KU LEUVEN

GeSDD: Learning of tractable SDDs using genetic algorithms

Michiel Baptist
KU Leuven
September 2019

1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms
2. Problem statement
3. GeSDD
4. Results

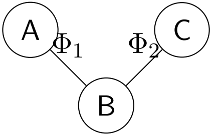
1 Outline

1 Markov Logic Network (MLN)

- Compact representation of a distribution over $\mathbf{X} = (X_1, \ldots, X_n)$:

1 Markov Logic Network (MLN)

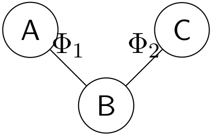
- Compact representation of a distribution over $\mathbf{X} = (X_1, \ldots, X_n)$:



$$\boxed{\Phi_i(\mathbf{X}) = \exp\left(w_i f_i(\mathbf{X})\right)}$$

1 Markov Logic Network (MLN)

- Compact representation of a distribution over $\mathbf{X} = (X_1, \ldots, X_n)$:



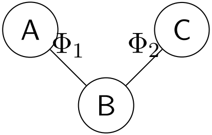
$$\boxed{\Phi_i(\mathbf{X}) = \exp\left(w_i f_i(\mathbf{X})\right)}$$

- MLN as a collection of feature-weight pairs:

$$M = \{(f_1, w_1), \ldots, (f_m, w_m)\} \quad =$$

MLN	
w_1	f_1
$\vdots$	$\vdots$
w_m	f_m

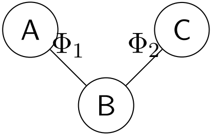
1 Markov Logic Network (MLN)



$=$

MLN	
w_1	f_1
$\vdots$	$\vdots$
w_n	f_n

1 Markov Logic Network (MLN)



=

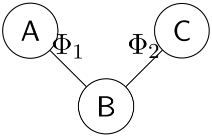
MLN	
w_1	f_1
$\vdots$	$\vdots$
w_n	f_n

- Meaning:

$$P(\mathbf{X} = x) = \frac{1}{Z} \prod_i \exp\left(w_i f_i(x)\right) \tag{1}$$

$$Z = \sum_{x'} \prod_i \exp\left(w_i f_i(x')\right) \tag{2}$$

1 Markov Logic Network (MLN)



$=$

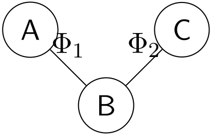
MLN	
w_1	f_1
$\vdots$	$\vdots$
w_n	f_n

- Meaning:

$$P(\mathbf{X} = x) = \frac{1}{Z} \prod_i \exp\Big(w_i f_i(x)\Big) \tag{1}$$

$$Z = \sum_{x'} \prod_i \exp\Big(w_i f_i(x')\Big) \tag{2}$$

1 Markov Logic Network (MLN)



$=$

MLN	
w_1	f_1
	$\vdots$
	f_n

- Meaning:

$$P(\mathbf{X} = x) = \frac{1}{Z} \prod_i \exp\Big(w_i f_i(x)\Big) \tag{1}$$

$$Z = \sum_{x'} \prod_i \exp\Big(w_i f_i(x')\Big) \tag{2}$$

KU LEUVEN

1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms

1 Weighted Model Counting (WMC)

- Seemingly unrelated concept: Weighted Model Counting (WMC)

1 Weighted Model Counting (WMC)

- Seemingly unrelated concept: Weighted Model Counting (WMC)

Given:

- Boolean function $f : \Omega \to \{0,1\}$ with $\Omega = \{0,1\}^m$

1 Weighted Model Counting (WMC)

- Seemingly unrelated concept: Weighted Model Counting (WMC)

Given:

- Boolean function $f : \Omega \rightarrow \{0, 1\}$ with $\Omega = \{0, 1\}^m$
- Weighted assignments $W(X_i = x_i) \in \mathcal{R}$

1 Weighted Model Counting (WMC)

- Seemingly unrelated concept: Weighted Model Counting (WMC)

