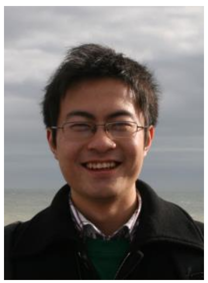
Probabilistic Graphical Models Parameter Learning with Transferred Prior and Constraints

Yun Zhou, Norman Fenton, Timothy Hospedales, Martin Neil




UAI-2015, Amsterdam, The Netherlands

13/07/2015

Queen Mary University of London

CHINA SCHOLARSHIP COUNCIL

NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY

The Scenario

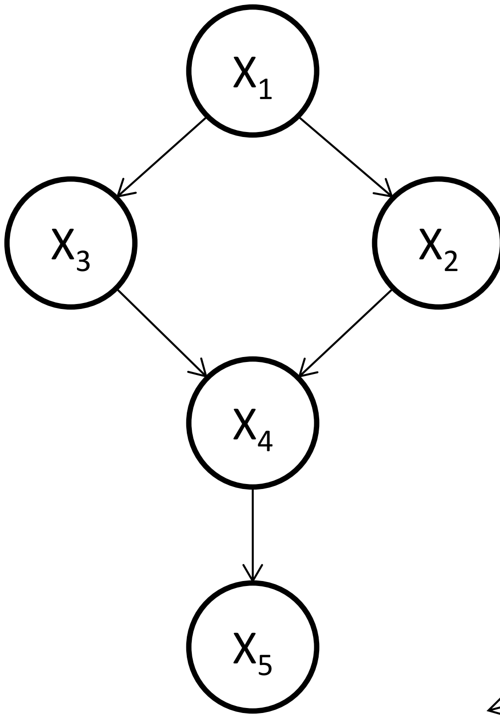
- *A Bayesian network (BN) structure has been hand-crafted by domain experts to model a real-world risk assessment problem.*
- *Only a small amount of data relevant to the model is available.*
- *The challenge is to build the model parameters by exploiting the limited data, expert knowledge and knowledge from related domains.*

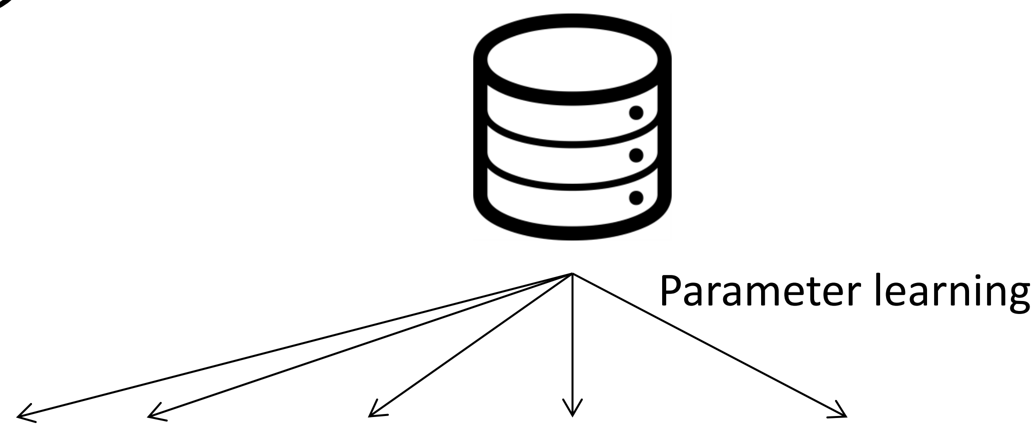
Overview

- Background
- Related Work
- The Model
- Experiments
- Conclusions

Background - The Basics

- Bayesian network

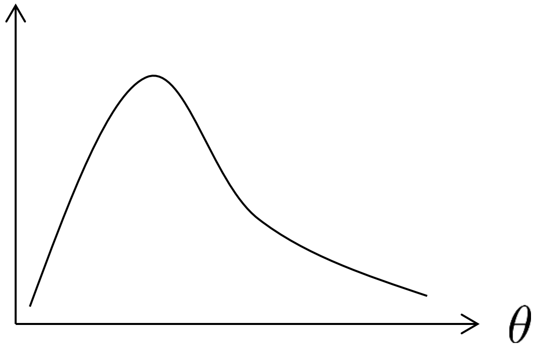




$$p(X_1, X_2, X_3, X_4, X_5) = p(X_1)p(X_2|X_1)p(X_3|X_1)p(X_4|X_2, X_3)p(X_5|X_4)$$

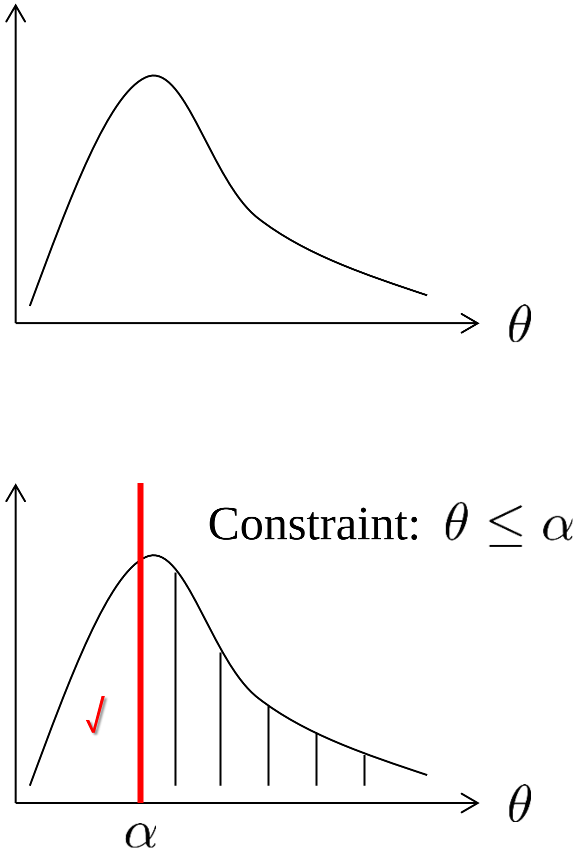
Background - The Idea

- Constraints and related information.



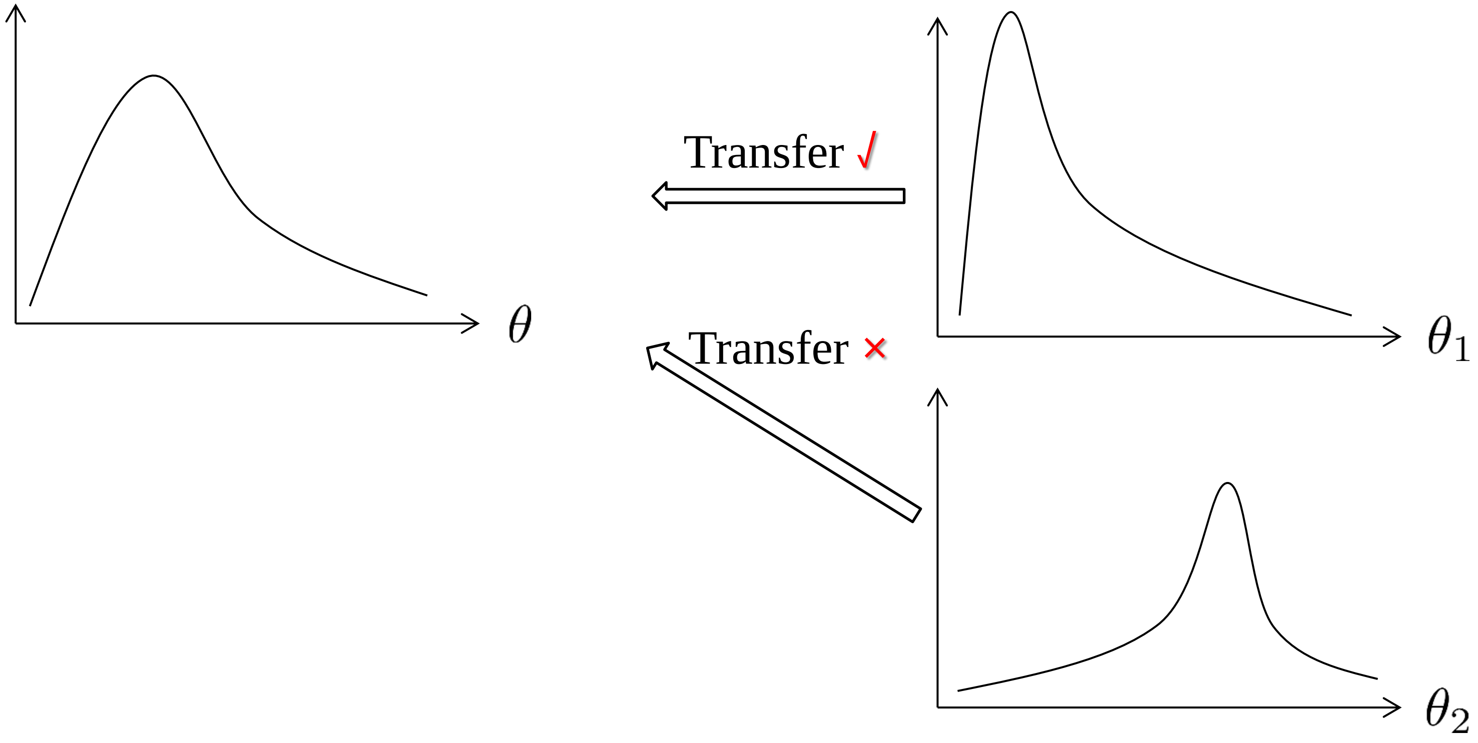
Background - The Idea

- Constraints and related information.



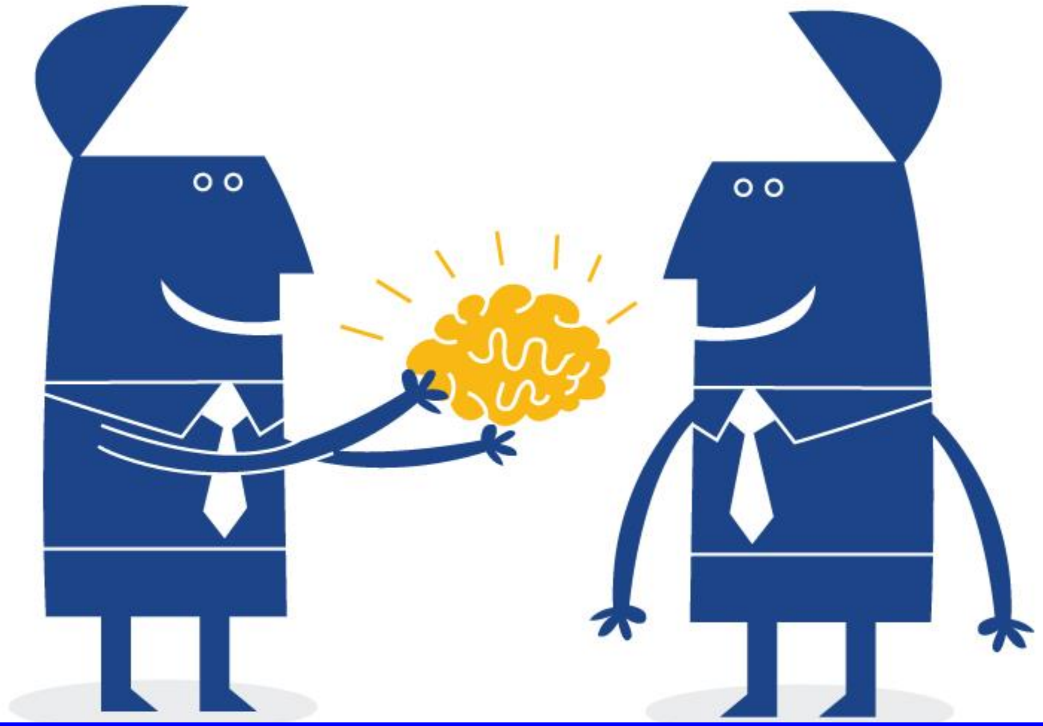
Background - The Idea

- Constraints and related information.



