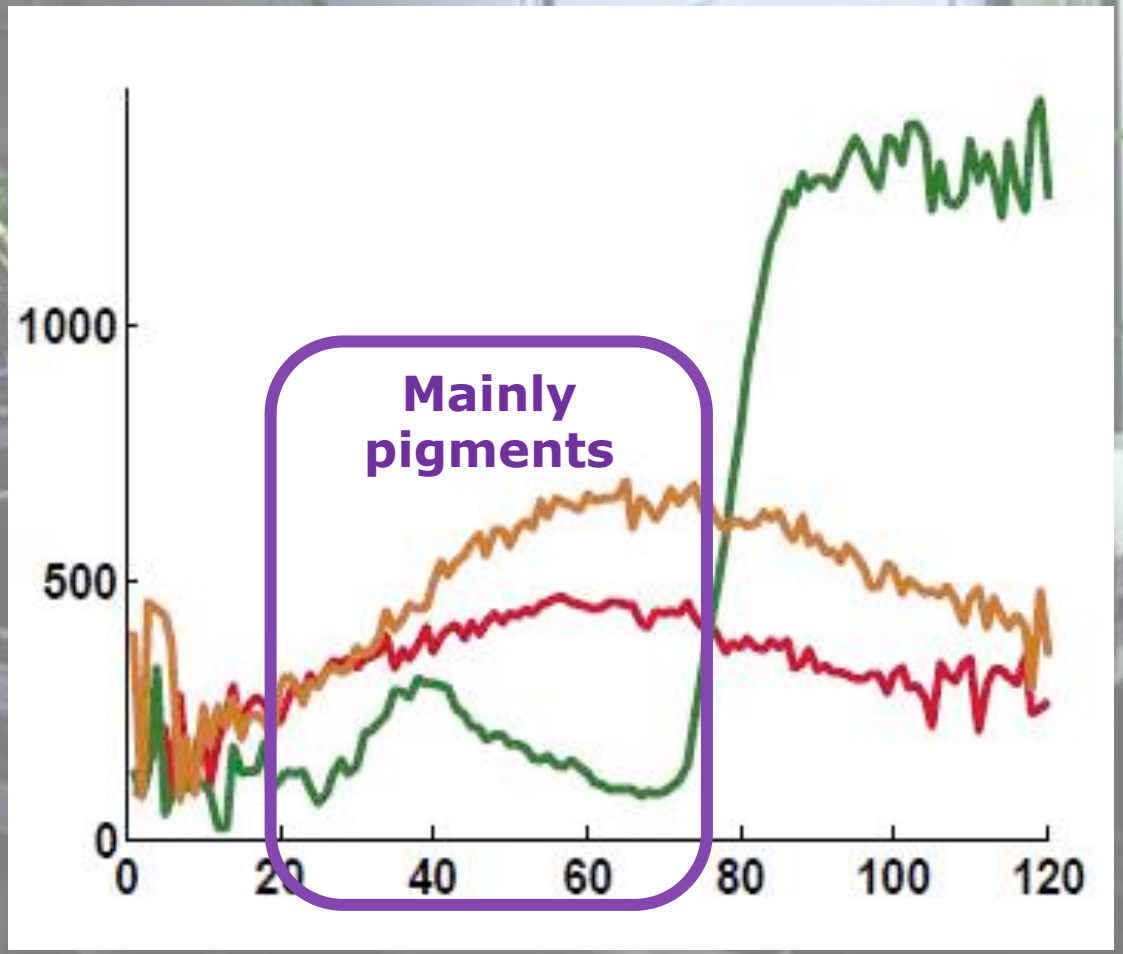
# Simplex Volume Maximization

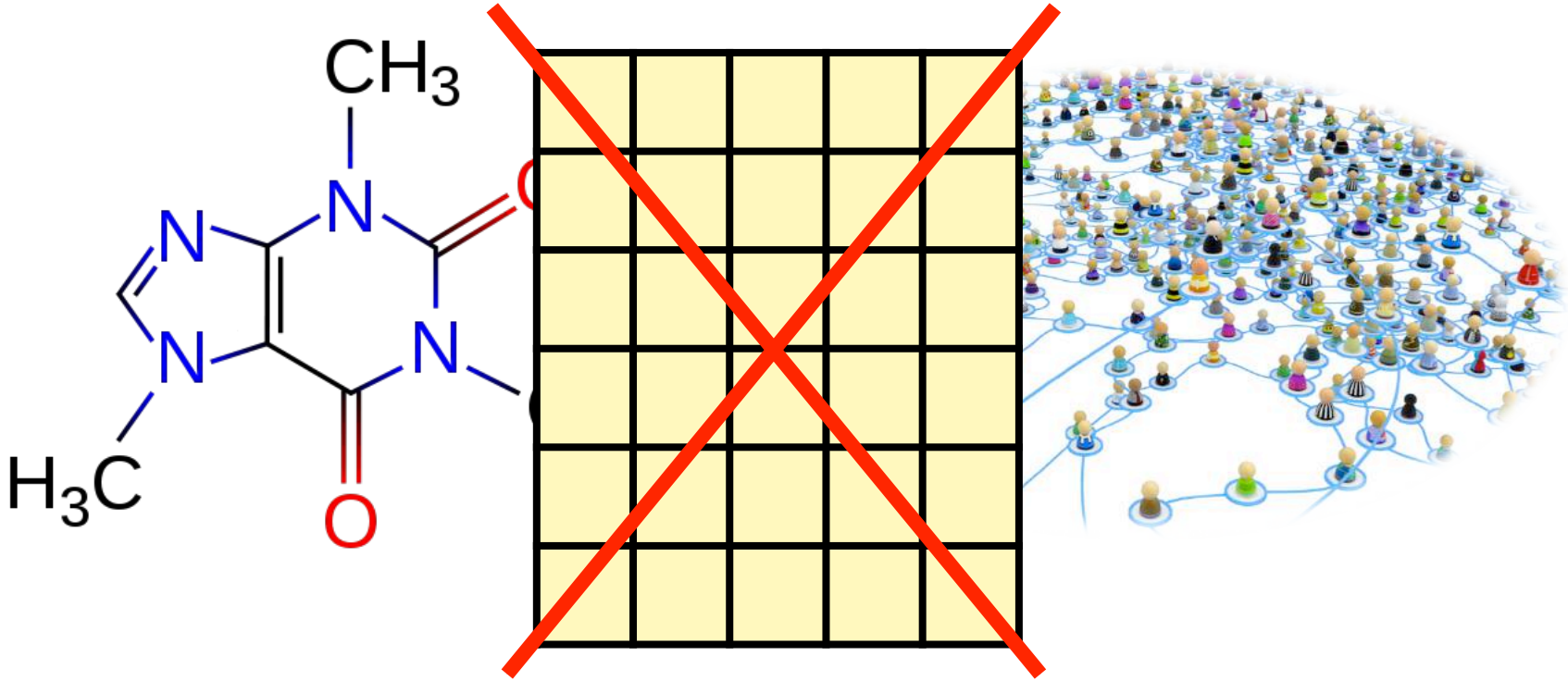


[Thurau, Kersting, Bauckhage, DAMI 2012]

# Simplex Volume Maximization

[Römer et al. Functional Plant Biology 2012, Wahabzada et al. PlosOne 2015, Wahabzada et al. Scientific Reports (Nature) 2016; Leuker et al. Functional Plant Biology 2016]






Several statistics used to characterize graphs: Degree distribution, average path length, diameter, cluster coefficients, ...

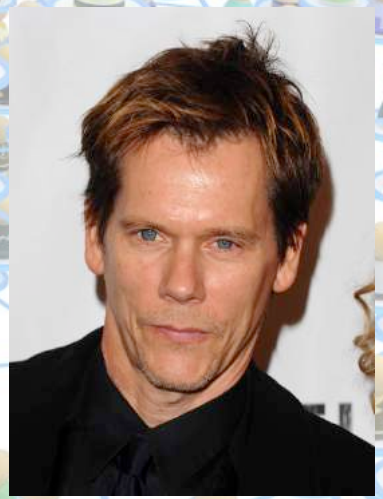
# IS IT REALLY THAT SIMPLE?

# MY IT'S A (SMALL) WORLD

## The (six-)degree of separation



S. Milgram
Psychology Today 2:60-67, 1967



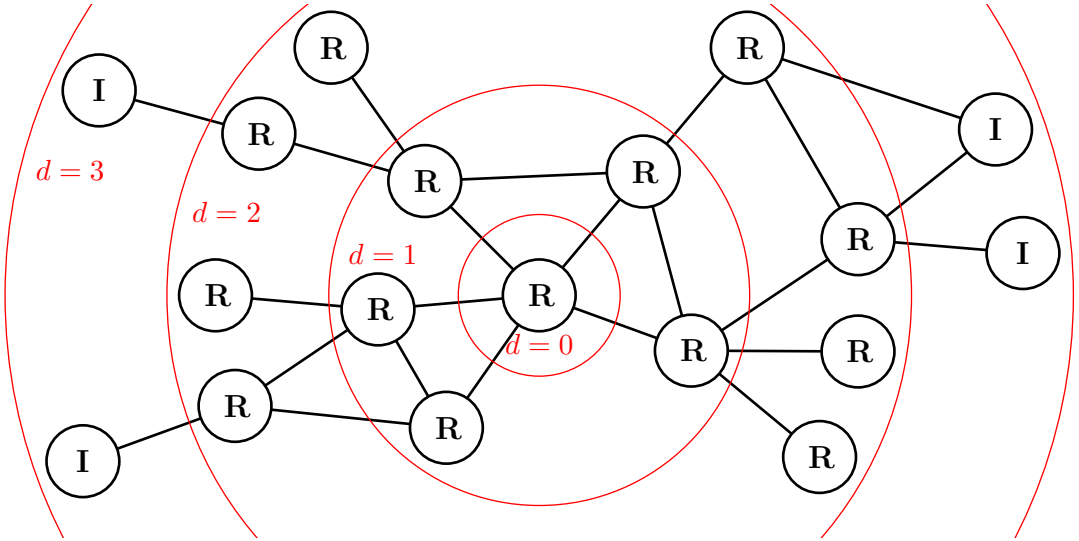
K. Bacon

[Bauckhage, **Kersting**, Hadiji UAI 2015]

# MY IT'S A (SMALL) WORLD

## The (six-)degree of separation is the mean of a generalized Gamma Distribution

$$f_{GG}(t \mid \omega, \alpha, \beta) = \frac{\beta}{\omega^{\alpha}} \frac{1}{\Gamma(\alpha/\beta)} t^{\alpha-1} \exp\left\{-\left(\frac{t}{\omega}\right)^{\beta}\right\}$$



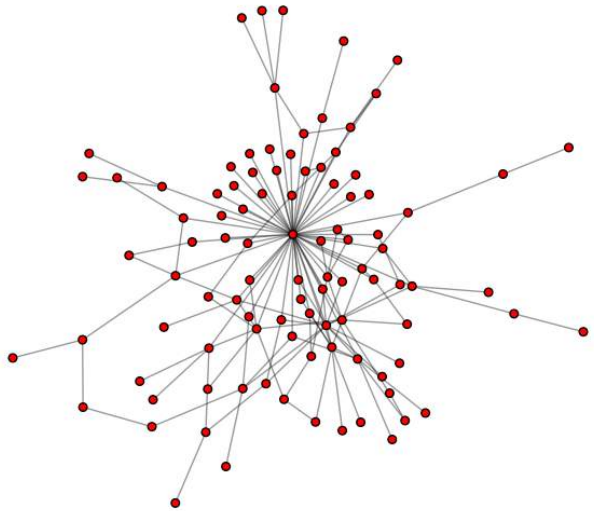

Proof: SIR-model, multinomial over histogram of distance, use Sterling's formula to turn into maxent, impose constraints such that polynomial reachability + finite moments

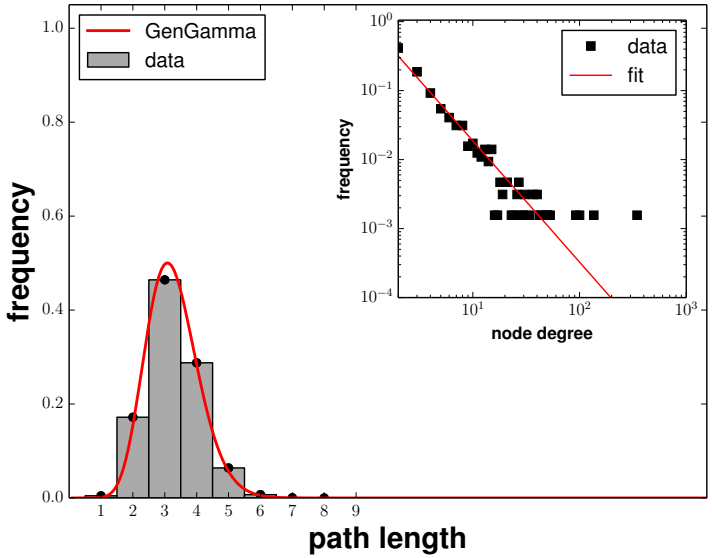
[Bauckhage, **Kersting**, Hadiji UAI 2015]

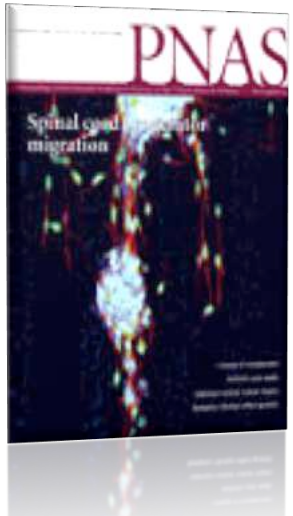
# MY IT'S A (SMALL) WORLD

## The (six-)degree of separation is the mean of a generalized Gamma Distribution

$$f_{GG}(t \mid \omega, \alpha, \beta) = \frac{\beta}{\omega^{\alpha}} \frac{1}{\Gamma(\alpha/\beta)} t^{\alpha-1} \exp\left\{-\left(\frac{t}{\omega}\right)^{\beta}\right\}$$








**„The subject of collective attention is central to an information age where millions of people are inundated with daily messages. “**

**- Wu and Huberman, PNAS, 104(45), 2007**







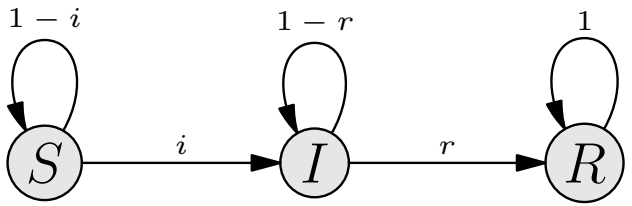



technische universität dortmund

Bauckhage, Kersting, Hadiji ICWSM 2015

# Can YouTube videos really go viral?

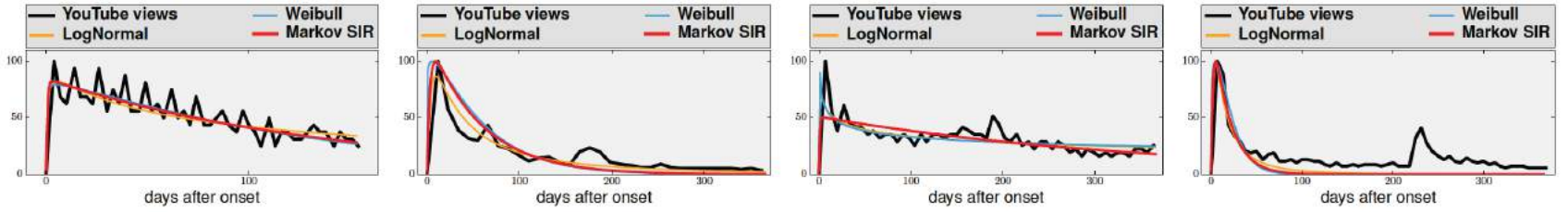
## Yes, they are ! Collective attention to YouTube videos follows an epidemic model




**Closed-from density of two convolved exponential distributions**

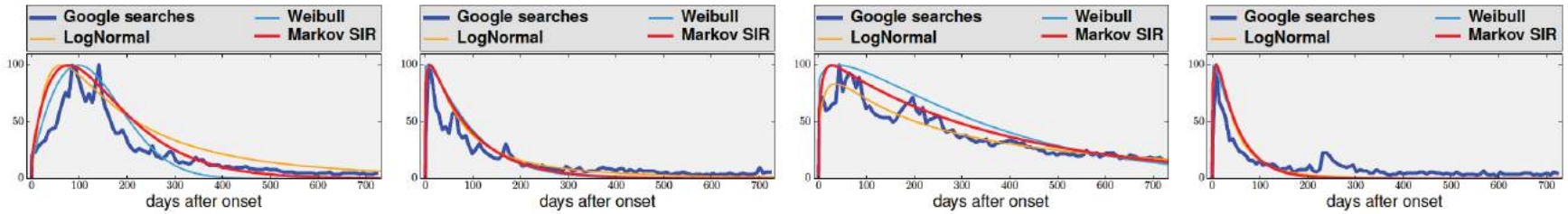
$$f(t) = \begin{cases} \frac{\alpha}{\alpha-\lambda}\,\lambda\,e^{-\lambda t} + \frac{\lambda}{\lambda-\alpha}\,\alpha\,e^{-\alpha t} & \text{if } \lambda \neq \alpha \\ \lambda^2\,t\,e^{-\lambda t} & \text{if } \lambda = \alpha. \end{cases}$$