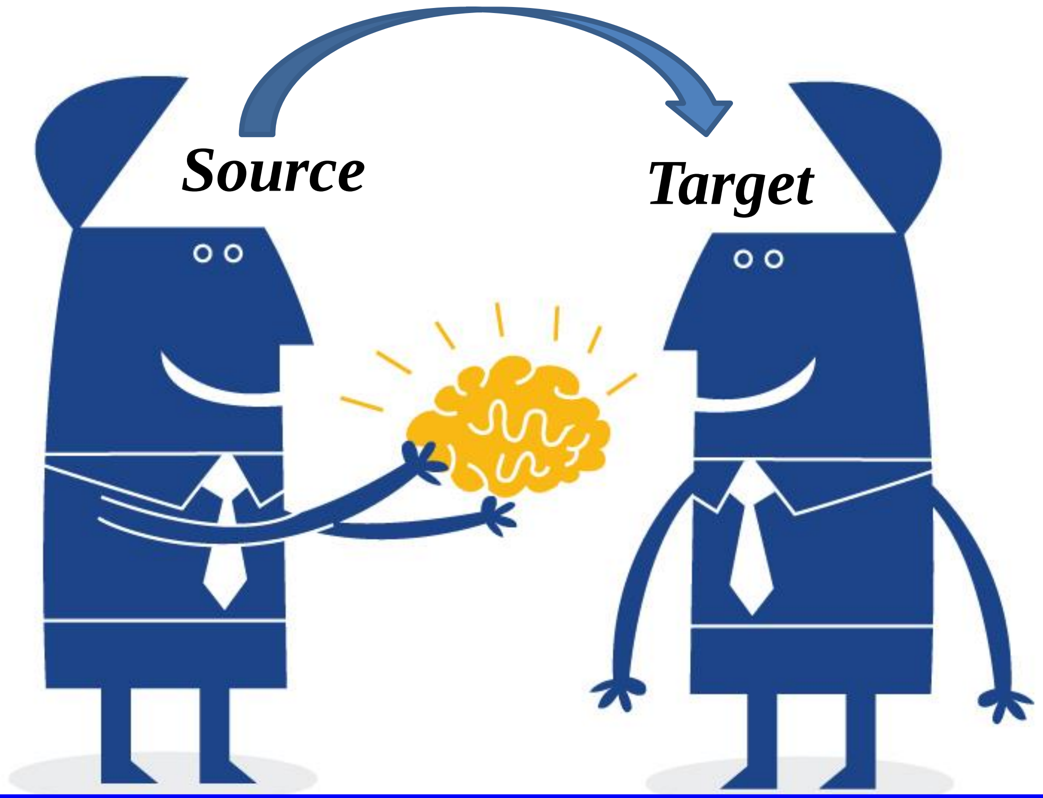
Background - The Idea

- If we are provided with two BNs, one source network (left) and one target (right) network.



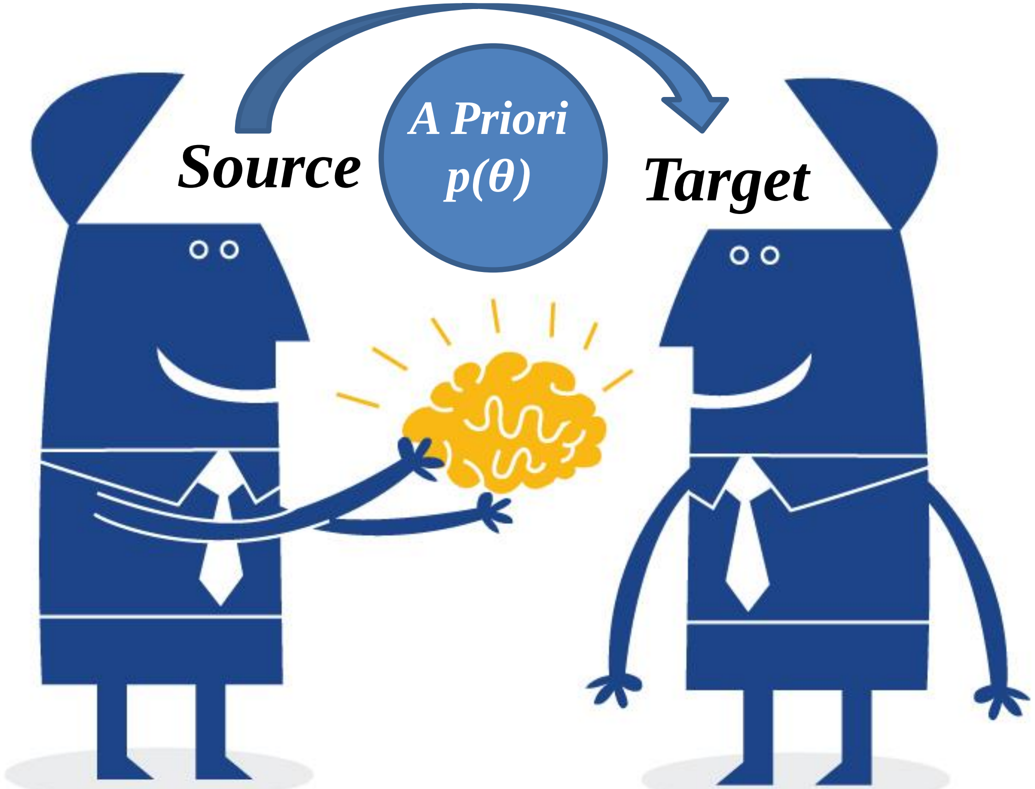
Background - The Idea

- We are interested in learning the target network parameter with the information in the source.



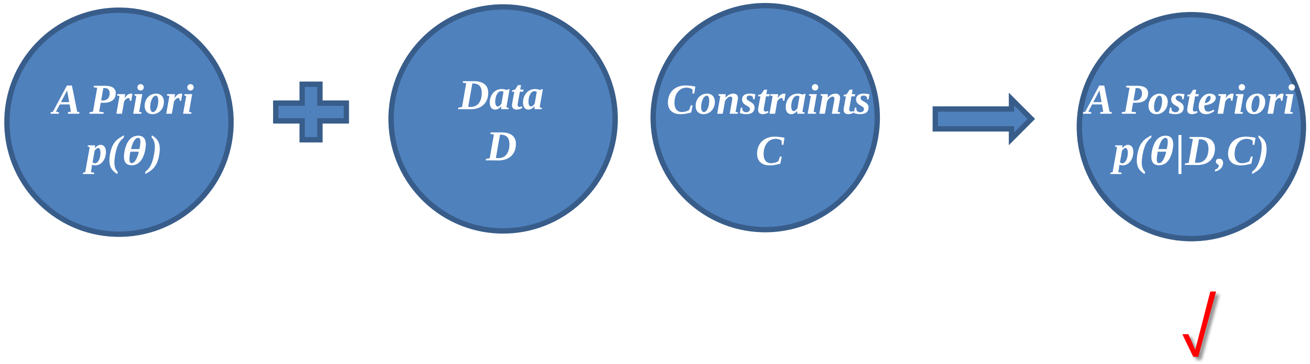
Background - The Idea

- By doing so, we use source data statistics to generate the target parameter prior.



Background - The Idea

- We update the target parameters with transferred prior, target data and target parameter constraints.



Overview

- Background
- Related Work
- The Model
- Experiments
- Conclusions

Related Work - The Basics

- Given data D, we can estimate the parameters θ with the help of the Bayes' Rule:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} = \frac{likelihood \cdot prior}{evidence} \quad (1)$$

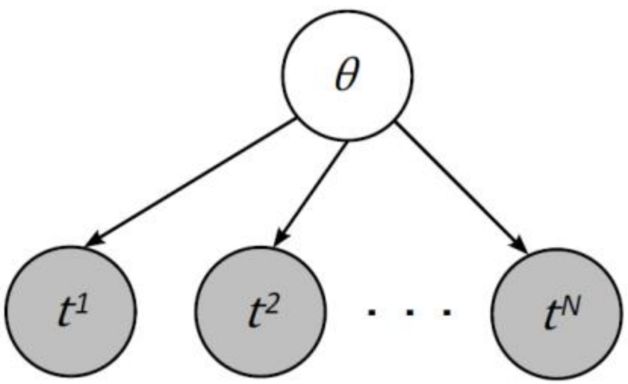
  - MLE
    - $\theta_{MLE} = \underset{\theta}{\operatorname{argmax}}\, logp(D|\theta)$
  - MAP
    - $\theta_{MAP} = \underset{\theta}{\operatorname{argmax}}(logp(D|\theta) + logp(\theta))$
  - Bayesian Estimation (BE)
    - $\theta_{BE} = p(\theta|D)$

Related Work - Constrained Parameter Learning

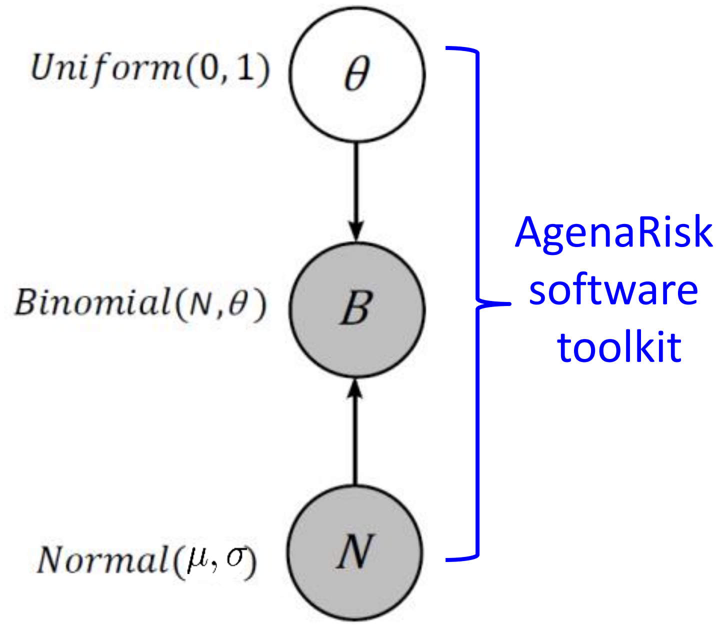
- MLE + Constrained convex optimization (CO)
  - Altendorf et al., 2005; Niculescu et al., 2006; de Campos and Ji, 2008; de Campos et al., 2008; Liao and Ji, 2009; de Campos et al., 2009; Yang and Natarajan, 2013.
  - $\underset{\theta}{\operatorname{argmax}}(logp(D|\theta) + penalty(\theta, C))$
- Bayesian Estimation + Constraints
  - Zhou et al., 2014a,b.
  - Multinomial Parameter Learning Model with Constraints (MPL-C)

Related Work - Constrained Parameter Learning

- MPL-C model
  - Learning as inference in auxiliary graphical models
  - Coin tossing problem



(a) The original representation



(b) The compact representation

Related Work - Parameter Transfer Learning

- Many works focus on structure transfer or multi-task learning.
  - Niculescu-mizil and Caruana, 2007; Oyen and Lane, 2012; Oates et al, 2014.
- CPTAgg
  - Luis et al., 2010 (a two-step framework).
    - 1) Measure the relatedness of tasks via calculating K-L divergence between target and source CPTs;
    - 2) Use a heuristic weighted sum model for aggregating target and selected source parameters.

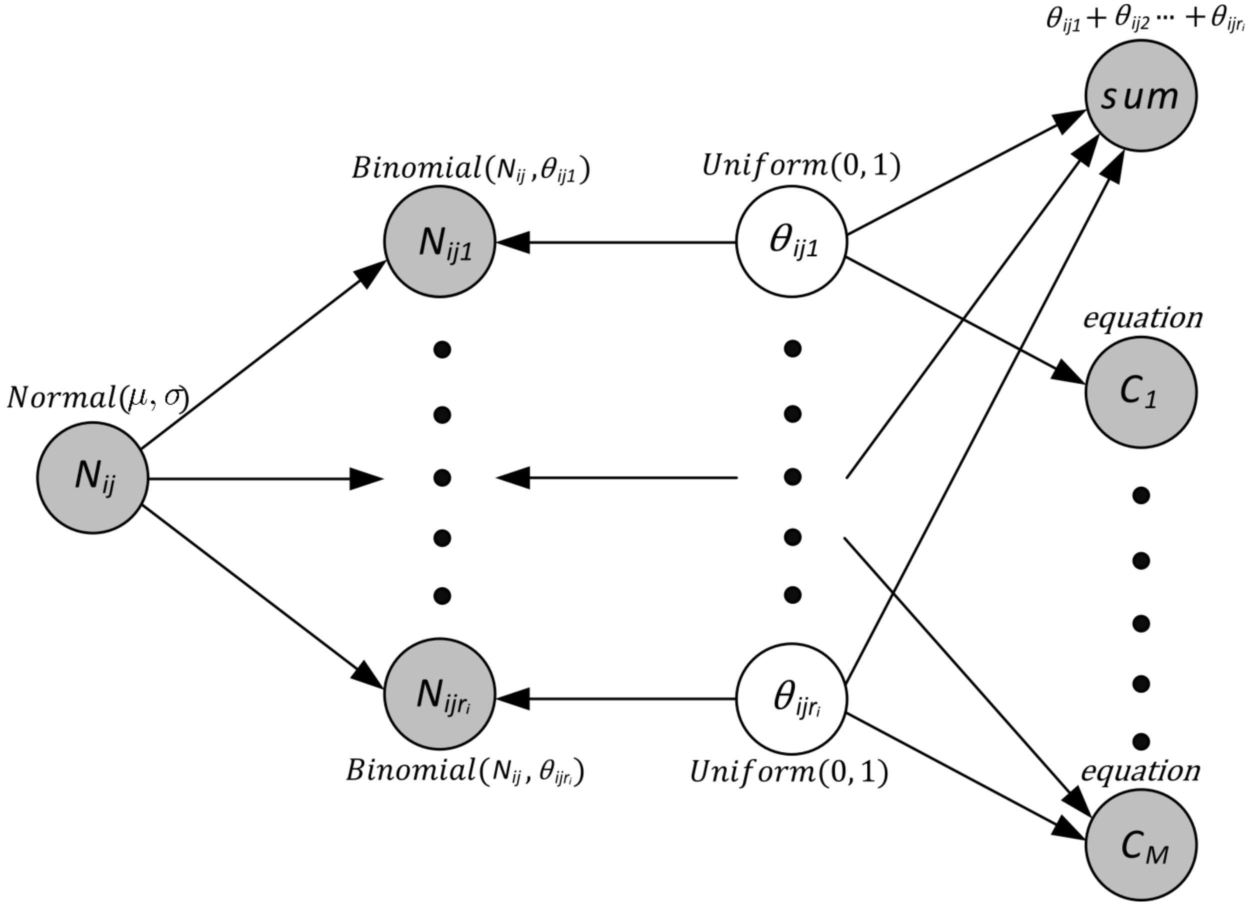
Related Work - Summary

- Either constraints or transferred information could improve parameter learning accuracy.
- No generic learning framework could synergistically exploit the benefits of both approaches.