YoutTube View Counts




(a) call me maybe (b) double rainbow (c) nyan cat (d) ultimate do tease

Google Trend Search Counts




(a) call me maybe (b) double rainbow (c) nyan cat (d) ultimate do tease

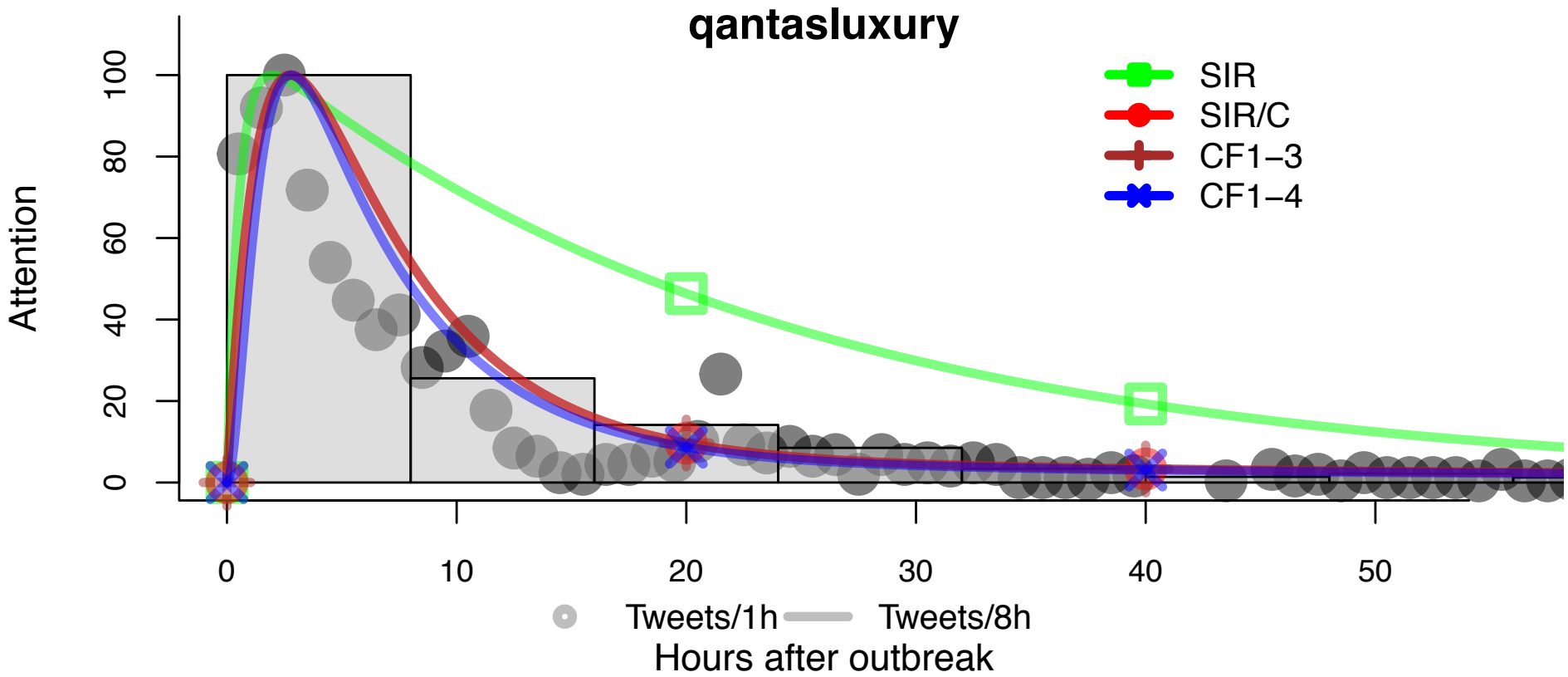
## Unsettledness in Politics and Business

# Let’s better not say anything. Otherwise, we will have an online firestorm tomorrow



reuters.com/article/2011/11/22/us-qantas-idUSTRE7AL0HB20111122

# Just a day after Qantas and its unions broke off contract negotiations and after Qantas grounded its fleet in late October, Qantas invited users to enter a "Qantas Luxury" competition, asking people to describe their "dream luxury in-flight experience”

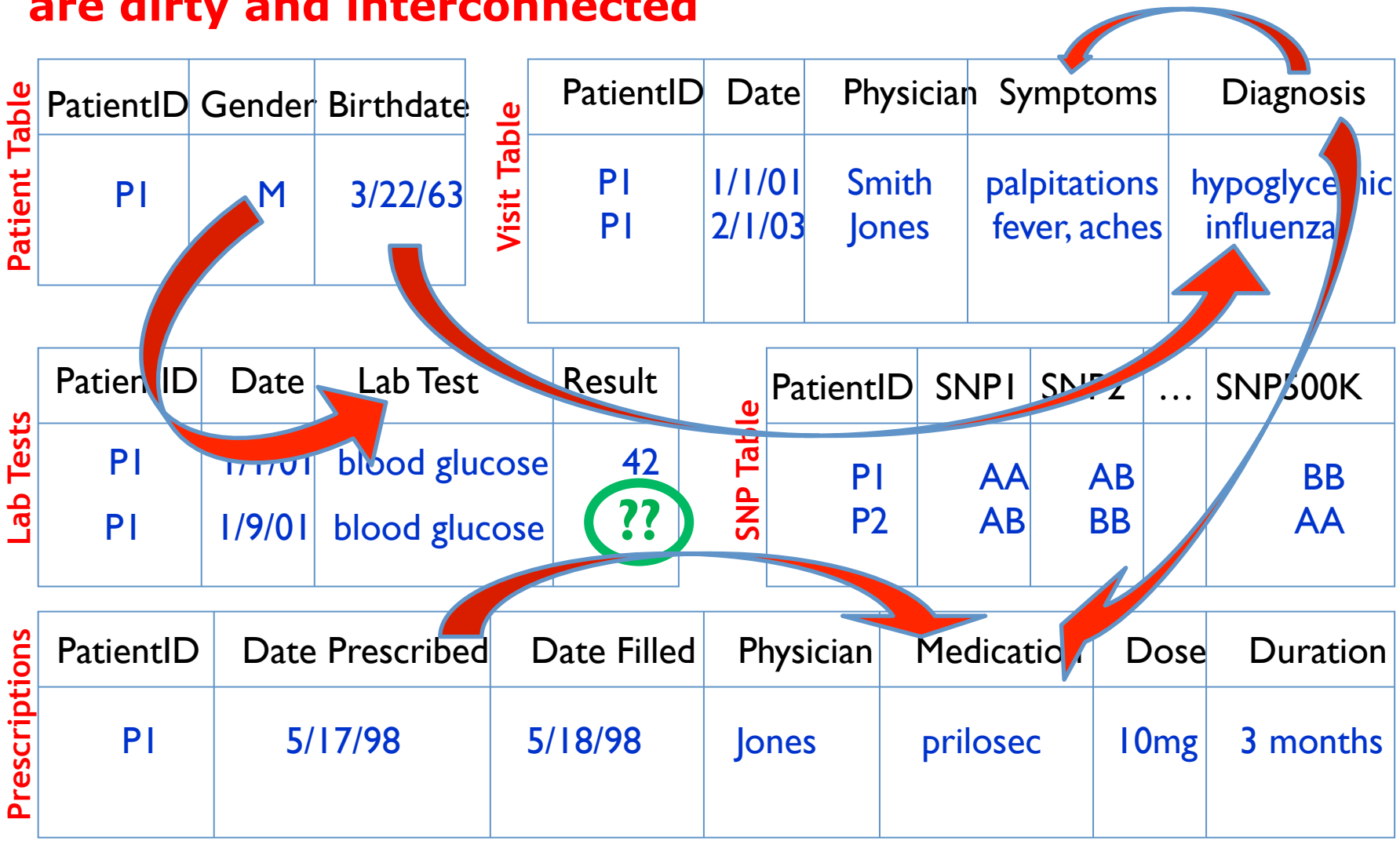



Kersting, Bauckhage, Köcher, Swazinna, Pfeffer 2016

# IS IT REALLY THAT SIMPLE?

# Actually, most data in the world are stored in relational DBs

**E.g. mining Electronic Health Records are an opportunities to save our lifes and a lot of money. Unfortunately, they are dirty and interconnected**



Patient Table

| PatientID | Gender | Birthdate |
|---|---|---|
| P1 | M | 3/22/63 |

Visit Table

| PatientID | Date | Physician | Symptoms | Diagnosis |
|---|---|---|---|---|
| P1 | 1/1/01 | Smith | palpitations | hypoglyce[illegible]ic |
| P1 | 2/1/03 | Jones | fever, aches | influenza |

Lab Tests

| Patien[illegible]ID | Date | Lab Test | Result |
|---|---|---|---|
| P1 | 1/1/01 | blood glucose | 42 |
| P1 | 1/9/01 | blood glucose | ?? |

SNP Table

| PatientID | SNP1 | SNP2 | … | SNP500K |
|---|---|---|---|---|
| P1 | AA | AB | | BB |
| P2 | AB | BB | | AA |

Prescriptions

| PatientID | Date Prescribed | Date Filled | Physician | Medication | Dose | Duration |
|---|---|---|---|---|---|---|
| P1 | 5/17/98 | 5/18/98 | Jones | prilosec | 10mg | 3 months |

[**Kersting**, Driessens ICML´08; Karwath, **Kersting**, Landwehr ICDM´08; Natarajan, Joshi, Tadepelli, **Kersting**, Shavlik. IJCAI´11; Khot, Natarajan, **Kersting**, Shavlik ICDM´13, MLJ´12, Springer Brief´15, MLJ´15]

# Relational Mining of EHRs

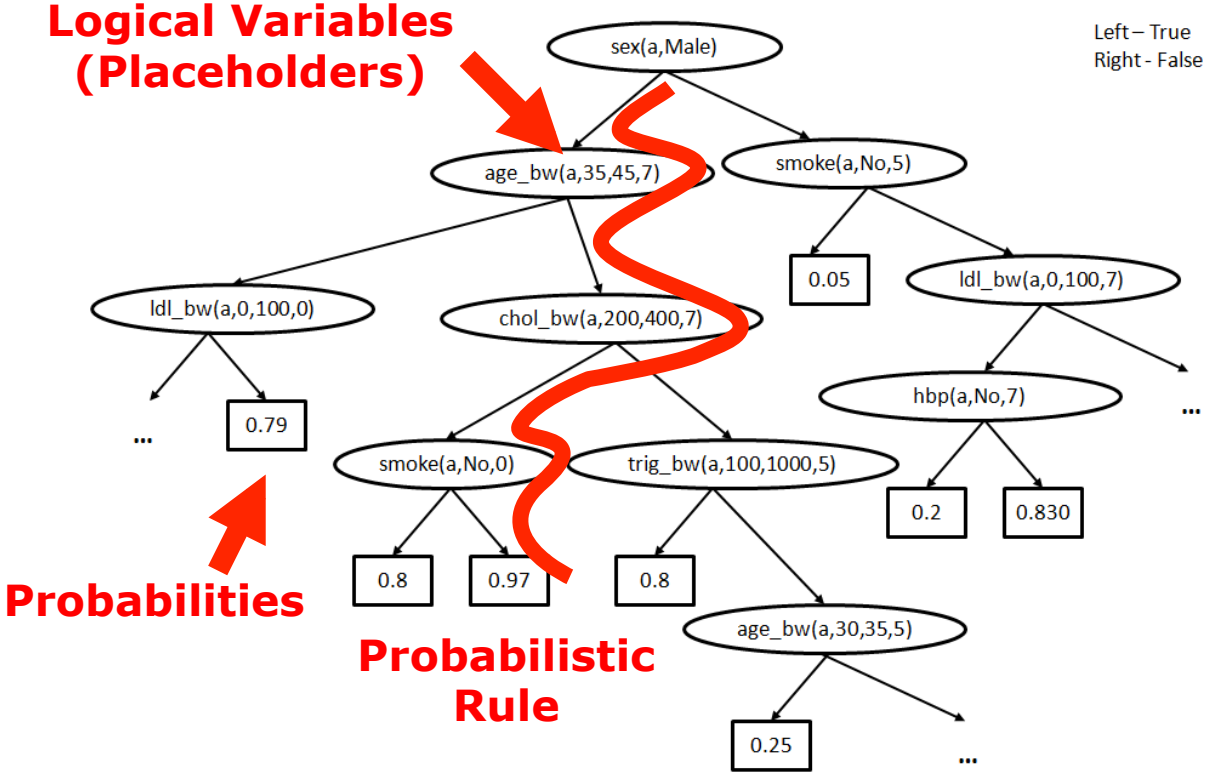
**Plaque in the left coronary artery**



[Circulation; 92(8), 2157-62, 1995; JACC; 43, 842-7, 2004]

## Atherosclerosis is the cause of the majority of Acute Myocardial Infarctions (heart attacks)

**Logical Variables (Placeholders)**



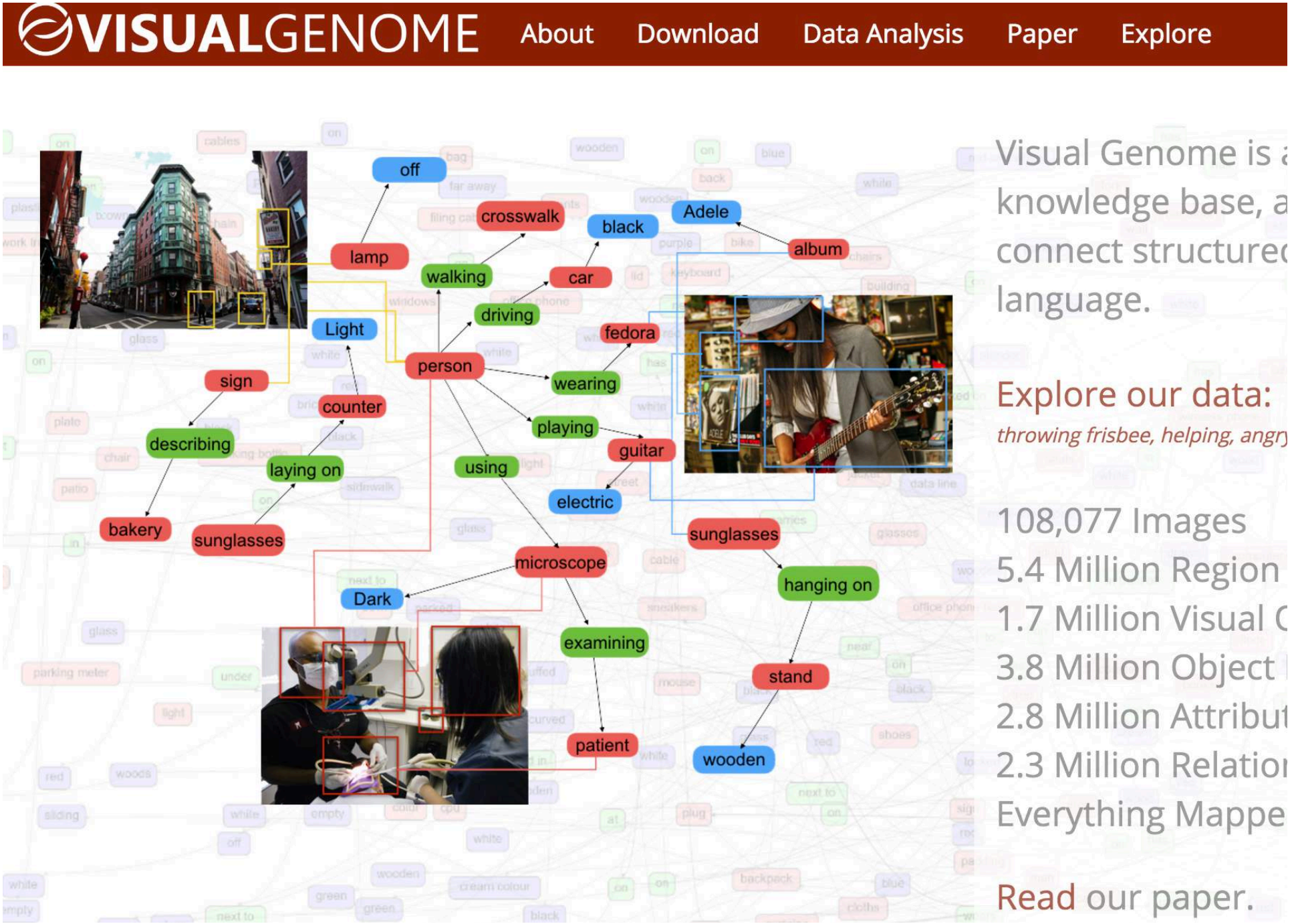

**Probabilities**

**Probabilistic Rule**

| Algorithm | Accuracy | AUC-ROC |
|---|---|---|
| J48 | 0.667 | 0.607 |
| SVM | 0.667 | 0.5 |
| AdaBoost | 0.667 | 0.608 |
| Bagging | 0.677 | 0.613 |
| NB | 0.75 | 0.653 |
| RPT | 0.669* | 0.778 |
| RFGB | 0.667* | 0.819 |

| Algo | Likelihood | AUC-ROC | AUC-PR | Time |
|---|---|---|---|---|
| **Boosting** | 0.810 | 0.961 | 0.930 | **9s** |
| **MLN** | 0.730 | 0.535 | 0.621 | **93 hrs** |