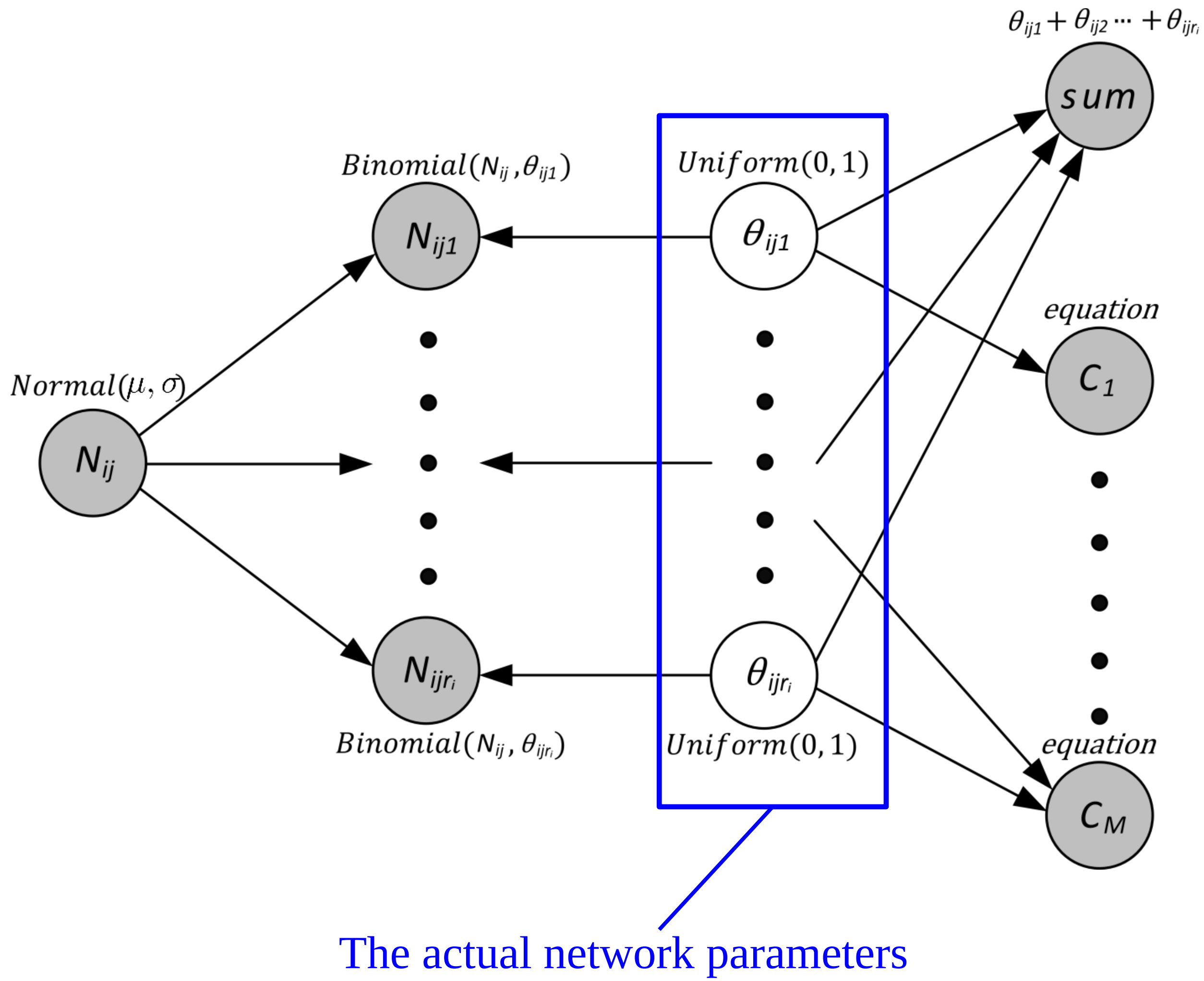
Overview

- Background
- Related Work
- The Model
- Experiments
- Conclusions

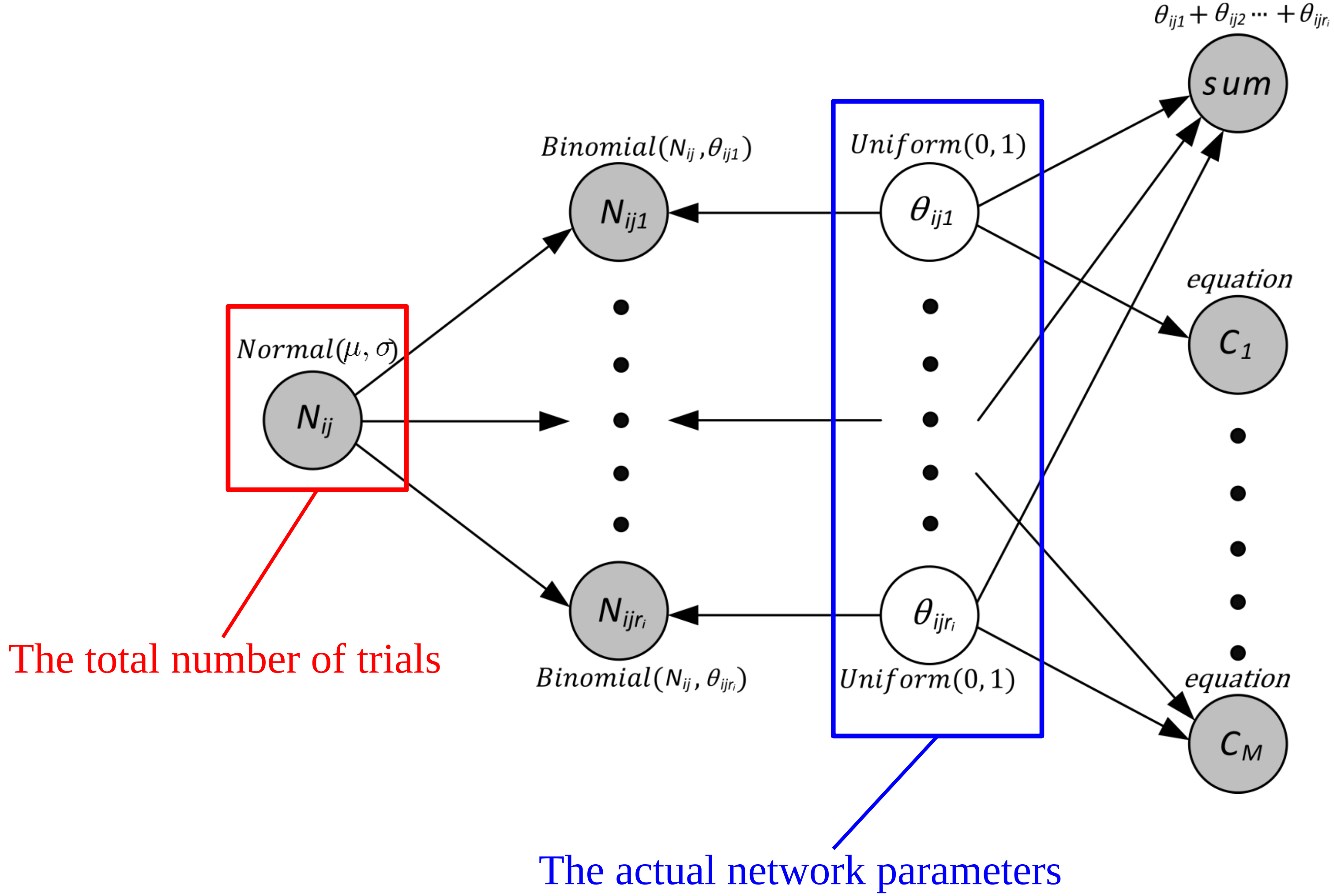
The Model - MPL-C



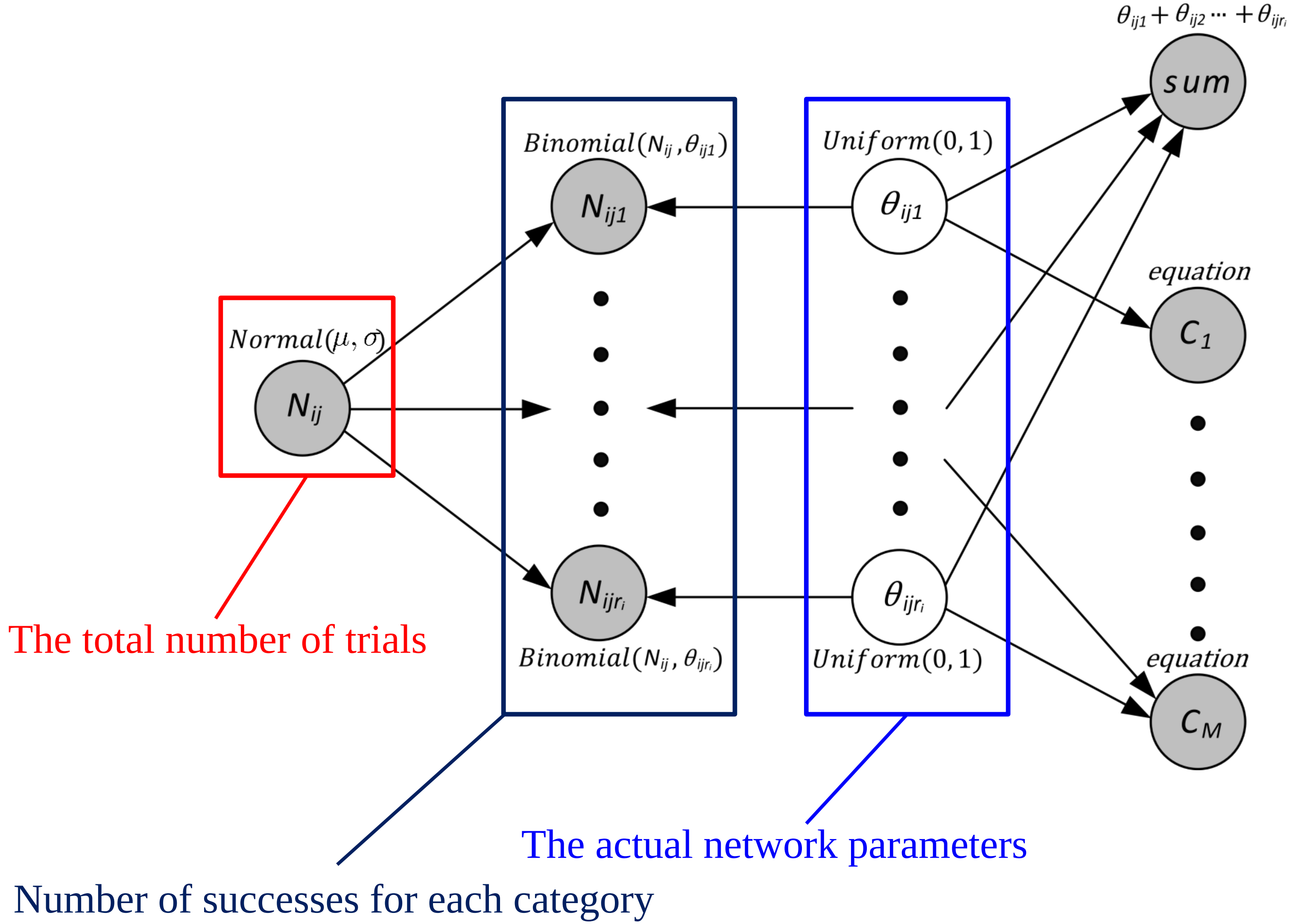
The Model - MPL-C



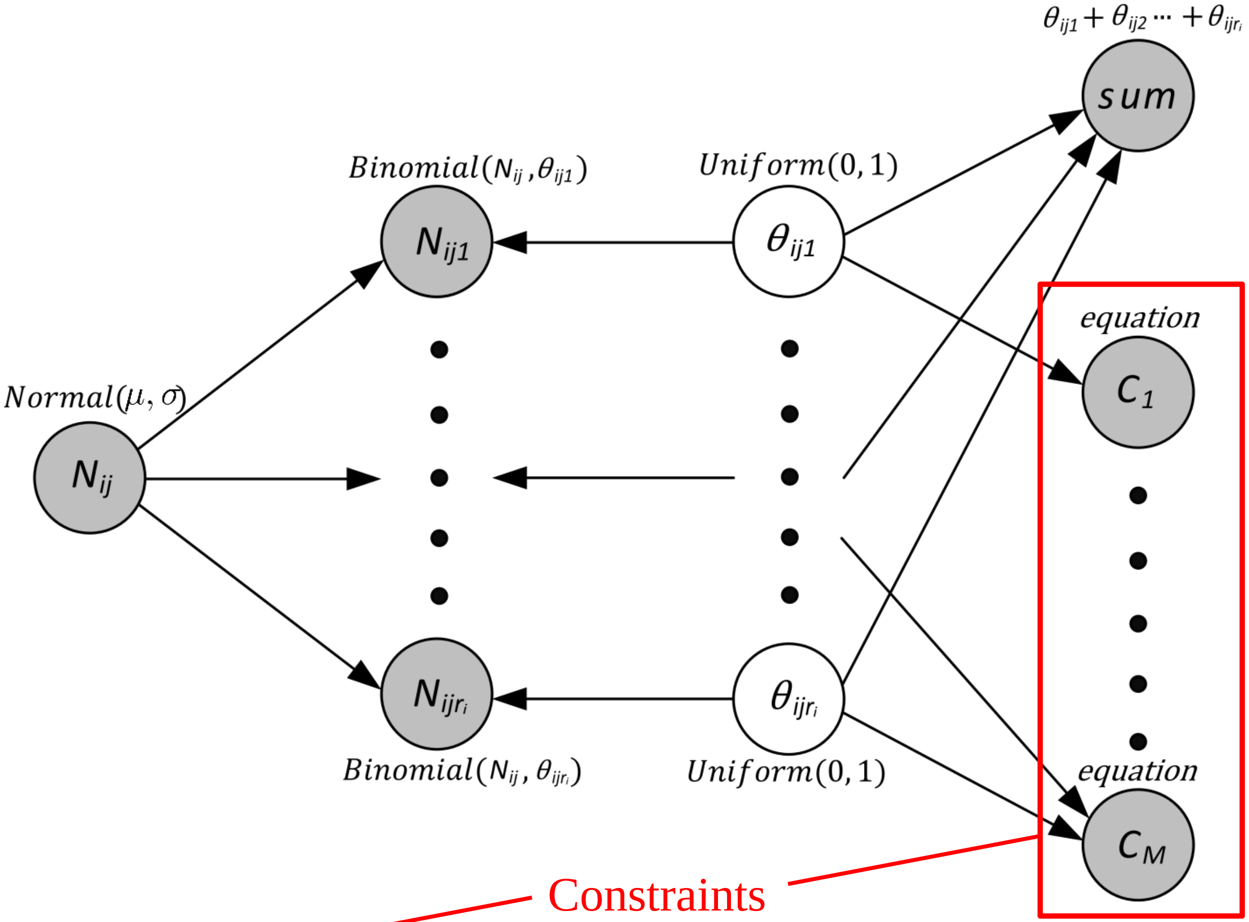
The Model - MPL-C



The Model - MPL-C

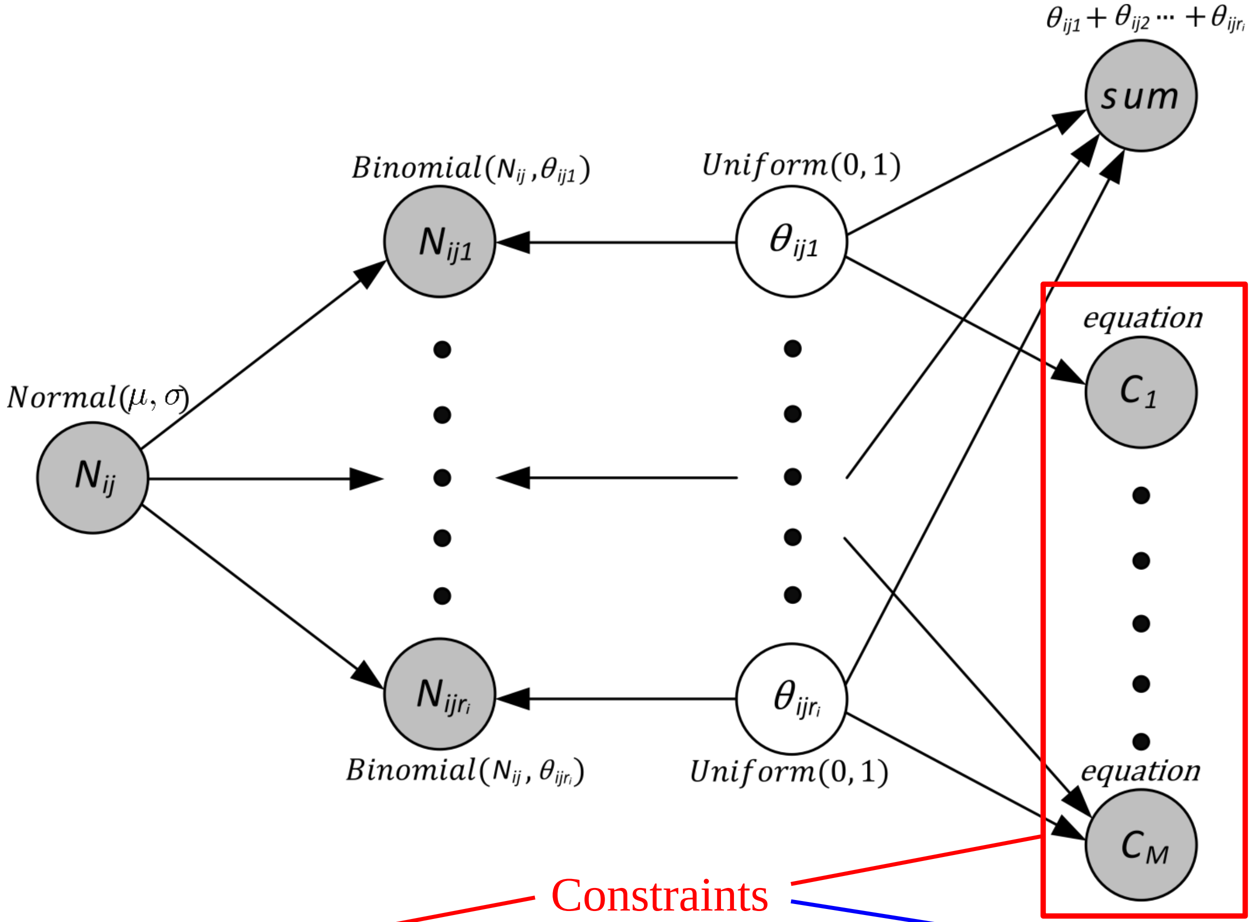


The Model - MPL-C



$$
\begin{cases}
\beta_0 + \sum_{k=1}^{r_i} \beta_k \theta_{ijk} \leq 0 \\
|\theta_{ijk} - \theta_{ijk'}| \leq \varepsilon \; (0 < \varepsilon < 1)
\end{cases}
$$

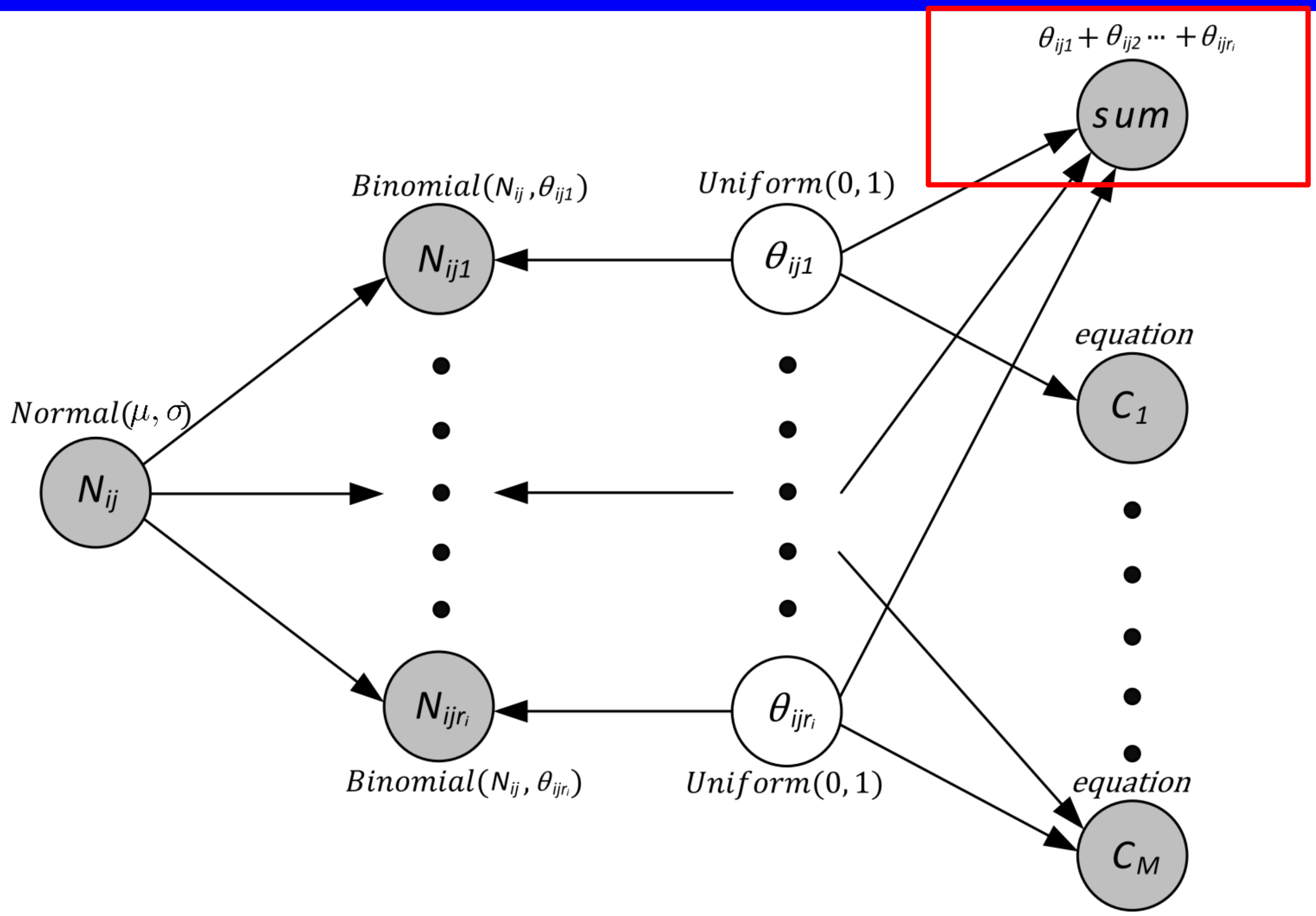
The Model - MPL-C



$$\begin{cases} \beta_0 + \sum_{k=1}^{r_i} \beta_k \theta_{ijk} \leq 0 \\ |\theta_{ijk} - \theta_{ijk'}| \leq \varepsilon \; (0 < \varepsilon < 1) \end{cases}$$

$$if(\beta_0 + \sum_{k=1}^{r_i} \beta_k \theta_{ijk} \leq 0, \; true, \; false)$$
$$if(abs(\theta_{ijk} - \theta_{ijk'}) \leq \varepsilon, \; true, \; false)$$

The Model - MPL-C



The Model - MPL-TC

- We extend MPL-C model with transferred parameter prior.

- Notations and definitions
  - Problem domain $\boldsymbol{\mathcal{D}} = \{V, G, D\}$
  - BN fragment $\boldsymbol{\mathcal{D}}_i = \{V_i, G_i, D_i\}$ is a single root node or a node with its direct parents in the original BN.
  - Target domain $\boldsymbol{\mathcal{D}}^t = \{\boldsymbol{\mathcal{D}}_i^t\}$
  - Source domains $\{\boldsymbol{\mathcal{D}}^s\} = \{\{\boldsymbol{\mathcal{D}}_{i'}^s\}\}$

The Model - MPL-TC

- Assumptions
  - We don’t assume corresponding structure or variable names.
  - There are multiple potential sources of varying relevance.
  - At least one of the sources is sampled from similar distributions as the target.

The Model - Three Challenges in Transfer

- Which source fragments are transferrable?
  - Check fragment compatibility.
- How to deal with variable name mapping?
  - Try all fragment permutation mappings.
- How to quantify the relatedness of each transferrable source fragment?
  - Use fitness measurement to find the best one.

The Model - 1) Fragment Compatibility

- For a target fragment *i* and putative source fragment *i'*, we say they are *compatible* if they have the same structure and state space.

$$compatible(\boldsymbol{\mathcal{D}}_i^t, \boldsymbol{\mathcal{D}}_{i'}^s) = \begin{cases} 1 & if\ G_i^t = G_{i'}^s\ \&\ \boxed{\theta_{i'}^s \in \Omega_{C_i^t}} \\ & \quad \&\ dim(\theta_i^t) = dim(\theta_{i'}^s) \\ 0 & otherwise \end{cases}$$

where $dim(\theta_i^t) = dim(\theta_{i'}^s)$ means $r_i^t = r_{i'}^s$ and $|\pi_i^t| = |\pi_{i'}^s|$

The Model - 2) Fragment Permutation Mapping

- In transfer, we may not know the mapping between variable names.
- For example, if target fragment *i* has parents [a, b] and source fragment *i'* has parents [d, c], the correspondence could be a − d, b − c or b − d, a − c.
- We exhaustively list possible mappings P_m that map states of *i* to states of *i'*.