[Lu, Krishna, Bernstein, Fei-Fei „Visual Relationship Detection" CVPR 2016]



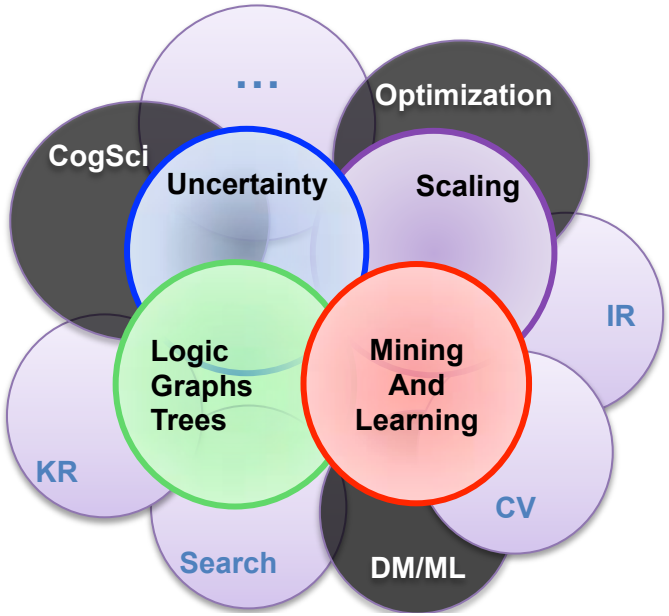
[Getoor, Taskar MIT Press '07; De Raedt, Frasconi, **Kersting**, Muggleton, LNCS'08; Domingos, Lowd Morgan Claypool '09; Natarajan, **Kersting**, Khot, Shavlik Springer Brief'15; Russell CACM 58(7): 88-97 '15]




De Raedt, **Kersting**, Natarajan, Poole, Statistical Relational Artificial Intelligence: Logic, Probability, and Computation. Morgan and Claypool Publishers, 2016.

# Statistical Relational Learning/AI Mining Probabilistic DBs

… **the study and design of intelligent agents that act in noisy worlds composed of objects and relations among the objects**




## And this had major impact on CogSci



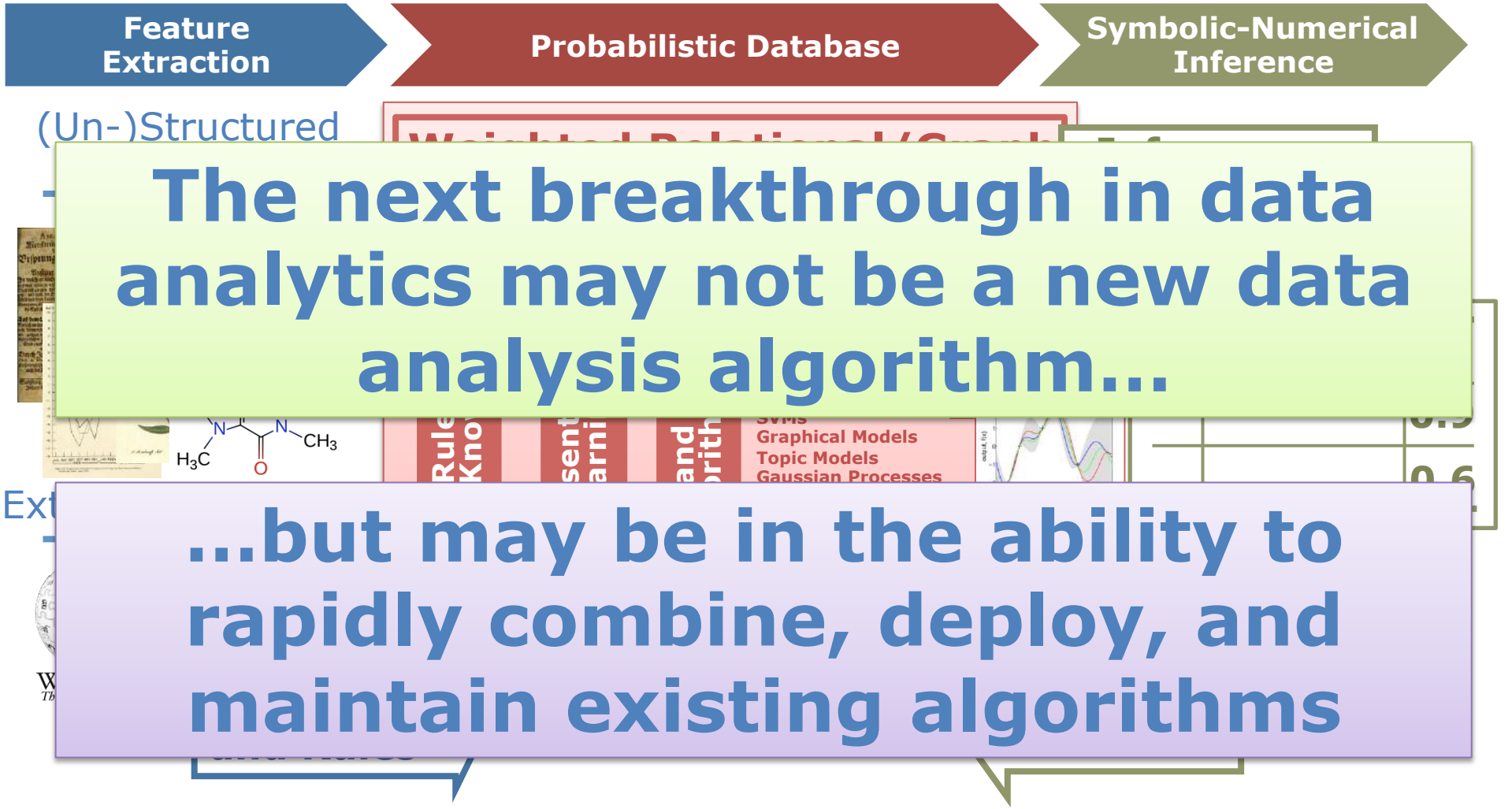




Lake, Salakhutdinov, Tenenbaum, Science 350 (6266), 1332-1338, 2015
Tenenbaum, Kemp, Griffiths, Goodman, Science 331 (6022), 1279-1285, 2011

[Ré, Sadeghian, Shan, Shin, Wang, Wu, Zhang IEEE Data Eng. Bull.'14; Natarajan, Picado, Khot, **Kersting**, Ré, Shavlik ILP'14; Natarajan, Soni, Wazalwar, Viswanathan, **Kersting** Solving Large Scale Learning Tasks'16, Mladenov, Heinrich, Kleinhans, Gonsior, **Kersting** DeLBP'16, …]

# Declarative Data Science Machines






technische universität dortmund




# BUT WHAT DOES "RAPIDLY COMBINE AND DEPLOY" MEAN?



technische universität dortmund

# A simple example



Guy van den Broeck
UCLA

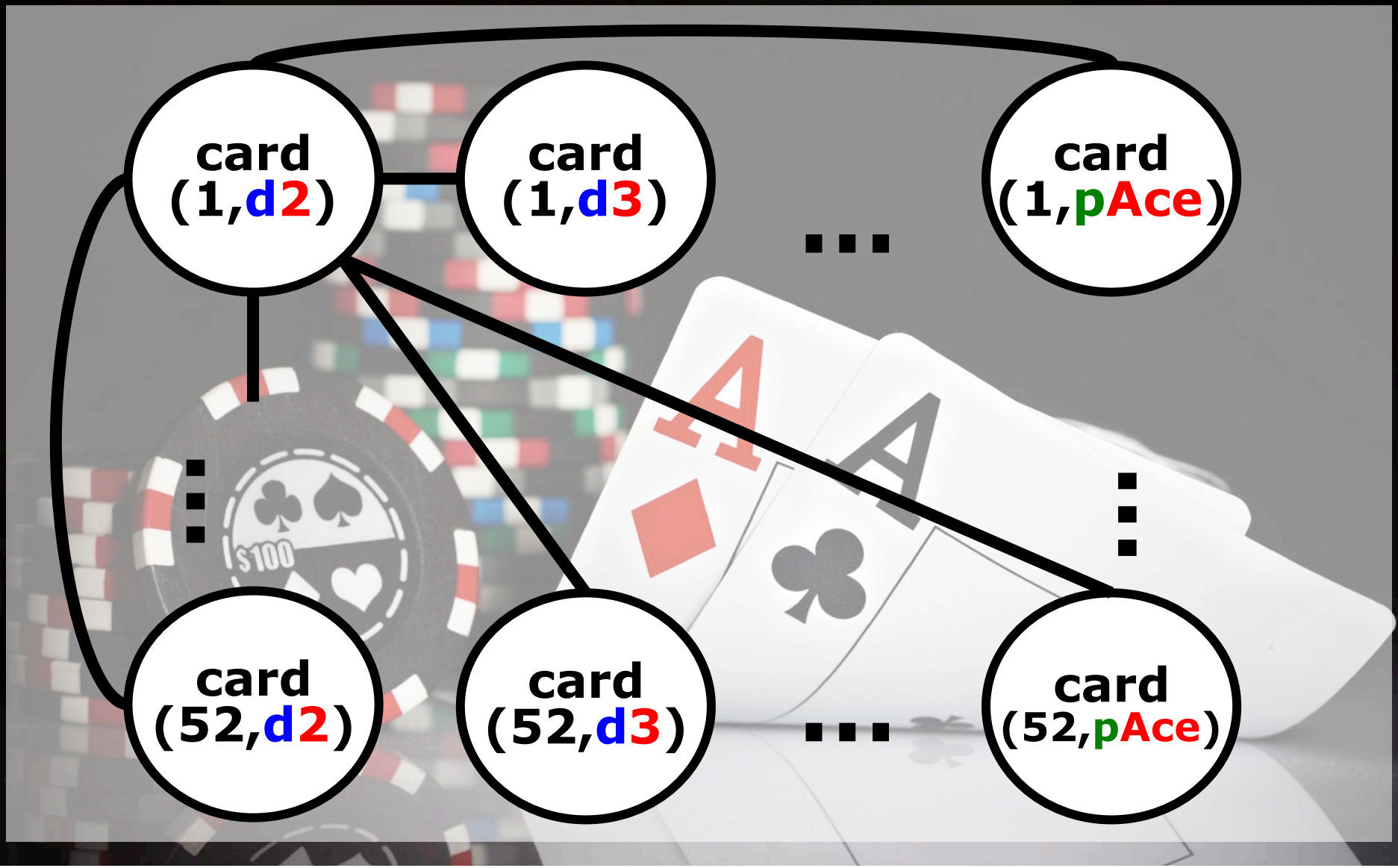
**What is the probability that the first card of a randomly shuffled deck with 52 cards is an Ace?**



# A simple example



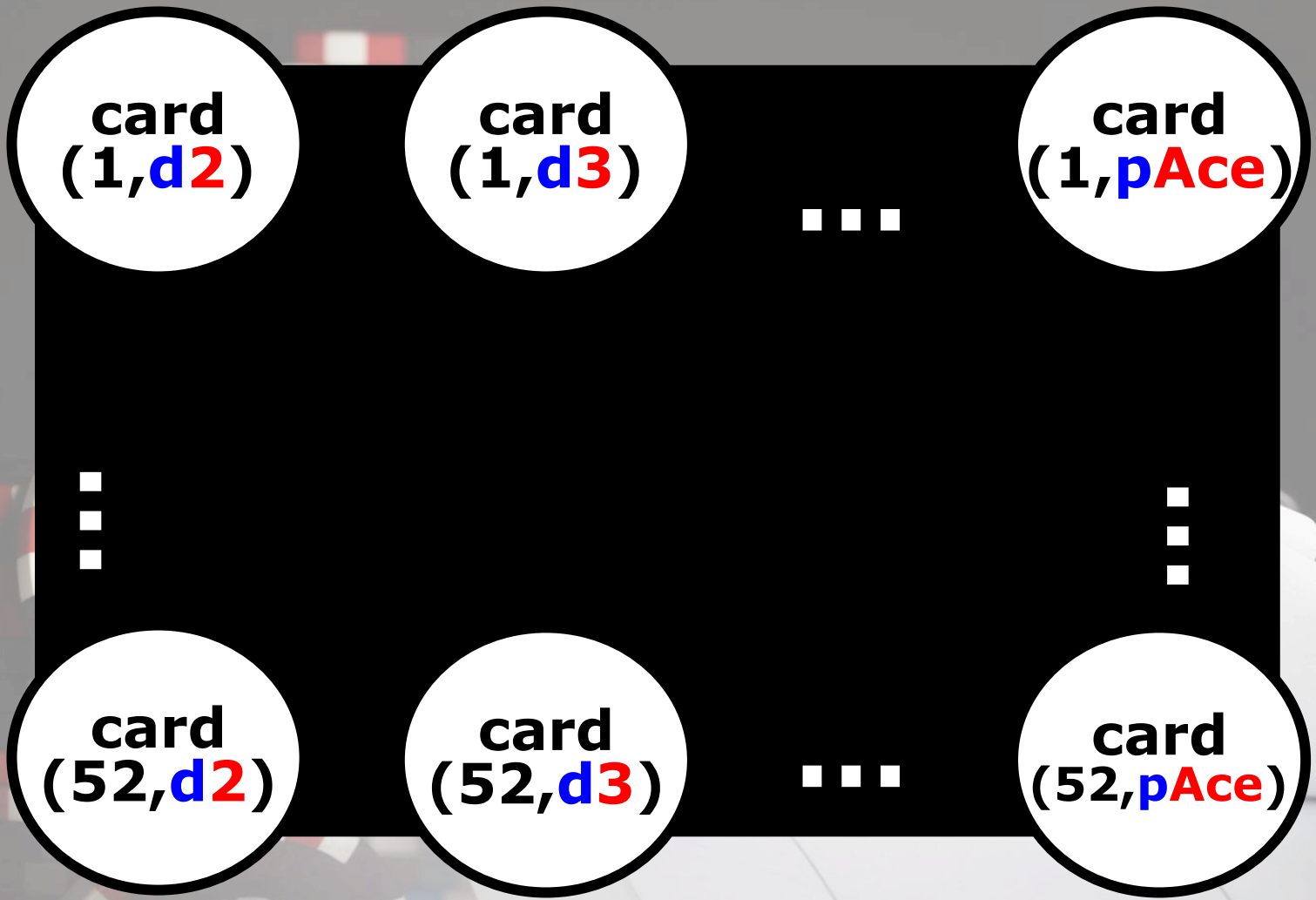
Guy van den Broeck
UCLA



# A simple example



Guy van den Broeck
UCLA



# A simple example



Guy van den Broeck
UCLA

card (1,d2)

card (1,d3)

...

card (1,pAce)

**No independencies.**
**Fully connected.**
**$2^{2704}$ states**

card (52,d2)

card (52,d3)

...

card (52,pAce)

# A simple example



Guy van den Broeck
UCLA

card (1,d2)

card (1,d3)

...

card (1,pAce)