The Model - 3) Fitness Measurement

- Bayesian model comparison for two hypotheses:
  - H1 - The relevance hypothesis that the source and target data share a common CPT.

$$p(H_1|D^s_{i'}, D^t_i) \propto \int p(D^t_i|\theta_i)p(\theta_i|D^s_{i'}, H_1)p(H_1)d\theta_i$$

$$p(D^t_i|D^s_{i'}, H_1) = \sum_{j=1}^{|\pi_i|} \left(\frac{\Gamma(\alpha^s_{i'j})}{\Gamma(N^t_{ij} + \alpha^s_{i'j})} \prod_{k=1}^{r_i} \frac{\Gamma(N^t_{ijk} + \alpha^s_{i'jk})}{\Gamma(\alpha^s_{i'jk})} \right)$$

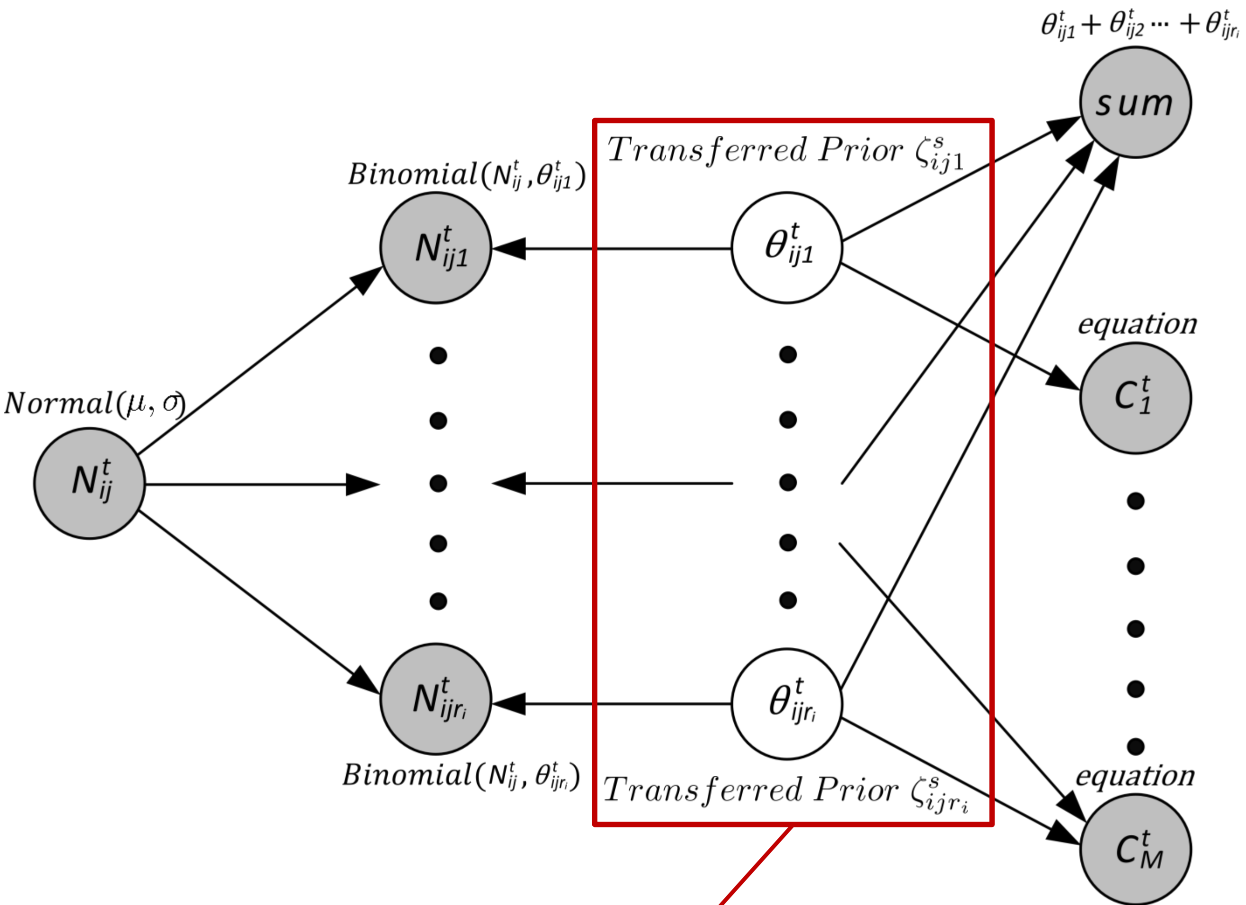
  - H0 - The independent hypothesis that the source and target data have distinct CPTs.

$$p(H_0|D^s_{i'}, D^t_i) = 1 - p(H_1|D^s_{i'}, D^t_i)$$

The Model - Generate Prior via Bootstrap

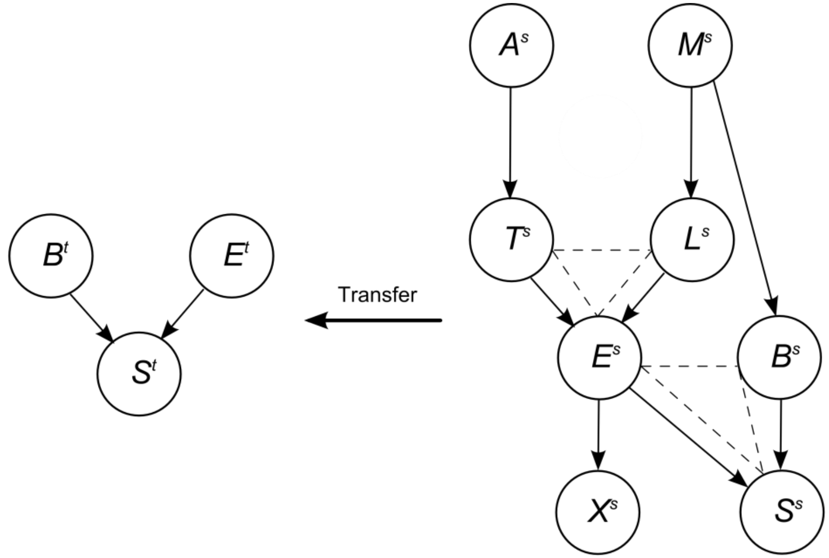
- We use selected best mapping source sample $D^s_{i'j}$ to generate the prior distributions of parameters in the target MPL-C model:
  - 1) Use bootstrap method to resample to form a new source data sample (a bootstrap sample);
  - 2) Repeat multiple times (100 or 1000);
  - 3) For each of these bootstrap samples, we compute the MLE of parameter $\theta^s_{i'jk}$;
  - 4) Fit a *TNormal* distribution ($\zeta^s_{i'jk}$) to the set of MLE values.

The Model - MPL-TC

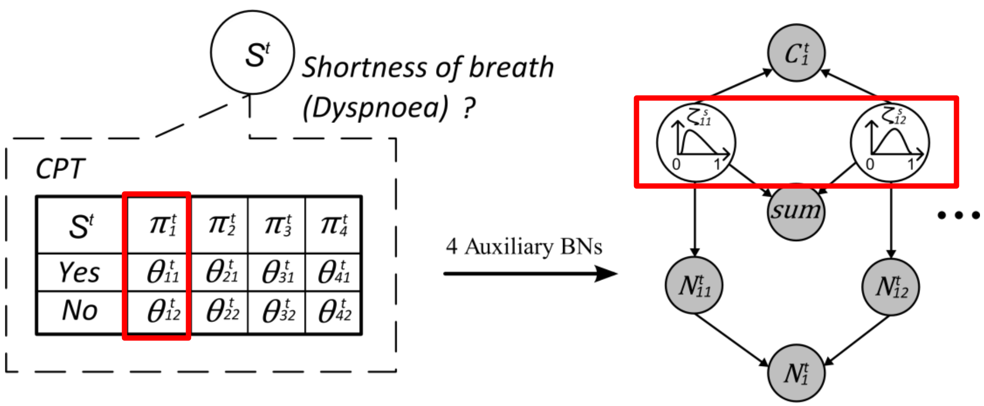


Transferred informative parameter priors

The Model - Example



(a)



S^t	π_1^t	π_2^t	π_3^t	π_4^t
Yes	θ_{11}^t	θ_{21}^t	θ_{31}^t	θ_{41}^t
No	θ_{12}^t	θ_{22}^t	θ_{32}^t	θ_{42}^t

(b)

The Model - Inference

Parameter posteriors in target BN

$$p(\hat{\theta}^t_{ij1}, \ldots, \hat{\theta}^t_{ijr_t} | N^t_{ij}, N^t_{ij1}, \ldots, N^t_{ijr_i}, C^t_1, \ldots, C^t_M, \zeta^s_{i'jk}, \ldots, \zeta^s_{i'jr_{i'}}, sum)$$

Total number of trials

Number of successes observed in the target samples

Constraints on target parameters

TNormal sufficient statistics

sum =1

- Dynamic discretization junction tree (DDJT) algorithm (Neil et al., 2007).
  - This algorithm uses the relative entropy error to iteratively adjust the discretization in response to new evidence, and so achieves more accuracy in the zones of high posterior density.

Overview

- Background
- Related Work
- The Model
- Experiments
- Conclusions

Experiments - Setting

- Source data - two noises are introduced during sampling:
  - `Soft' noise - generate three source domains with 200, 300 and 400 sample sizes to simulate continuously varying relatedness among a set of sources.
  - `Hard' noise - choose a portion (20%) of each source's fragments uniformly at random and randomise their data/CPTs to make them irrelevant.
- Target constraints:

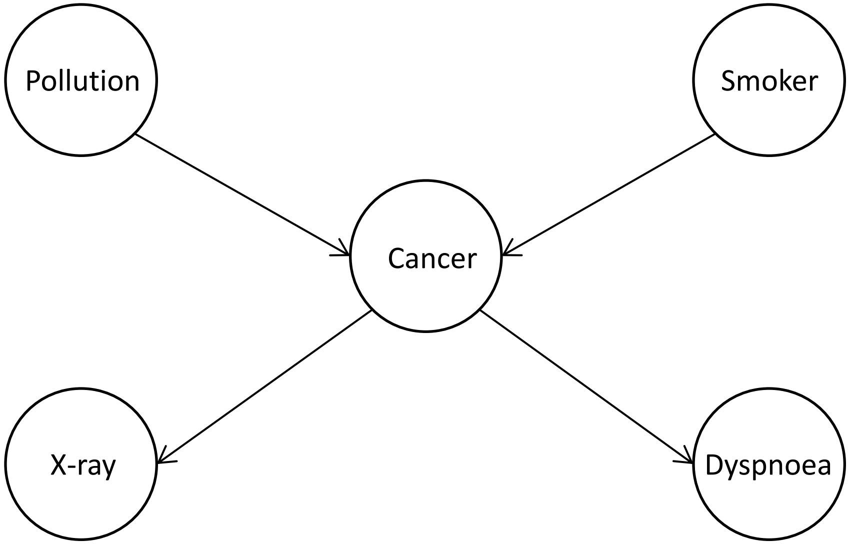
$$\min((1+\varepsilon)\theta_{ijk}, 1) \geq \theta^{t}_{ijk} \geq \max((1-\varepsilon)\theta_{ijk}, 0)$$

Experiments - Setting

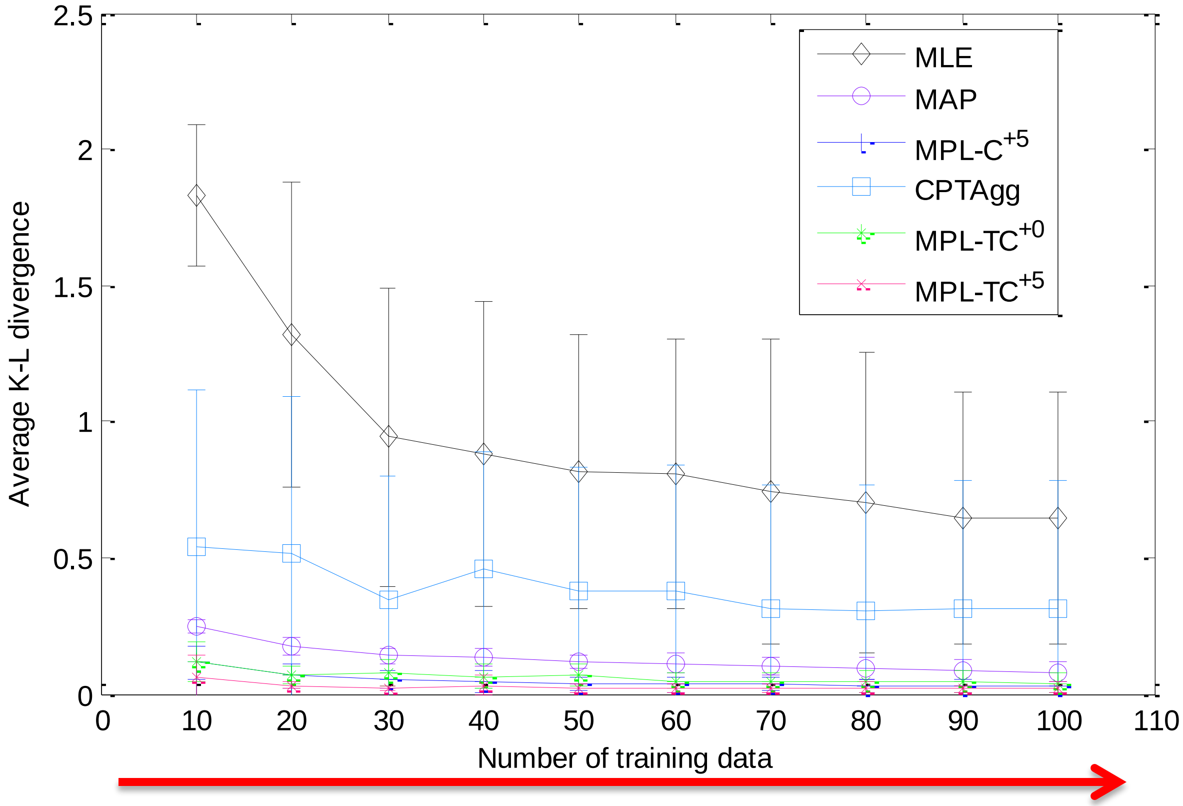
- Matlab BNT toolbox
  - https://code.google.com/p/bnt/
- MPL-TC model is built with AgenaRisk API
  - http://www.agenarisk.com/products/freedownload

Experiments - Cancer BN

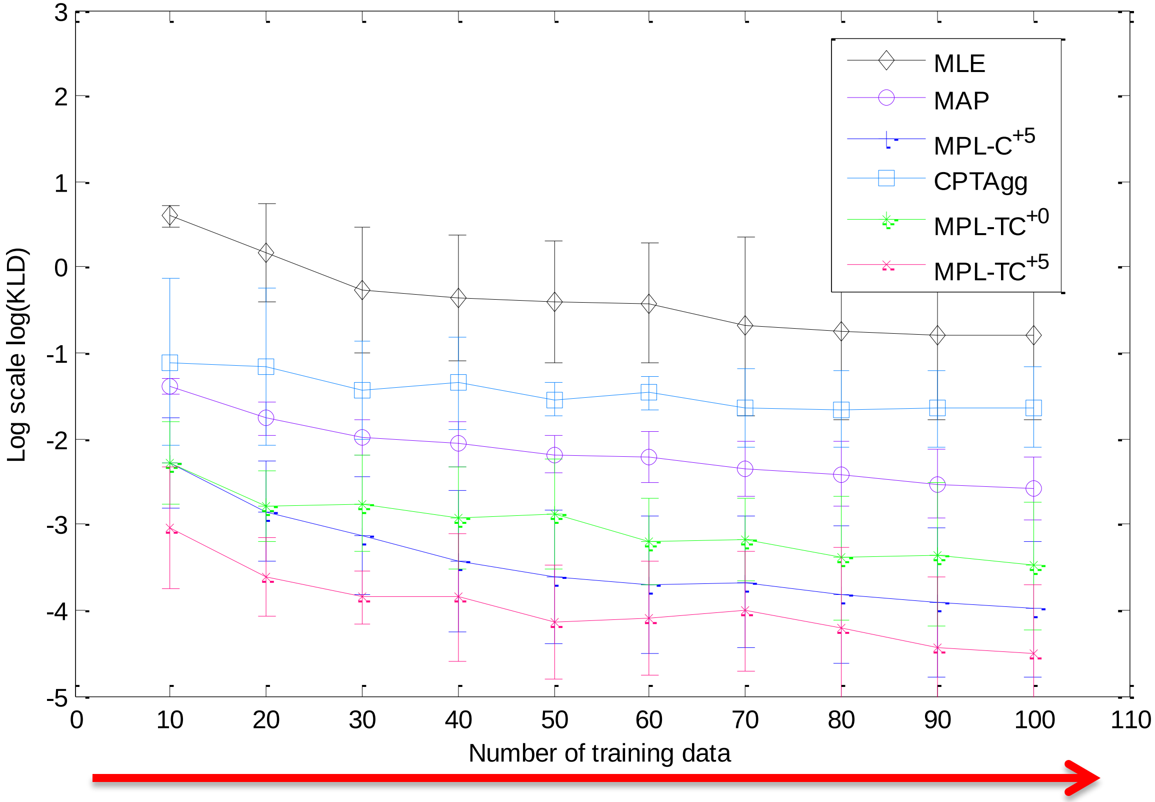
- The Cancer BN (Korb and Nicholson, 2010) models the interaction between risk factors and symptoms for the purpose of diagnosing the most likely condition for a patient getting lung cancer.



Experiments - Varying Data Sizes

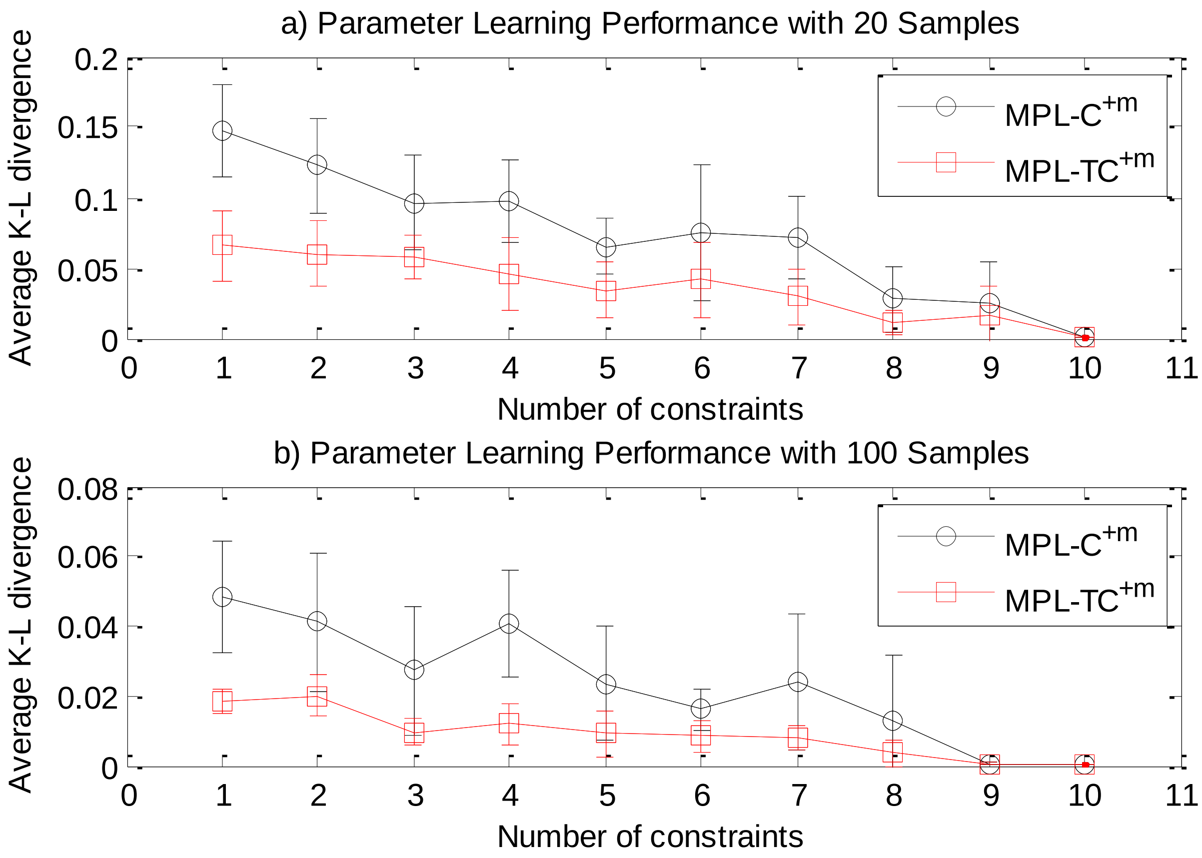


Experiments - Varying Data Sizes (Log scale)

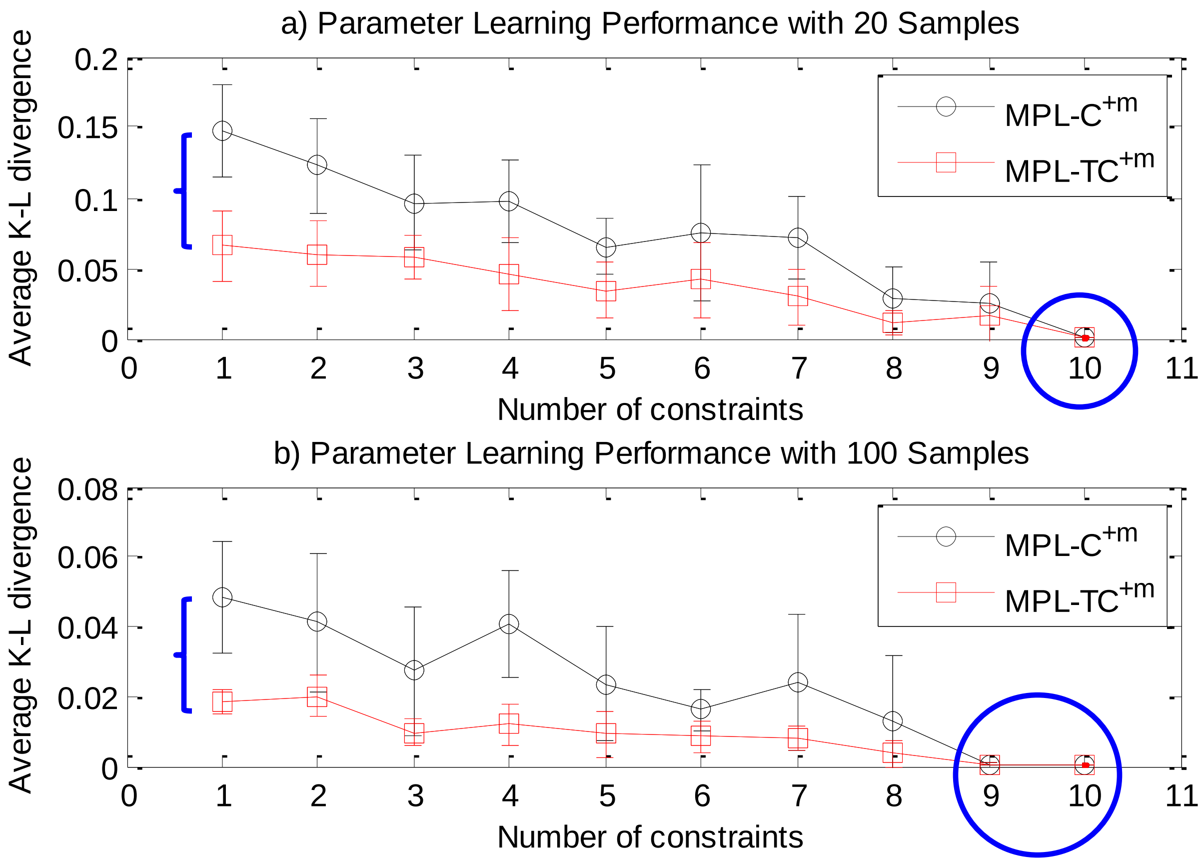


- MPL-TC^{+5} greatly outperforms the conventional MLE and MAP algorithms, and the CPTAgg and MPL-C^{+5} that only use transfer or constraints alone.