**A machine will not solve the problem**

⋮

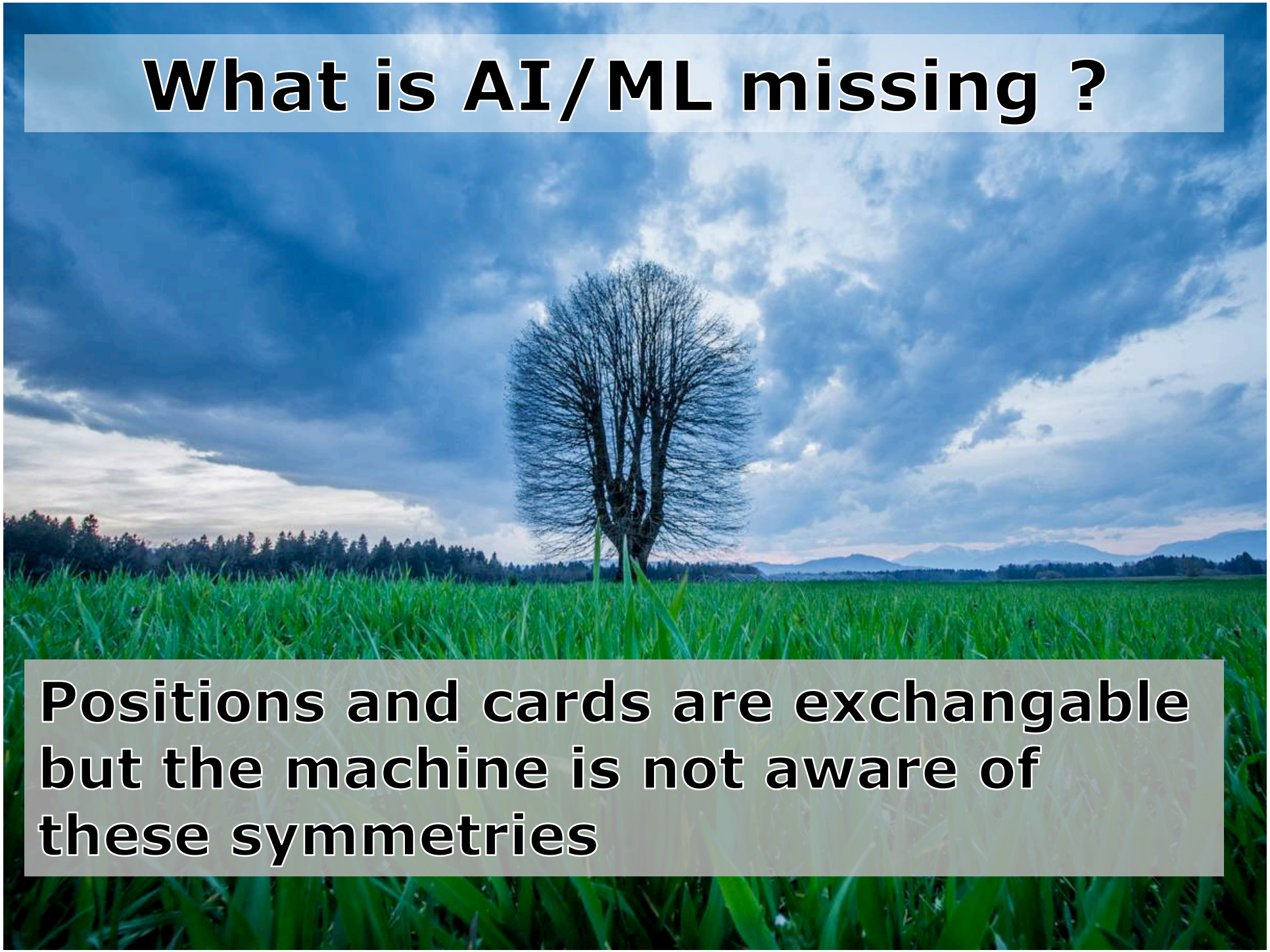
⋮

card (52,d2)

card (52,d3)

...

card (52,pAce)


What is AI/ML missing ?
Positions and cards are exchangable but the machine is not aware of these symmetries

## Faster modelling

Let's use a high-level language e.g. Markov Logic Networks (MLNs)

w1: $\forall$ p,x,y: card(P,X),card(P,Y)$\Rightarrow$x=y

w2: $\forall$ c,x,y: card(X,C),card(Y,C)$\Rightarrow$x=y

and symmetry- and language-aware infernce

## Faster inference and learning

# WHAT ARE SYMMETRIES IN APPROXIMATE PROBABILISTIC INFERENCE?

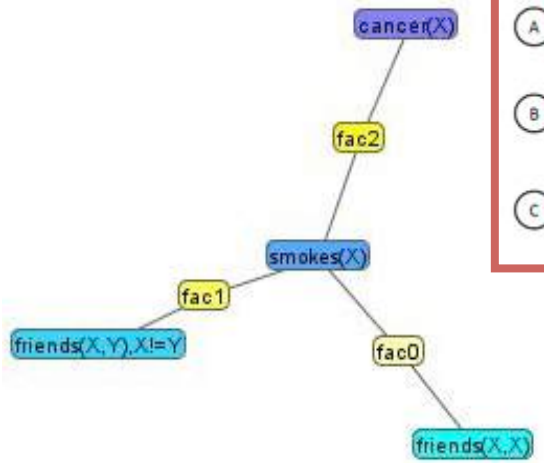


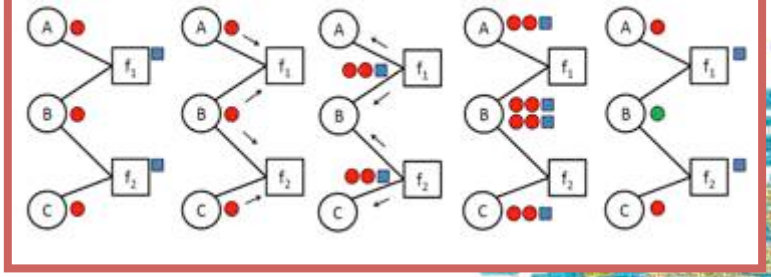
technische universität
dortmund

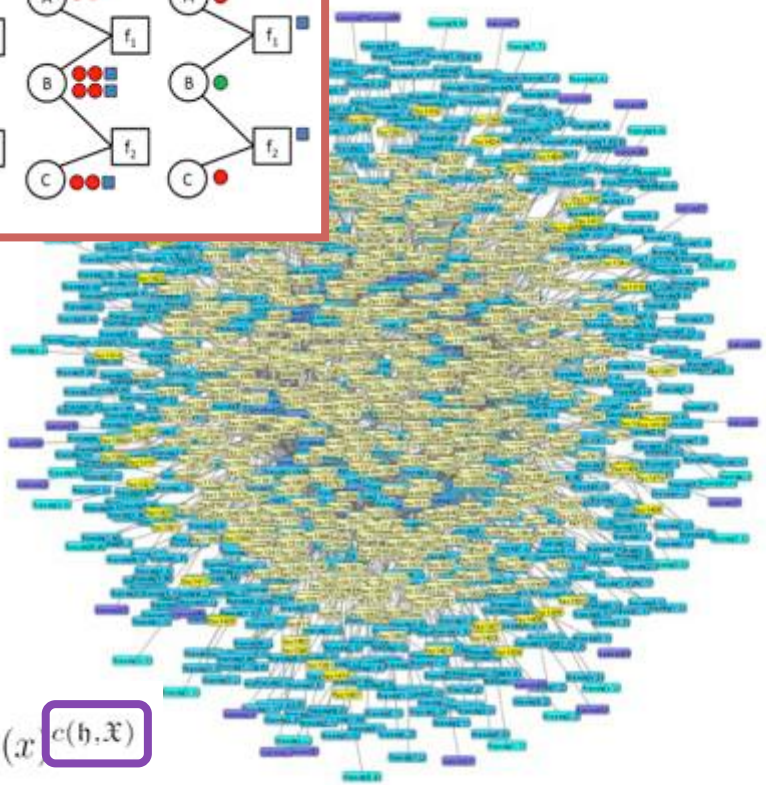
[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

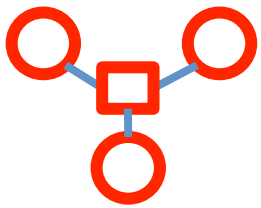
# Lifted Loopy Belief Propagation = Exploit computational symmetries

**Compress the model**





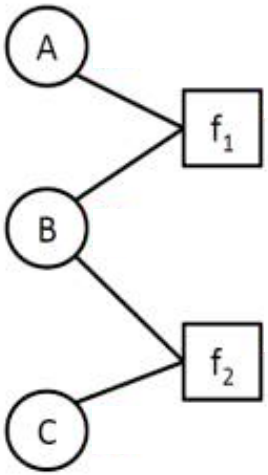
$$b_i(x_i) = \prod_{\mathrm{f}\in \mathrm{nb}(\mathfrak{X}_i)} \mu_{\mathrm{f}\to\mathfrak{X}_i}(x_i)^{c(\mathrm{f},\mathfrak{X})}$$

$$\mu_{\mathfrak{X}\to\mathrm{f}}(x) = \mu_{\mathrm{f}\to\mathfrak{X}}(x)^{c(\mathrm{f},\mathfrak{X})-1} \cdot \prod_{\mathfrak{h}\in \mathrm{nb}(\mathfrak{X})\setminus\{\mathrm{f}\}} \mu_{\mathfrak{h}\to\mathfrak{X}}(x)^{c(\mathfrak{h},\mathfrak{X})}$$

**Run a modified Loopy Belief Propagation**

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

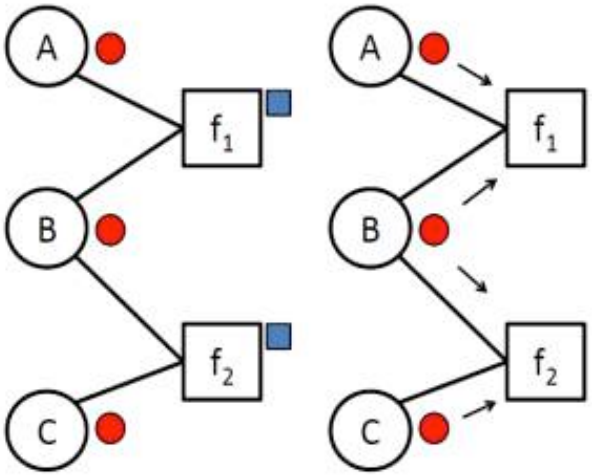
# Compression: Coloring the graph




- **Color nodes according to the evidence you have**
  - No evidence, say **red**
  - State „one", say **brown**
  - State „two", say **orange**
  - ...
- **Color factors distinctively according to their equivalences**
  For instance, assuming $f_1$ and $f_2$ to be identical and B appears at the second position within both, say **blue**

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

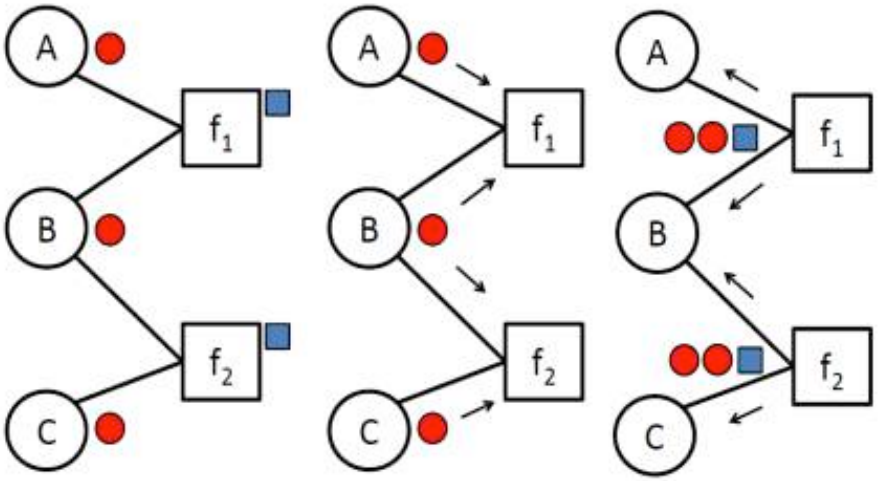
# Compression: Pass the colors around




1. Each factor collects the colors of its neighboring nodes

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

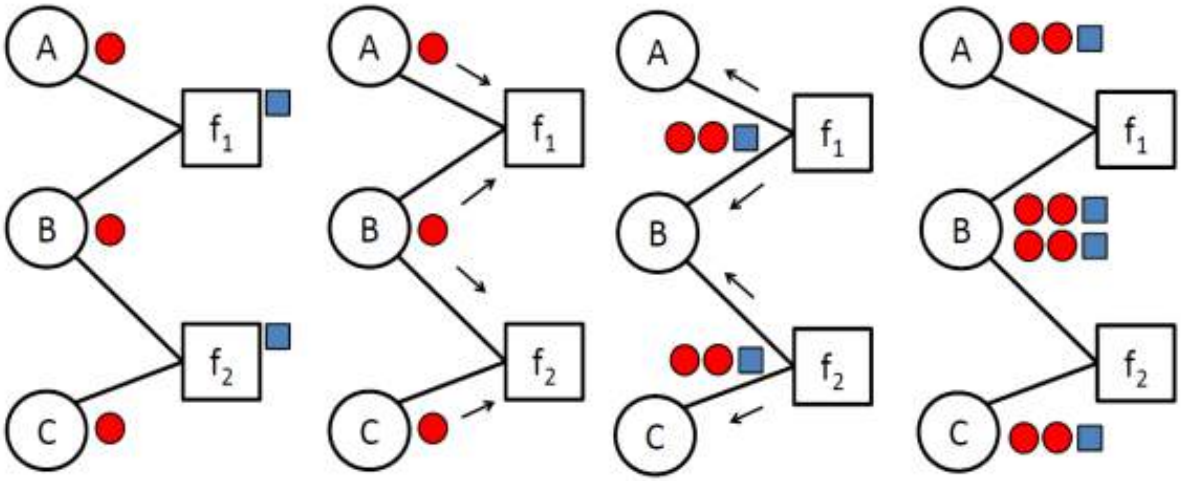
# Compression: Pass the colors around




1. Each factor collects the colors of its neighboring nodes
2. Each factor „signs" ist color signature with its own color

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

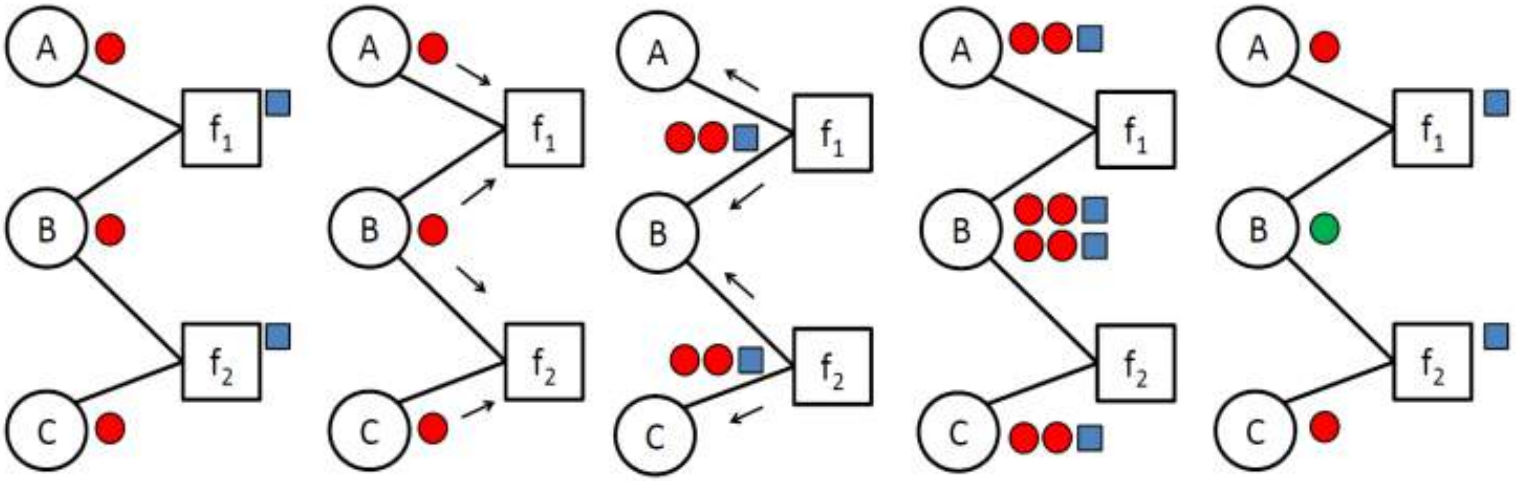
# Compression: Pass the colors around




1. Each factor collects the colors of its neighboring nodes
2. Each factor „signs" ist color signature with its own color
3. Each node collects the signatures of its neighboring factors

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

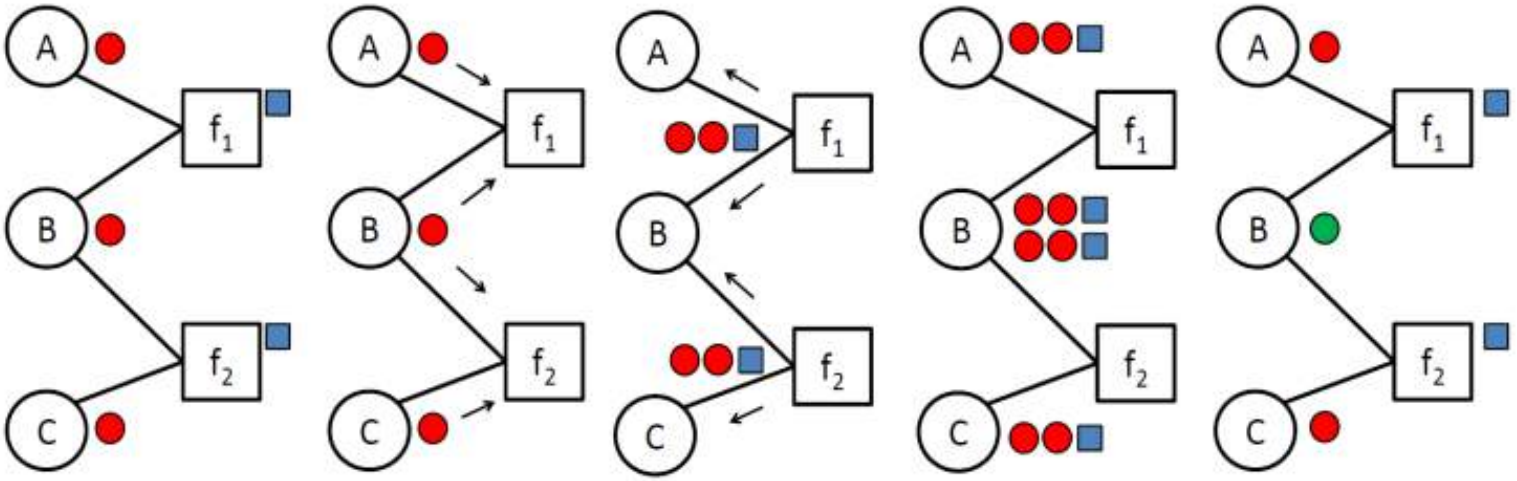
# Compression: Pass the colors around




1. Each factor collects the colors of its neighboring nodes
2. Each factor „signs" ist color signature with its own color
3. Each node collects the signatures of its neighboring factors
4. Nodes are recolored according to the collected signatures

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13]

# Compression: Pass the colors around



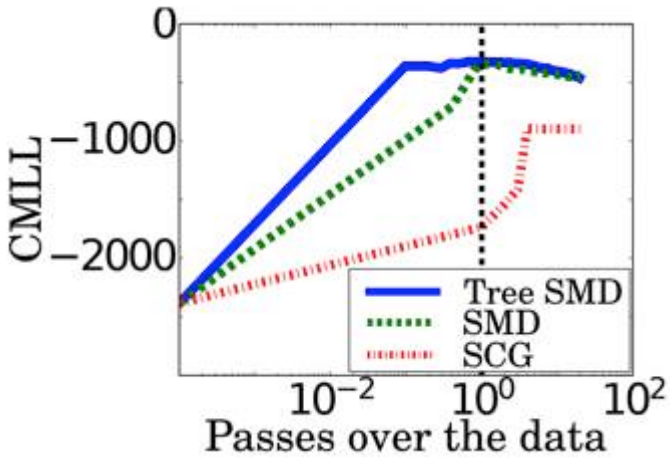

1. Each factor collects the colors of its neighboring nodes
2. Each factor „signs" ist color signature with its own color
3. Each node collects the signatures of its neighboring factors
4. Nodes are recolored according to the collected signatures
5. If no new color is created stop, otherwise go back to 1

[Singla, Domingos AAAI'08; **Kersting**, Ahmadi, Natarajan UAI'09; Ahmadi, **Kersting**, Mladenov, Natarajan MLJ'13; Van Haaren, Van den Broeck, Meert, Davis MLJ'16]

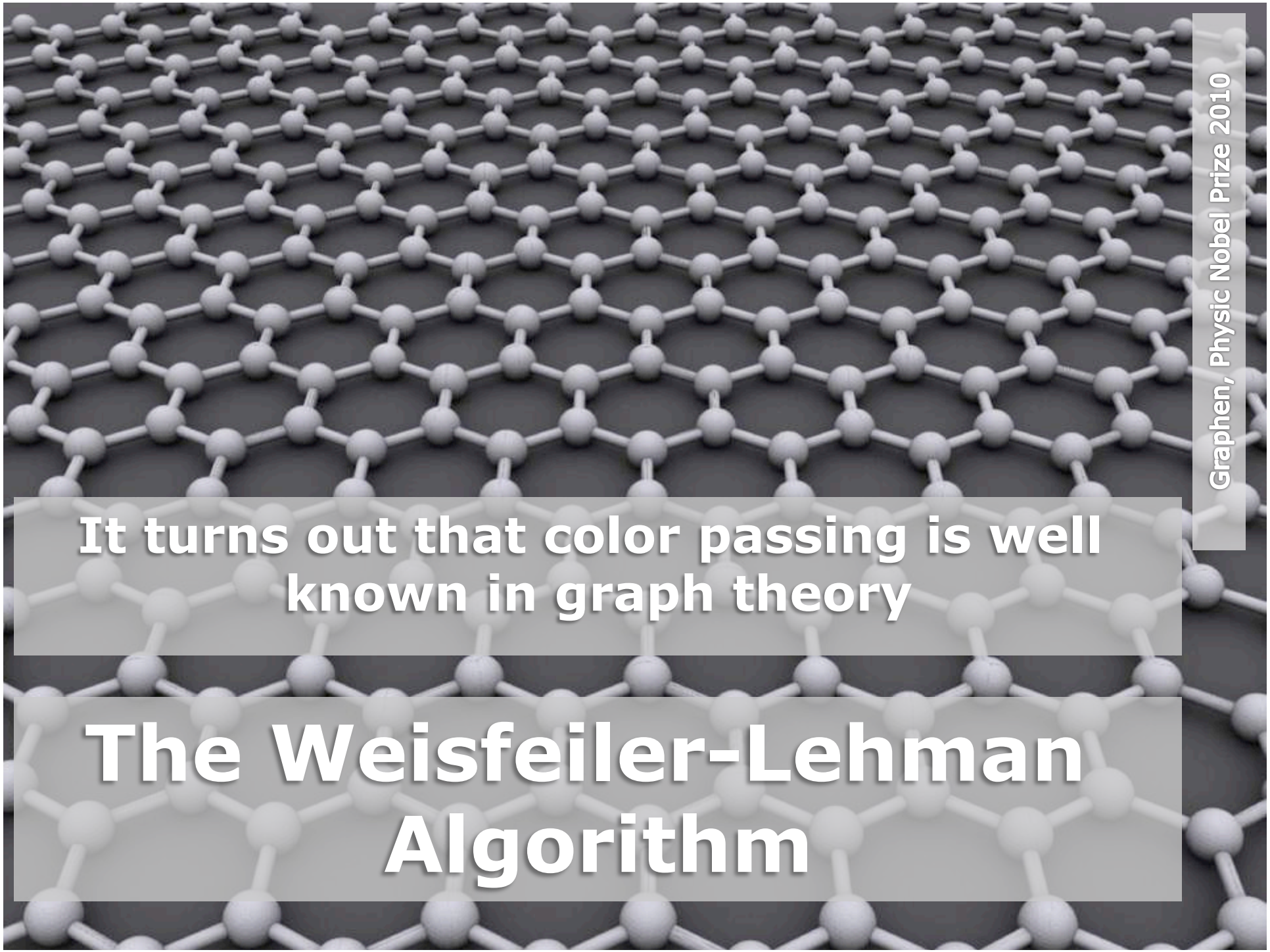
# This compression can speed up considerably loopy BP and also training graphical models

| Domain | Time (in seconds) | | | | | | No. of (Super) Features | |
|---|---|---|---|---|---|---|---|---|
| | Construction | | BP | | Total | | | |
| | Ground | Lifted | Ground | Lifted | Ground | Lifted | Ground | Lifted |
| Cora | 263.1 | 1173.3 | 12368.4 | 3997.7 | 12631.6 | 5171.1 | 2078629 | 295468 |
| UW-CSE | 6.9 | 22.1 | 1015.8 | 602.5 | 1022.8 | 624.7 | 217665 | 86459 |
| Friends & Smokers | 38.8 | 89.7 | 10702.2 | 4.4 | 10741.0 | 94.2 | 1900905 | 58 |

**can be orders of magnitude faster than state-of-the-art**




**CORA entity resolution parameter training using a lifted stochastic gradient**

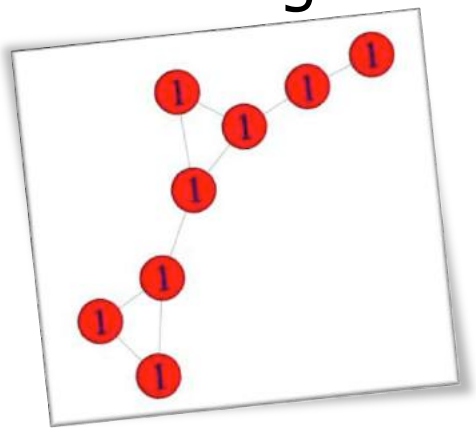



It turns out that color passing is well known in graph theory

# The Weisfeiler-Lehman Algorithm

# Weisfeiler-Lehman (WL) Algorithmus aka "naive vertex classification"

- Basic subroutine for graph isomorphism testing
- Computes LP-relaxations of GA-ILP, aka. **fractional automorphisms**
- **Quasi-linear** running time $O((n+m)\log(n))$ when using asynchronous updates [Berkholz, et al. ESA´13]

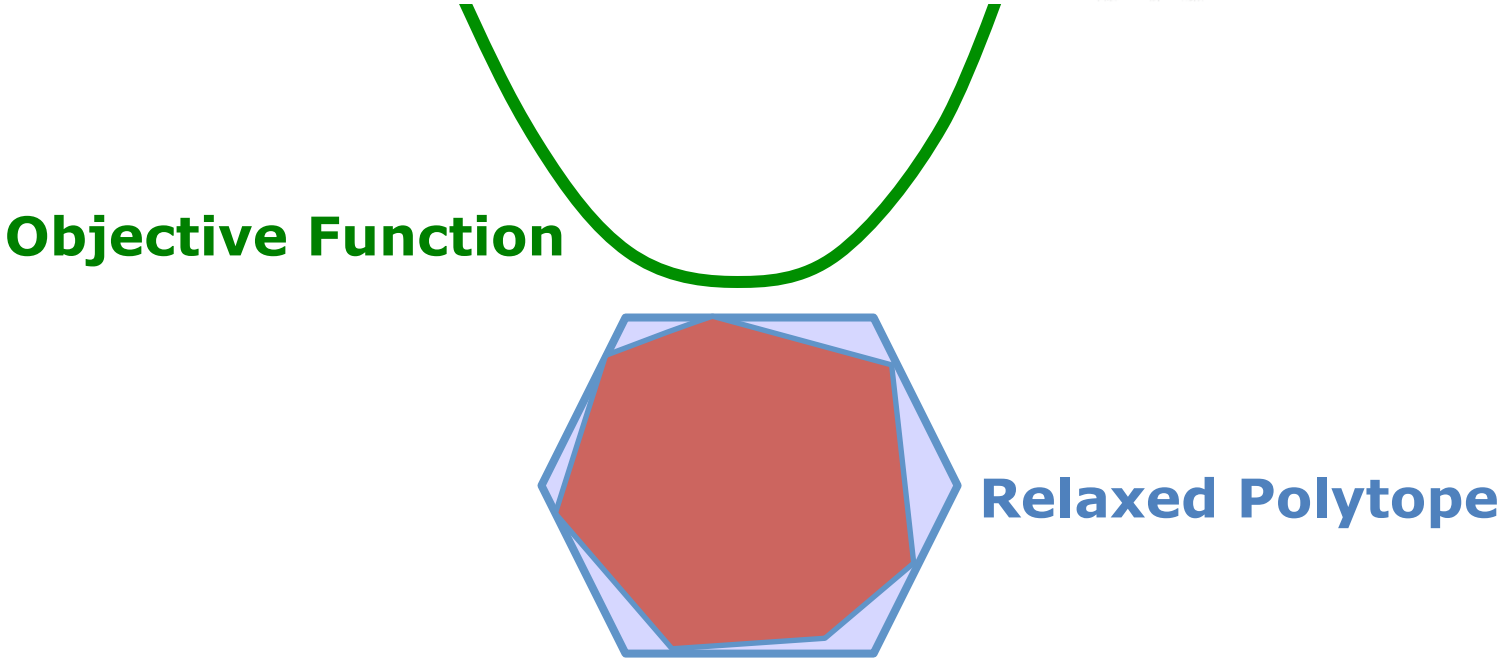



- **Part of graph tool SAUCY** [See e.g. Darga, Sakallah, Markov DAC´08]
- Has lead to highly performant **graph kernels** [Shervashidze et al. JMLR 12:2539-2561 ´11, Neumann, Garnett, Bauckhage, **Kersting** MLJ 102(2):209-245´16]
- Can be extended to **weighted graphs**/real-valued matrices [Grohe, **Kersting**, Mladenov, Selman ESA´14]
- Can be viewed as **recursive spectral clustering using random walks** [**Kersting**, Mladenov, Garnett, Grohe AAAI´14]


... and it paves the way to an algebraic understanding of lifted inference:
Instead of looking at ML through the glasses of probabilities over possible worlds, let's approach it using optimization

# Approximate inference in graphical models is closely connected to linear programs

$$\hat{\mathbf{x}} \quad \in \arg\max_{\mathbf{x}\in\mathcal{X}^N} \left\{ \sum_{s\in V} \theta_s(x_s) + \sum_{(s,t)\in E} \theta_{st}(x_s, x_t) \right\}$$

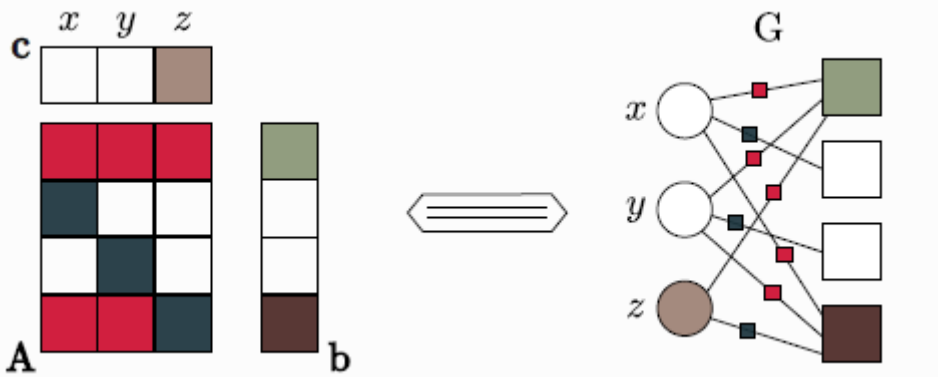


[Mladenov, Ahmadi, **Kersting** AISTATS´12, Grohe, **Kersting**, Mladenov, Selman ESA´14, **Kersting**, Mladenov, Tokmatov AIJ´15]

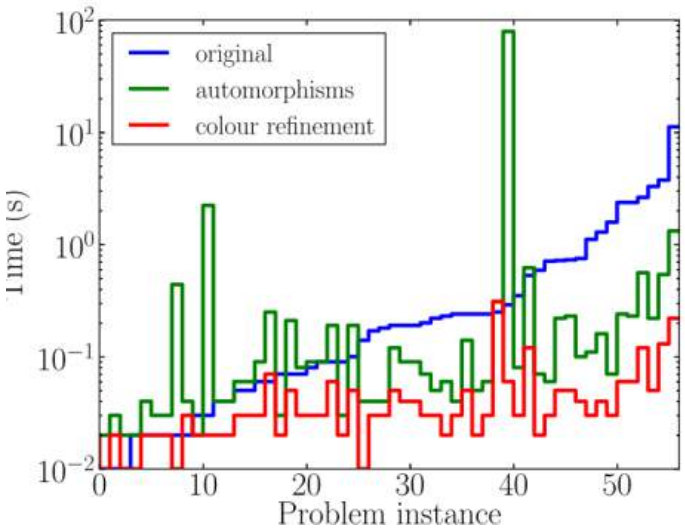
# Lifted Linear Programming

$$\max_{[x,y,z]^T \in \mathbb{R}^3} \quad 0x + 0y + 1z$$

$$\text{s.t.} \quad \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \leq \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$