Experiments - Varying Number of Constraints

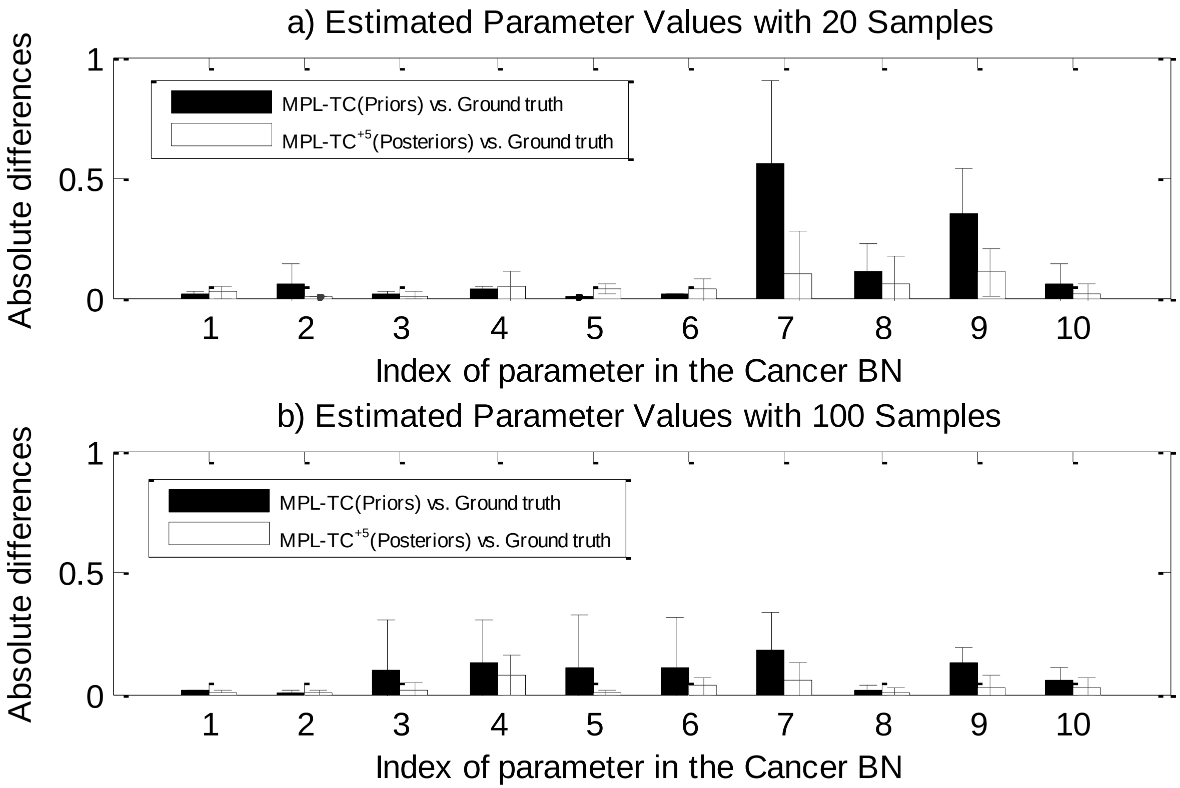


Experiments - Varying Number of Constraints

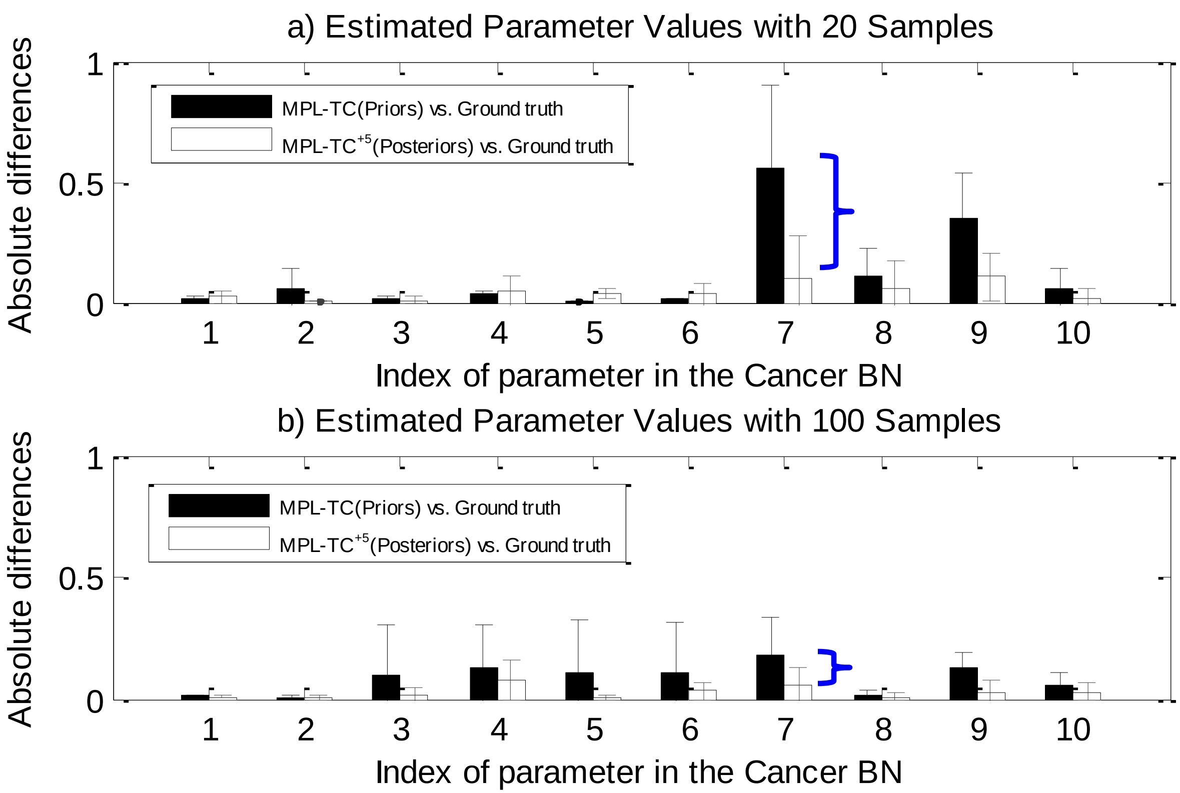


- Both MPL-TC and MPL-C can be improved with increased number of constraints.
- MPL-TC always beats the method without transfer (MPL-C).

Experiments - Priors vs. Posteriors



Experiments - Priors vs. Posteriors



- Compared with MPL-TC(Priors), MPL-TC^{+5}(Posteriors) achieves better results with the regularization of introduced constraints.

Experiments - Standard BNs

- We evaluate the algorithms on 12 standard BNs.
  - http://www.bnlearn.com/bnrepository/
- Parser bif2bnt
- For each target BN, we generate:
  - 100 training samples;
  - 5 constraints.

Experiments - Standard BNs

Table 1: Parameter learning performance (average K-L divergence) in 12 standard Bayesian networks.

Name	Nodes	Edges	Para	MLE	MAP	MPL-C^{+5}	CPTAgg	MPL-TC^{+0}	MPL-TC^{+5}
Alarm	37	46	509	2.36±0.10*	0.66±0.01*	0.61±0.02*	1.61±0.08*	**0.42** ±0.02	**0.42** ±0.01
Andes	223	338	1157	1.03±0.06*	0.17±0.01*	0.15±0.01*	0.65±0.05*	**0.08** ±0.00	**0.08** ±0.00
Asia	8	8	18	0.57±0.16*	0.34±0.04*	0.28±0.03*	0.31±0.05*	0.22±0.02*	**0.18** ±0.03
Cancer	5	4	10	0.86±0.35*	0.09±0.04*	0.07±0.05*	0.54±0.11*	0.05±0.01*	**0.03** ±0.01
Earthquake	5	4	10	1.50±0.82*	0.15±0.04*	0.13±0.03*	0.35±0.22*	0.11±0.01	**0.10** ±0.01
Hailfinder	56	66	2656	2.85±0.01*	0.46±0.00*	0.41±0.00*	1.98±0.01*	**0.31** ±0.01	**0.31** ±0.01
Hepar2	70	123	1453	3.18±0.13*	0.33±0.01*	0.33±0.01*	2.58±0.15*	0.30±0.01	**0.29** ±0.00
Insurance	27	52	984	1.95±0.18*	1.17±0.03*	1.07±0.03*	0.93±0.06*	**0.75** ±0.03	**0.75** ±0.02
Sachs	11	17	178	1.74±0.29*	0.78±0.04*	0.71±0.05*	0.98±0.08*	**0.50** ±0.03	**0.50** ±0.02
Survey	6	6	21	0.35±0.20*	0.05±0.01*	0.05±0.01*	0.24±0.15*	0.04±0.01	**0.03** ±0.01
Weather	4	4	9	**0.02**±0.02	0.03±0.00	**0.02**±0.00	**0.02**±0.00	**0.02**±0.00	**0.02**±0.00
Win95pts	76	112	574	3.59±0.07*	0.81±0.01*	0.78±0.02*	3.20±0.10*	0.67±0.02*	**0.64** ±0.01

Overview

- Background
- Related Work
- The Model
- Experiments
- Conclusions

Conclusions

- Findings
  - This is the first attempt at BN parameter learning with both transferred prior and qualitative constraints.
  - Improved learning performance is observed across a range of networks.
- Limitations
  - Only most relevant source is transferred.
  - Data-driven transfer (source selection) may be biased by inaccurate target data.
  - Not robust to totally irrelevant sources.