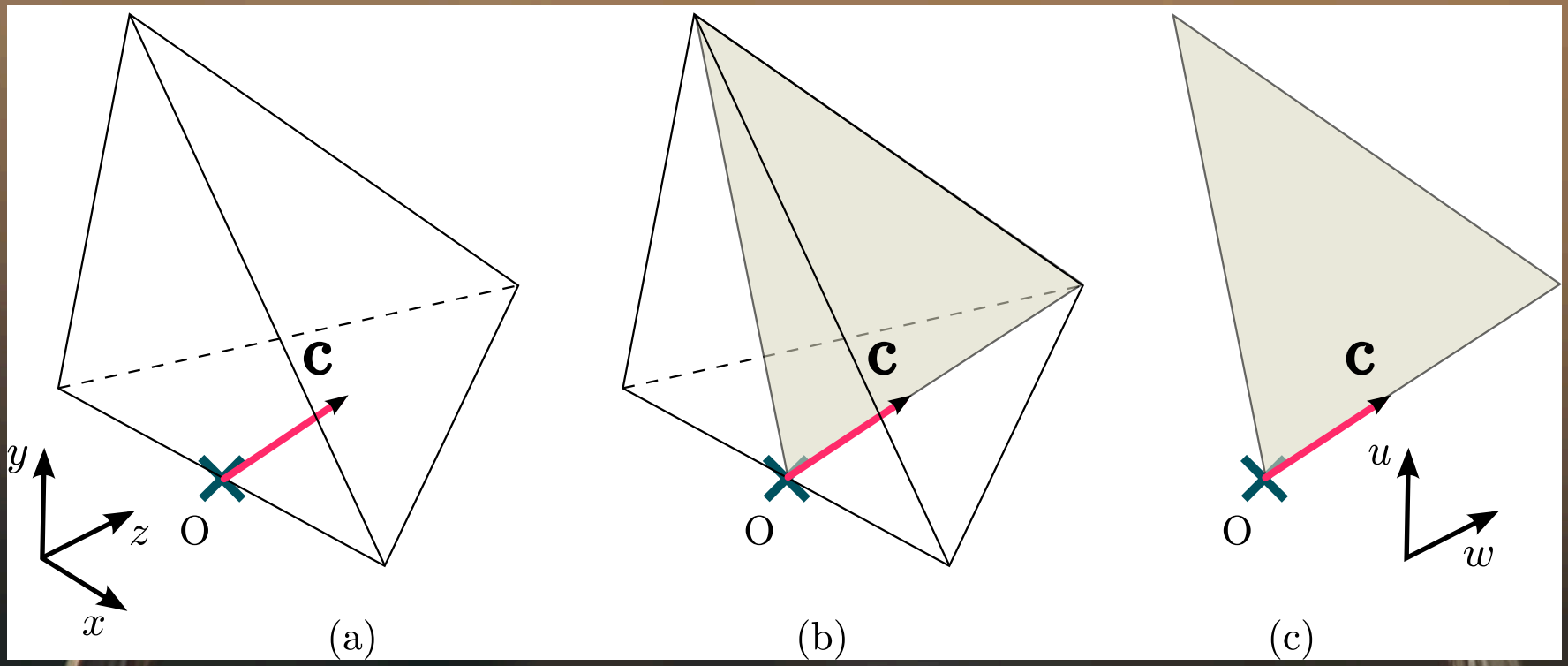

**(1) Reduce the LP by running WL on the LP-Graph**
**(2) Run any solver on the (hopefully) smaller LP**




quasi-linear overhead that may result in exponential speed up



technische universität dortmund




**Feasible region of LP and the objective vectors**

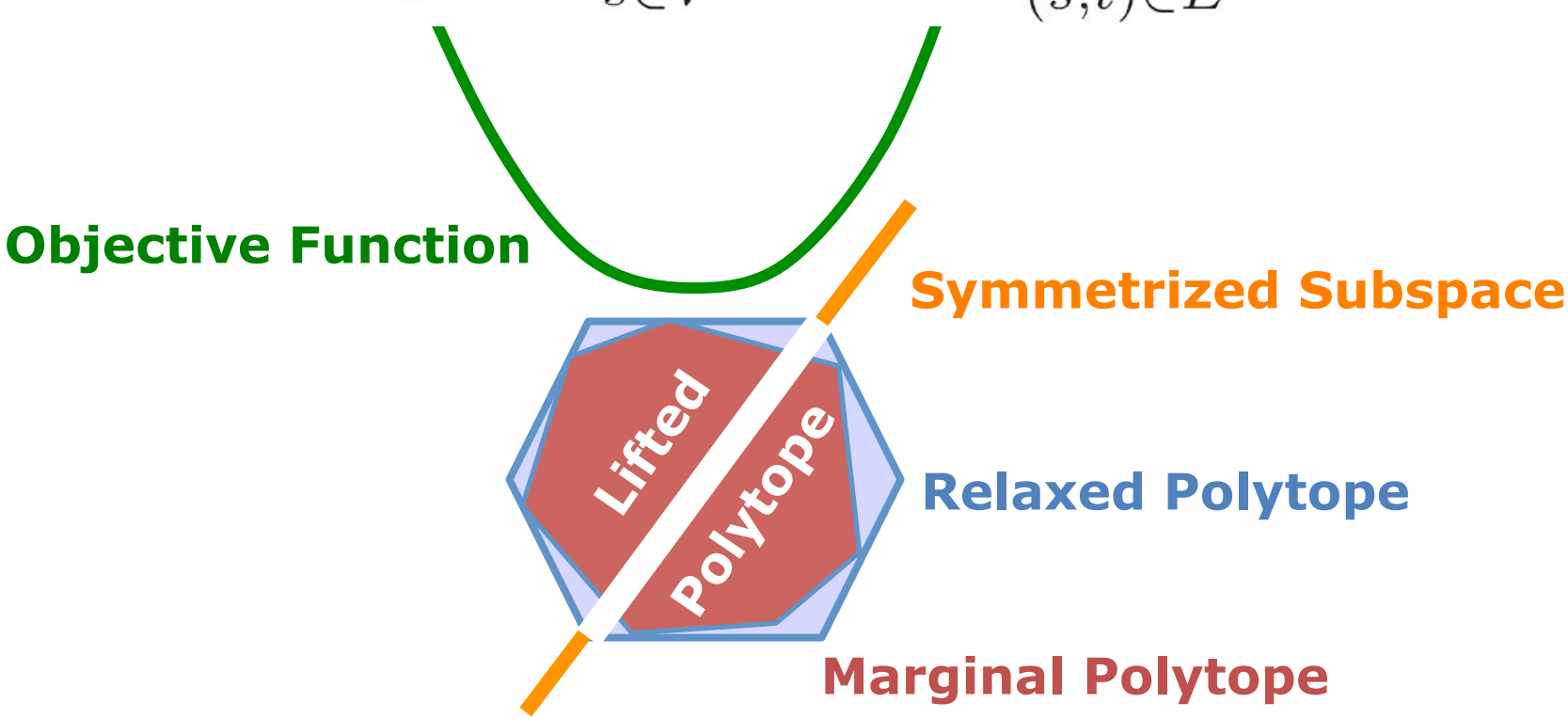
**Span of the fractional auto-morpishm of the LP**

**Projections of the feasible region onto the span of the fractional auto-morphism**

# Why does this work?

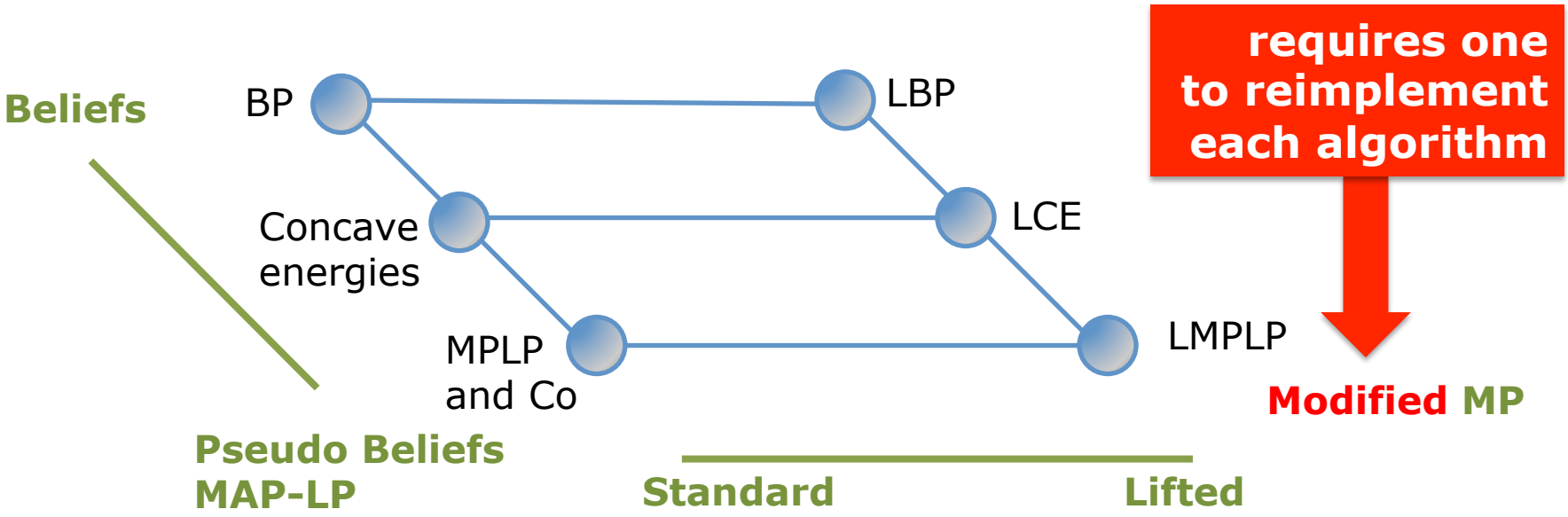
# Lifted approximate inference in graphical models

$$\hat{\mathbf{x}} \in \arg\max_{\mathbf{x}\in\mathcal{X}^N} \left\{ \sum_{s\in V} \theta_s(x_s) + \sum_{(s,t)\in E} \theta_{st}(x_s, x_t) \right\}$$

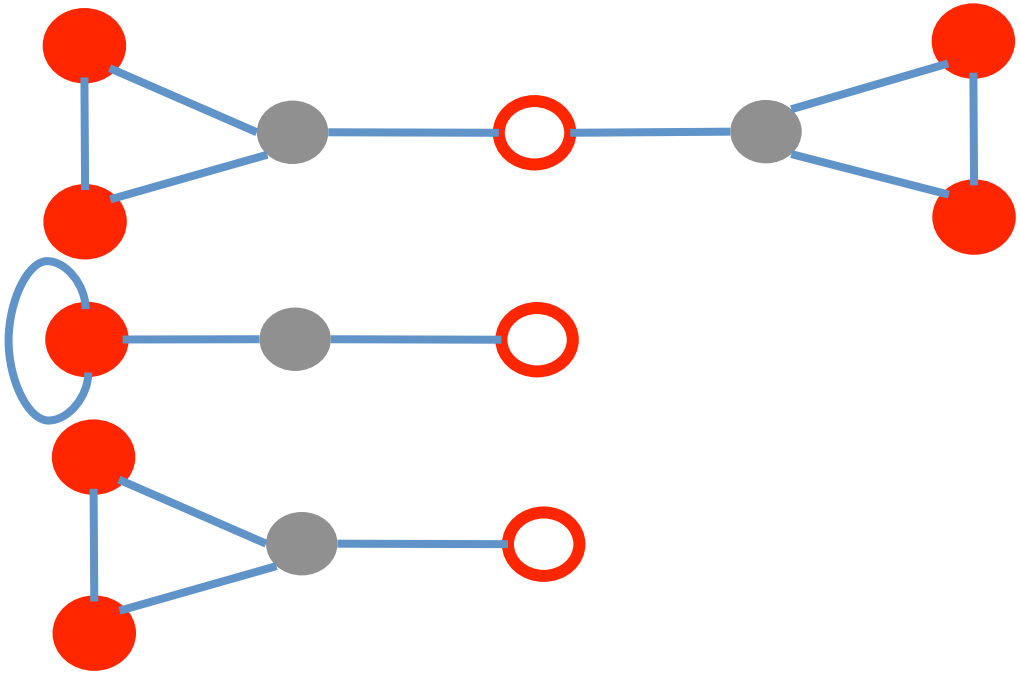


[Mladenov, Globerson, **Kersting** UAI 2014; Mladnov, **Kersting** UAI 2015]

# All message-passing inference approaches based on LPs are liftable



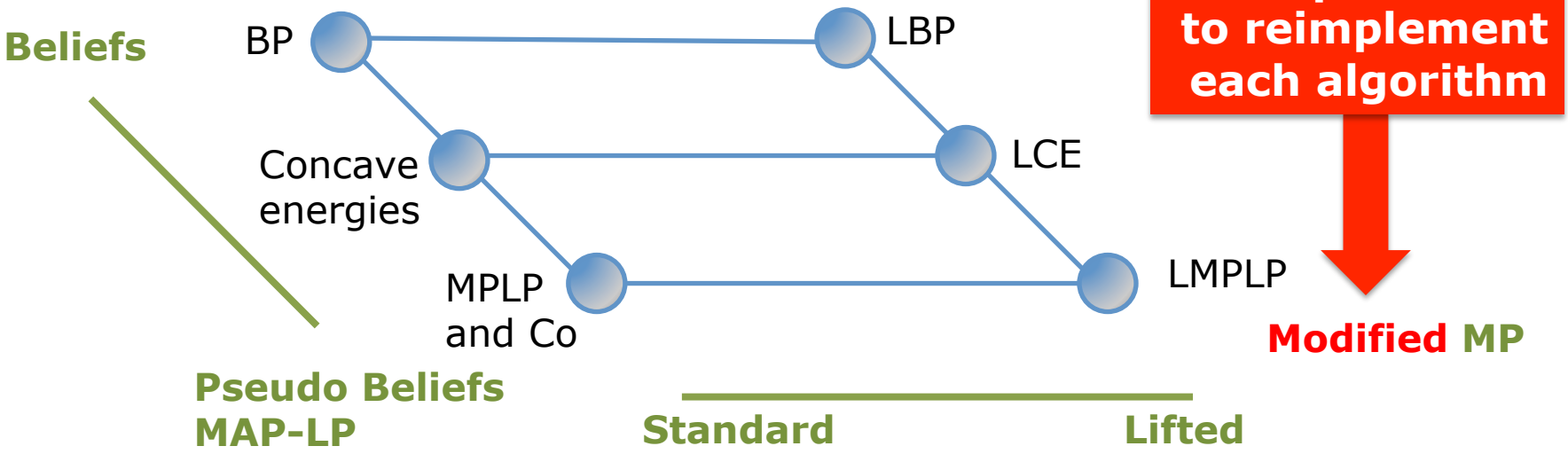
[Mladenov, Globerson, **Kersting** UAI 2014; Mladnov, **Kersting** UAI 2015]



**lifting**

**refine**

**requires one to reimplement each algorithm**



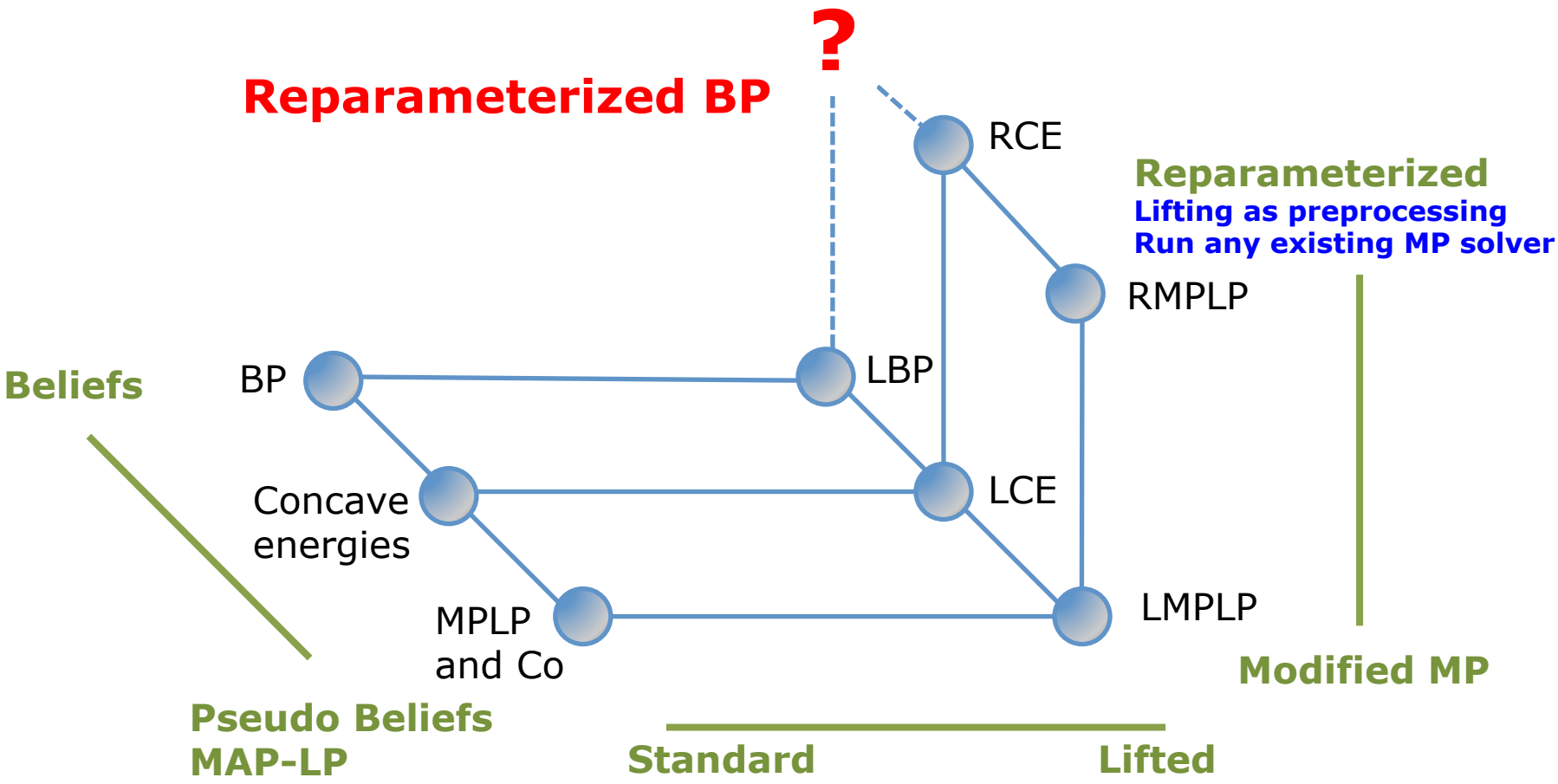



technische universität dortmund

[Mladenov, Globerson, **Kersting** UAI 2014; Mladnov, **Kersting** UAI 2015]

# Lifted inference =

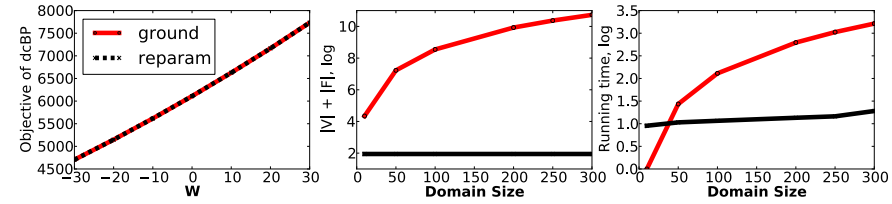
## Inference in a smaller, reparameterized model



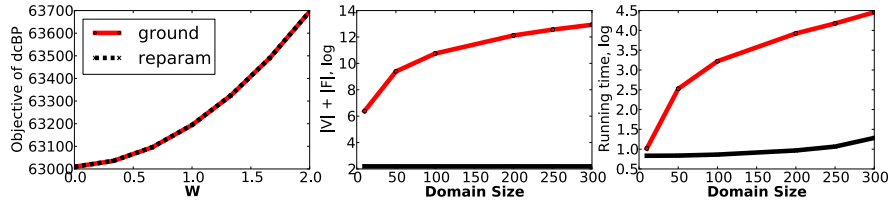



technische universität dortmund

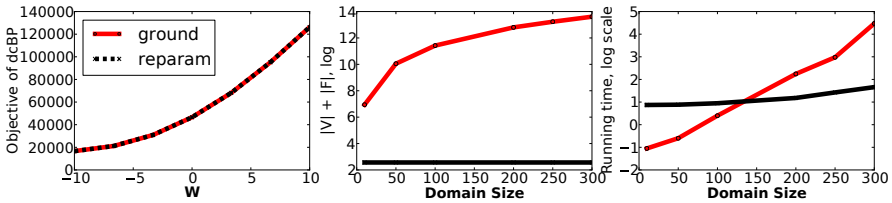
[Mladenov, Globerson, **Kersting** AISTATS `14, UAI `14; Mladenov, **Kersting** UAI´15]




(a) Complete Graph MLN.




(b) Clique-Cycle MLN.



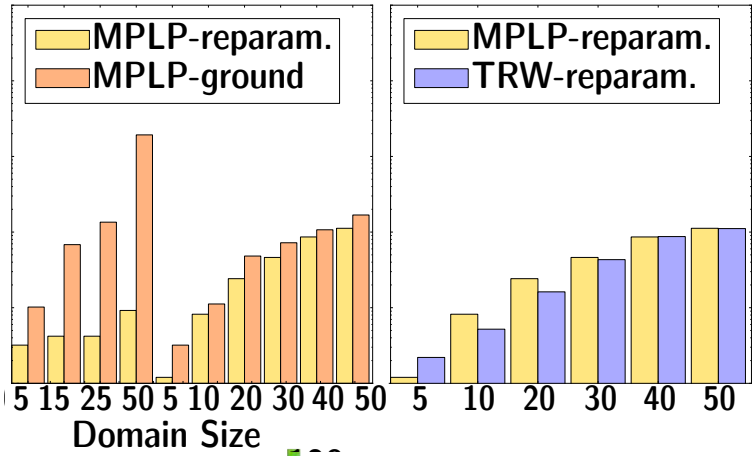

(c) Friends-smokers MLN.

# Marginals

## Any concave free energy is liftable

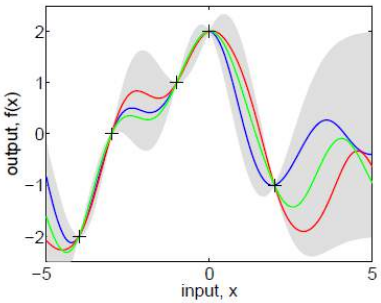
# MAP

## Any MAP-LP message-passing approach is liftable

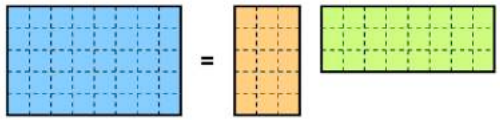




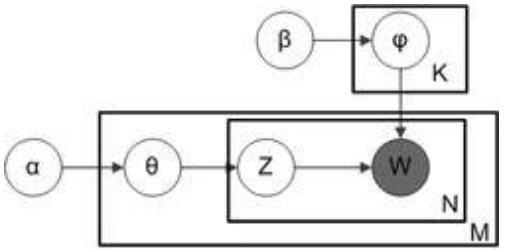

technische universität dortmund



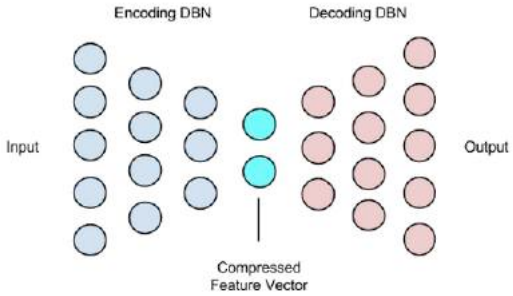

**Gaussian Processes**



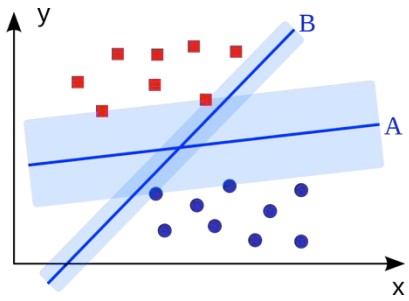
**Matrix Factorization**



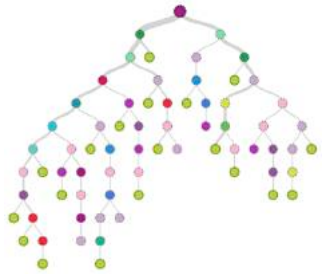

**Latent Dirichlet Allocation**




**Autoencoder/Deep Learning**




**Support Vector Machines**



**Decision Trees/Boosting**

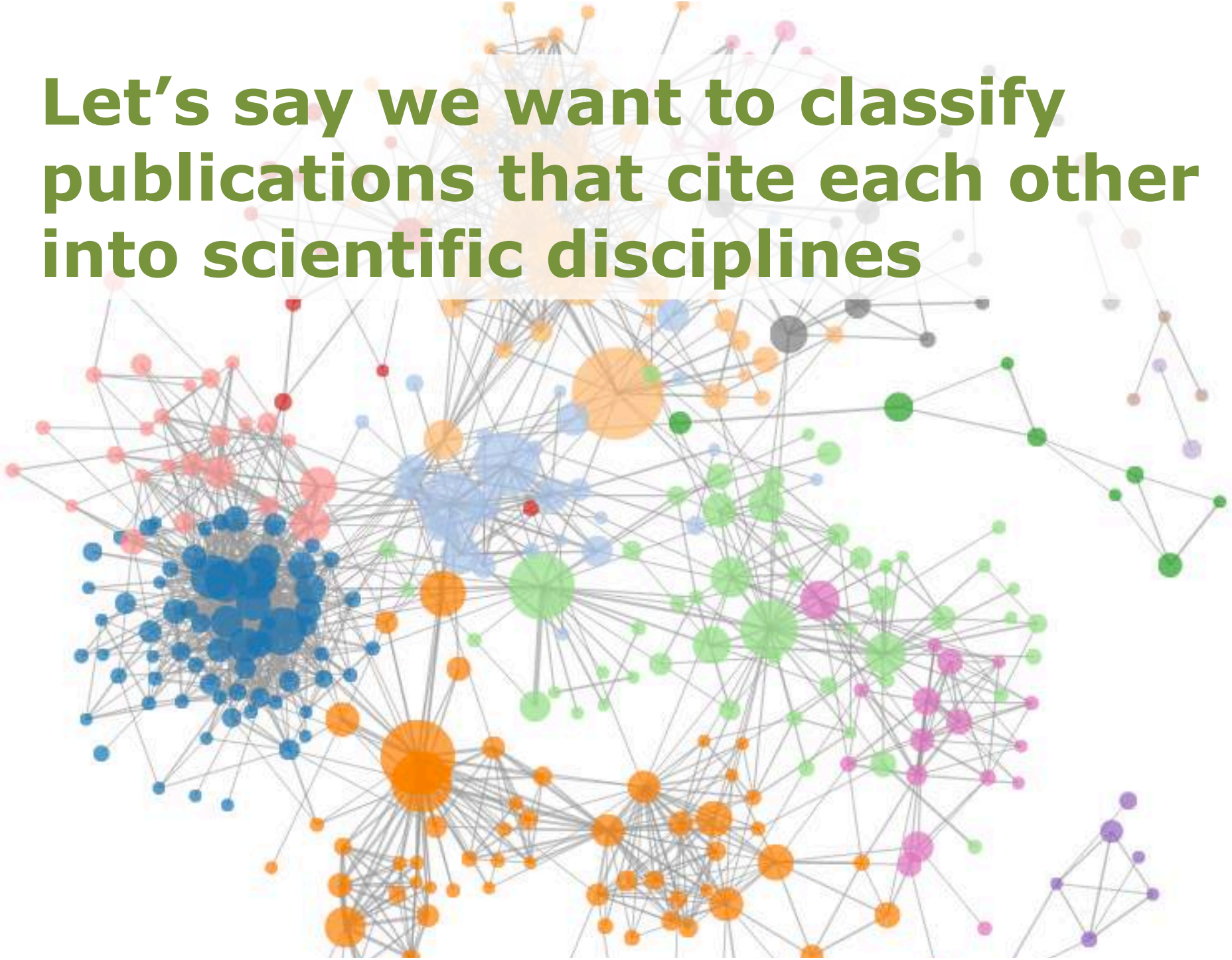
**and many more ...**

# BUT WAIT A MINUTE! WE WANT TO USE ANY DATA SCIENCE METHOD, NOT JUST GRAPHICAL MODELS!



technische universität dortmund

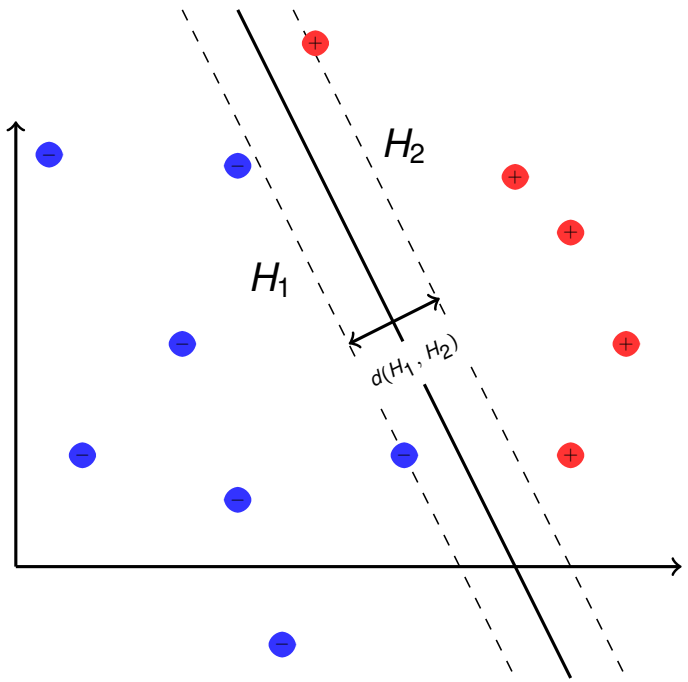
# Let’s say we want to classify publications that cite each other into scientific disciplines



[Vapnik ´79; Bennett´99; Mangasarian´99; Zhou, Zhang, Jiao´02, ... ]

# Standard data science approach: Support Vector Machines

$$H^* = \left\{ \vec{x} \mid \left\langle \vec{x}, \vec{\beta} \right\rangle + \beta_0 = 0 \right\}$$




$$d(H_1, H_2) = \frac{2}{||\vec{\beta}||}$$

This is a **quadratic program**. If you replace $l_2$- by $l_1$-,$l_\infty$-norm you get a **linear program**

[Kersting, Mladenov, Tokmakov AIJ´15, Mladenov, Heinrich, Kleinhans, Gonsio, Kersting DeLBP´16]

# Relational Mathemtical Programming

Write down the problem in „paper form“. The machine compiles into algebraic solver form.

reloop

http://www-ai.cs.uni-dortmund.de/weblab/static/RLP/html/

RELOOP: A Toolkit for Relational Convex Optimization

**Logically parameterized objective**

**Data stored externally**

**Logically parameterized variable (set of ground variables)**

```
abeled(
#QUADRATIC OBJECTIVE
minimize: sum{J in feature(I,J)} weight(J)**2 + c1 * slack;

#labeled examples should be on the correct side
subject to forall {I in labeled(I)}: labeled(I)*predict(I) >= 1 - slack(I)
```

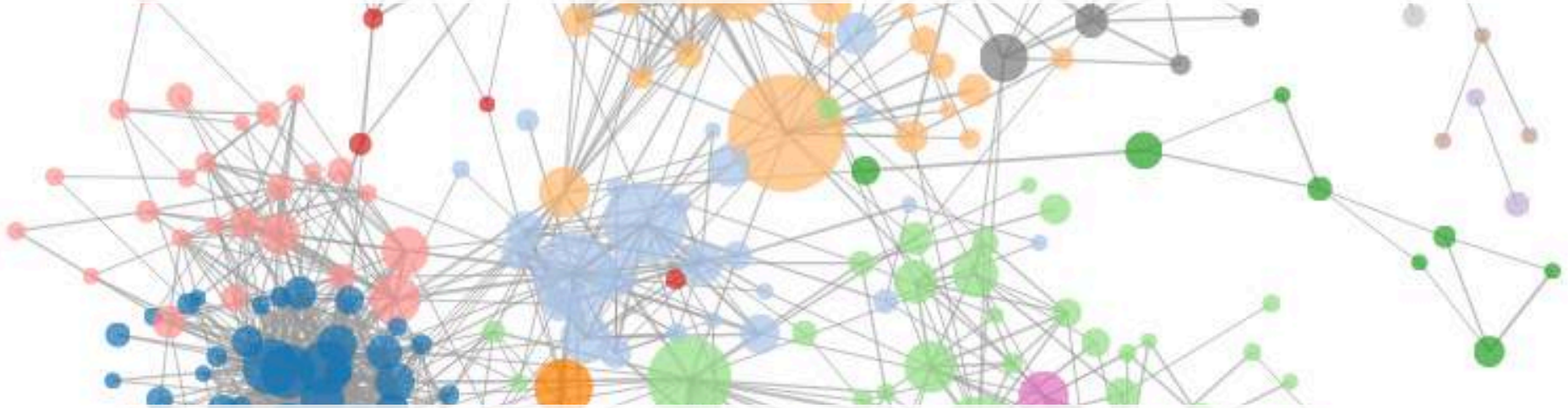
**Logically parameterized constraint**

DeepDive

LogicBlox

Google

**But wait, publications are citing each other. OMG, I have to use graph kernels!**



# REALLY?

[Kersting, Mladenov, Tokmakov AIJ´15, Mladenov, Heinrich, Kleinhans, Gonsio, Kersting DeLBP´16]

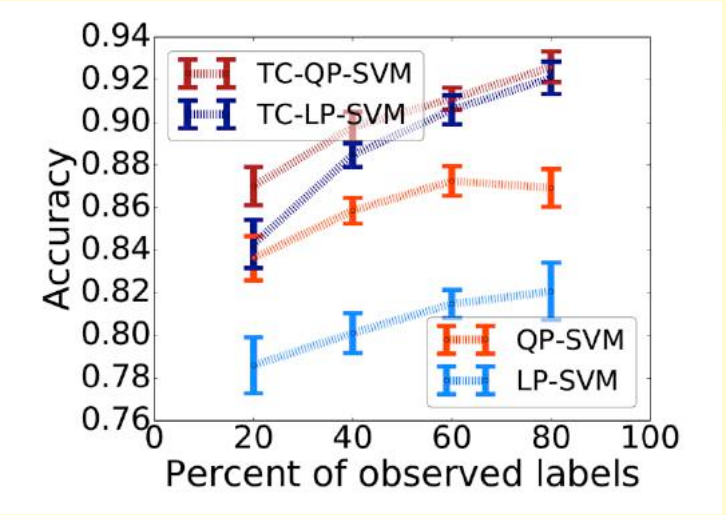
# Simply program some additional constraints

```
#inline definitions
slacks = sum{I in labeled(I)} slack(I);


#QUADRATIC OBJECTIVE
minimize: sum{J in feature(I,J)} weight(J)**2 + c1 * sl

#labeled examples should be on the correct side
subject to forall {I in labeled(I)}: labeled(I)*predict

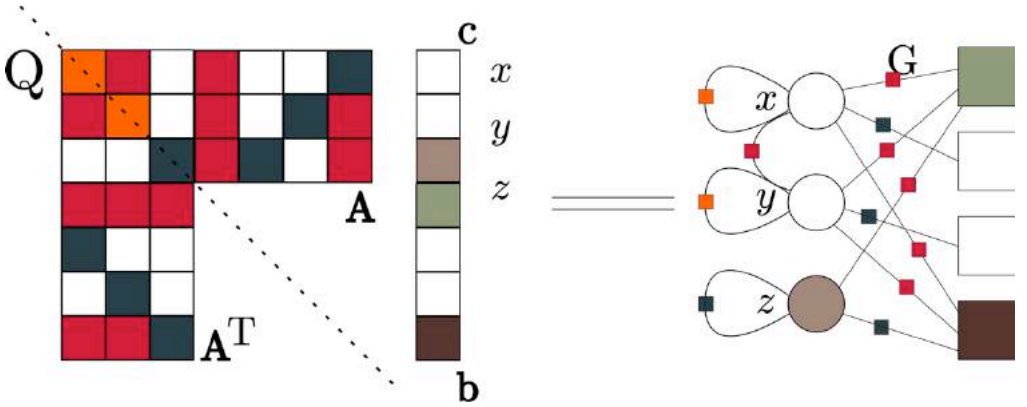
#slacks are positive
subject to forall {I in labeled(I)}: slack(I) >= 0;
```



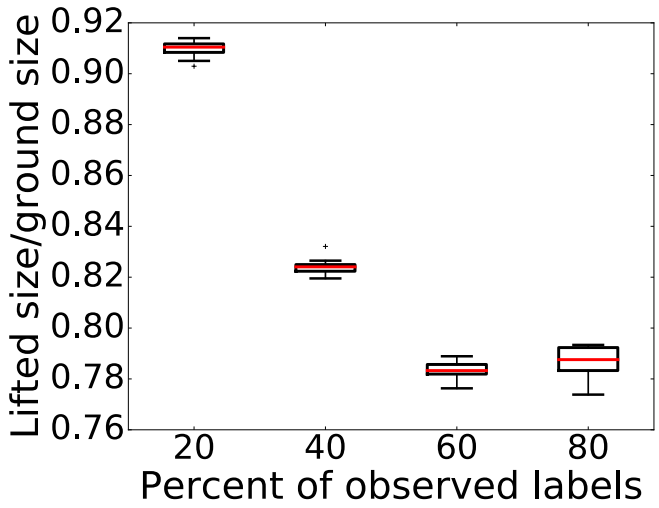
[**Kersting**, Mladenov, Tokmakov AIJ 2015, Mladenov, Kleinhans, **Kersting** CoRR abs/1606.04486 2016]

# Finally, the „-O1“ flag

$$\max_{[x,y,z]^T \in \mathbb{R}^3} \quad 0x + 0y + 1z$$
$$-1z^2 - 2x^2 - 2y^2 + 1xy + 1yx$$
$$\text{s.t.} \quad \begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \leq \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

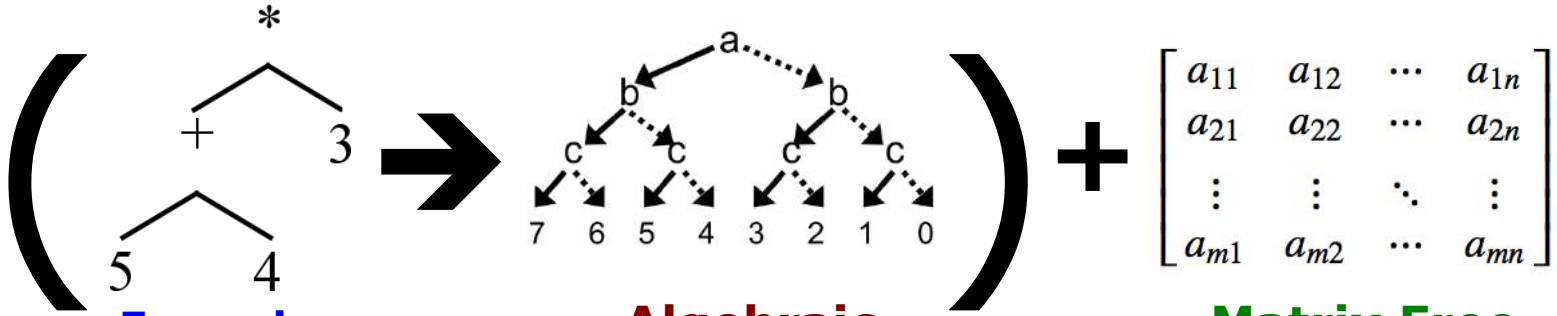



**(1) Reduce the QP via symmetries**
**(2) Run any solver on the reduced QP**



[Mladenov, Belle, **Kersting** CoRR abs/1605.08187 2016]

# Actually, there there are other "-02", "-03", … flags ahead, e.g symbolic-numerical interior point solvers




**Formulae parse trees** | **Algebraic Decision Diagrams** | **Matrix Free Optimization**

| Problem Statistics | | | | Symbolic IPM | | Ground IPM |
|---|---|---|---|---|---|---|
| name | #vars | #constr | $nnz(A)$ | \|ADD\| | time[s] | time[s] |
| factory | 131.072 | 688.128 | 4.000.000 | 1819 | 6899 | **516** |
| factory0 | 524.288 | 2.752.510 | 15.510.000 | 1895 | **6544** | 7920 |
| factory1 | 2.097.150 | 11.000.000 | 59.549.700 | 2406 | **34749** | 159730 |
| factory2 | 4.194.300 | 22.020.100 | 119.099.000 | 2504 | **36248** | > 48hrs. |

**>4.8x faster than state-of-the-art**

Applies to QPs but here illustrated on MDPs for a factory agent which must paint two objects and connect them. The objects must be smoothed, shaped and polished and possibly drilled before painting, each of which actions require a number of tools which are possibly available. Various painting and connection methods are represented, each having an effect on the quality of the job, and each requiring tools. Rewards (required quality) range from 0 to 10 and a discounting factor of 0. 9 was used used

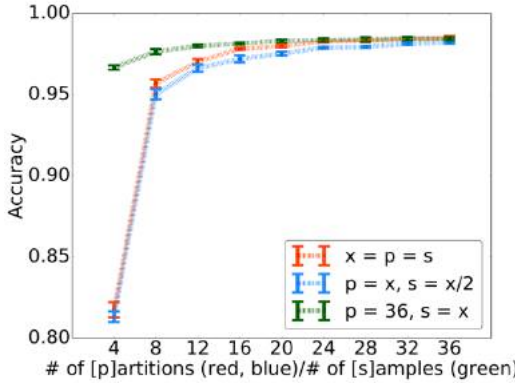
Conclusions

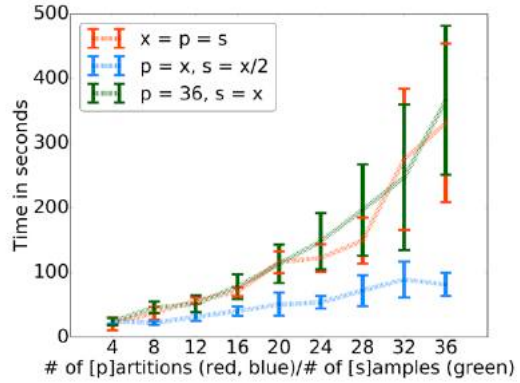
# FRACTIONAL SYMMETRIES ARE …

[**Kersting**, Kleinhans, Mladenov to be submitted]

# … a natural foundation for

# SYMMETRY-BASED DATA SCIENCE




(d) Learning time: Translation

- Learning (rich) representations is a central problem of data science
- (Fractional) symmetry / group theory is a natural foundation for learning representations
- Symmetries = “unimportant” variants of data (graphs, relational structures, …)
- Let’s move beyond QPs: CSPs, SDPs, Autoencoders, Deep Learners, …

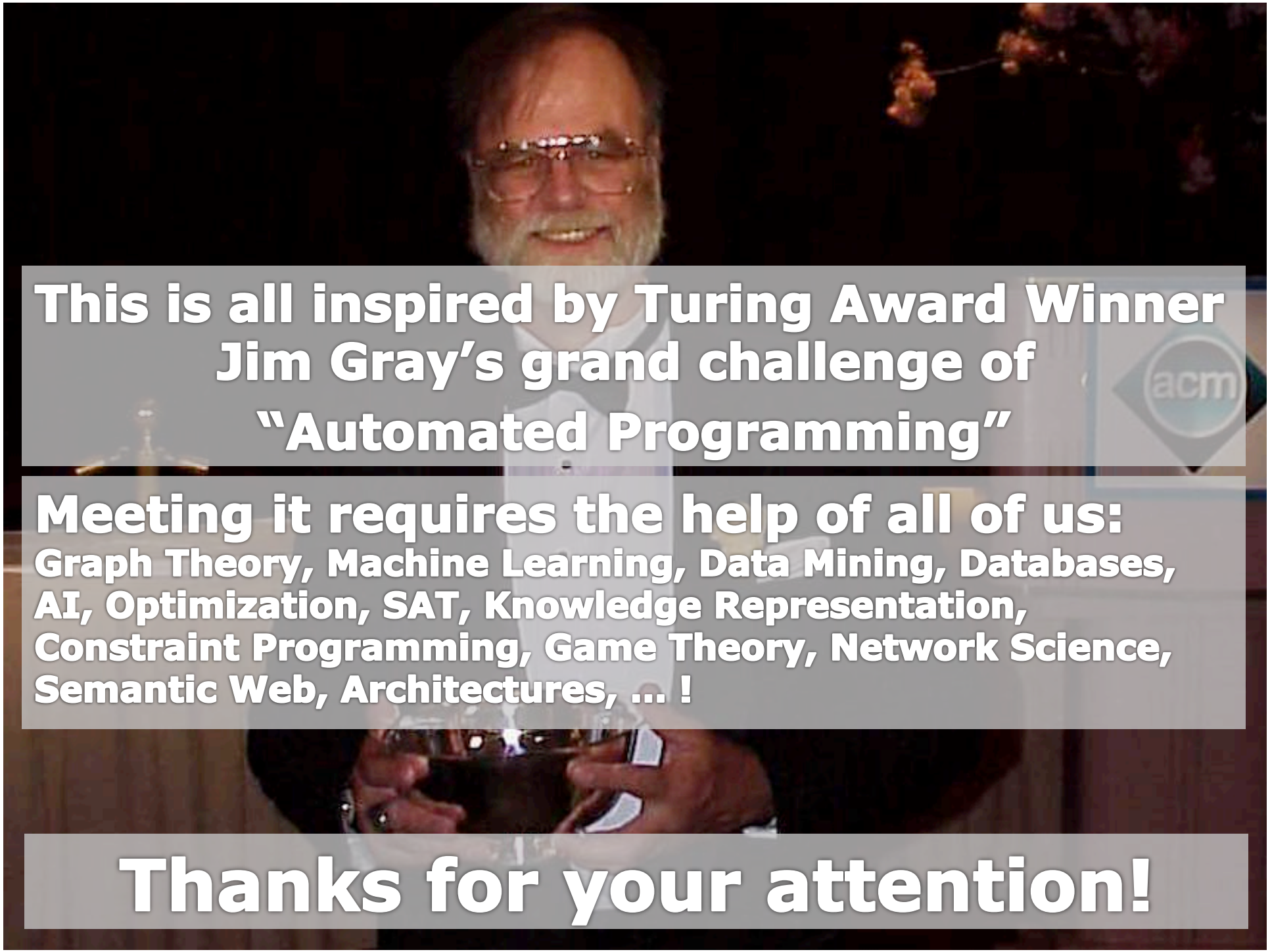


technische universität dortmund

# Together with high-level data science languages this is a step towards

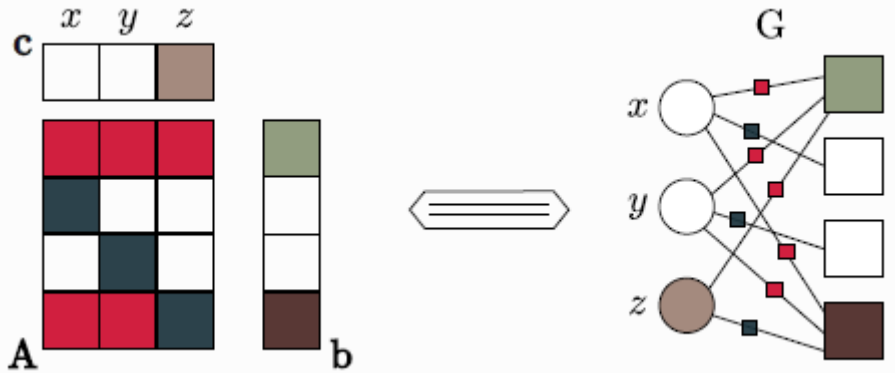
## DECLARATIVE DATA SCIENCE

- Reduces the level of expertise necessary to build data science applications, makes models faster to write and easier to communicate
- Facilitate the construction of sophisticated models with rich domain knowledge
- Speed up solvers by exploiting language properties, compression, and compilation


This is all inspired by Turing Award Winner
Jim Gray's grand challenge of
"Automated Programming"
Meeting it requires the help of all of us:
Graph Theory, Machine Learning, Data Mining, Databases,
AI, Optimization, SAT, Knowledge Representation,
Constraint Programming, Game Theory, Network Science,
Semantic Web, Architectures, ... !
Thanks for your attention!

[Mladenov, Ahmadi, **Kersting** AISTATS´12, Grohe, **Kersting**, Mladenov, Selman ESA´14, **Kersting**, Mladenov, Tokmatov AIJ´15]

# Compute Equitable Partition (EP) of the LP using WL




$$\mathcal{P} = \{P_1, \ldots, P_p; Q_1, \ldots, Q_q\}$$

**Partition of LP variables** | **Partition of LP constraints**

Intuitively, we group together variables resp. constraints that interact in the very same way in the LP.

# Fractional Automorphisms of LPs

The EP induces a fractional automorphism of the coefficient matrix **A**

$$\mathbf{X}_Q \mathbf{A} = \mathbf{A} \mathbf{X}_P$$

where $\mathbf{X}_Q$ and $\mathbf{X}_p$ are doubly-stochastic matrixes (relaxed form of automorphism)

$$(\mathbf{X}_P)_{ij} = \begin{cases} 1/|P| & \text{if both vertices } i,\ j \text{ are in the same } P, \\ 0 & \text{otherwise.} \end{cases}$$

$$(\mathbf{X}_Q)_{ij} = \begin{cases} 1/|Q| & \text{if both vertices } i,\ j \text{ are in the same } Q, \\ 0 & \text{otherwise} \end{cases}$$

# Fractional Automorphisms Preserve Solutions

If **x** is feasible, then $\mathbf{X}_p\mathbf{x}$ is feasible, too.

By induction, one can show that left-multiplying with a double-stochastic matrix preserves directions of inequalities. Hence,

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \Rightarrow \mathbf{X}_Q\mathbf{A}\mathbf{x} \leq \mathbf{X}_Q\mathbf{b} \Leftrightarrow \mathbf{A}\mathbf{X}_P\mathbf{x} \leq \mathbf{b}$$

# Fractional Automorphisms Preserve Solutions

If $\mathbf{x}^*$ is optimal, then $\mathbf{X}_p\mathbf{x}^*$ is optimal, too.

Since by construnction $\mathbf{c}^T\mathbf{X}_P = \mathbf{c}^T$ and hence

$$\mathbf{c}^T(\mathbf{X}_P\mathbf{x}) = \mathbf{c}^T\mathbf{x}$$

# What have we established so far?

Instead of considering the original LP

$$(\mathbf{A}, \mathbf{b}, \mathbf{c})$$

It is sufficient to consider

$$(\mathbf{A}\mathbf{X}_P, \mathbf{b}, \mathbf{X}_P{}^T\mathbf{c})$$

i.e. we "average" parts of the polytope.

**But why is this dimensionality reduction?**

# Dimensionality Reduction

The doubly-stochastic matrix $\mathbf{X}_P$ can be written as

$$\mathbf{X}_P = \mathbf{B}\mathbf{B}^T$$

$$\mathbf{B}_{iP} = \begin{cases} \frac{1}{\sqrt{|P|}} & \text{if vertex } i \text{ belongs to part } P, \\ 0 & \text{otherwise.} \end{cases}$$

Since the column space of B is equivalent to the span of $\mathbf{X}_P$, it is actually sufficient to consider only

$$(\mathbf{A}\mathbf{B}_P, \mathbf{b}, \mathbf{B}_P^T\mathbf{c})$$

This is of reduced size, and actually we can also drop any constraint that becomes identical