Given:

- Boolean function $f : \Omega \rightarrow \{0, 1\}$ with $\Omega = \{0, 1\}^m$
- Weighted assignments $W(X_i = x_i) \in \mathcal{R}$

$$
\begin{aligned}
WMC(f) &= \sum_{x \in \Omega : f = 1} W(\mathbf{X} = x) \\
&= \sum_{x \in \Omega : f = 1} \prod_{i=1}^{m} W(X_i = x_i)
\end{aligned}
\tag{3}
$$

KU LEUVEN

1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms

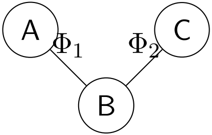
1 MLN encoding

Problem: Multivariate inference (often) is hard[3] ⚠

1 MLN encoding

Problem: Multivariate inference (often) is hard[3]

Reduction to: WMC [2]



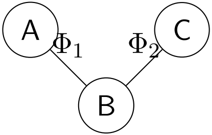
$\xrightarrow{\text{Encoding}}$

$$\boxed{\begin{array}{l} f_1 \Leftrightarrow I_{f_1} \\ f_2 \Leftrightarrow I_{f_2} \end{array}}$$

1 MLN encoding

Problem: Multivariate inference (often) is hard[3] ⚠

Reduction to: WMC [2]



$$\xrightarrow{\text{Encoding}} \boxed{\begin{array}{l} f_1 \Leftrightarrow I_{f_1} \\ f_2 \Leftrightarrow I_{f_2} \end{array}}$$

$$Z = WMC\left(\boxed{\begin{array}{l} f_1 \Leftrightarrow I_{f_1} \\ f_2 \Leftrightarrow I_{f_2} \end{array}} \right)$$

1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms

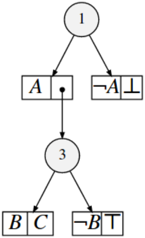
1 Sentential Decision Diagrams (SDDs)

1 Sentential Decision Diagrams (SDDs)

$$f(A, B, C) = A \wedge (B \Rightarrow C)$$

1 Sentential Decision Diagrams (SDDs)

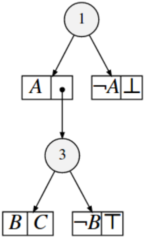
$$f(A, B, C) = A \land (B \Rightarrow C) \xrightarrow{\text{Compilation}}$$



KU LEUVEN

1 Sentential Decision Diagrams (SDDs)

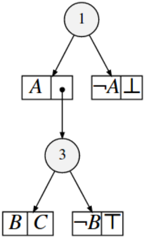
$$f(A, B, C) = A \wedge (B \Rightarrow C) \xrightarrow{\text{Compilation}}$$



$\rightarrow$ 1 Disjunction

1 Sentential Decision Diagrams (SDDs)

$$f(A, B, C) = A \wedge (B \Rightarrow C) \xrightarrow{\text{Compilation}}$$

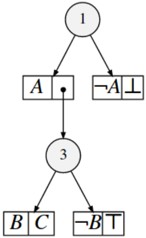


$\rightarrow$  Disjunction

$\rightarrow$ $\boxed{B \mid C}$ Conjunction

1 Sentential Decision Diagrams (SDDs)

$$f(A,B,C) = A \land (B \Rightarrow C) \quad \xrightarrow{\text{Compilation}}$$



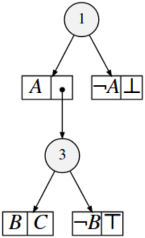
$\rightarrow$ 1 Disjunction

$\rightarrow$ $\boxed{B \mid C}$ Conjunction

SDDs have important properties:

1 Sentential Decision Diagrams (SDDs)

$$f(A, B, C) = A \land (B \Rightarrow C) \xrightarrow{\text{Compilation}}$$



$\rightarrow$ 1 Disjunction

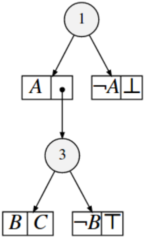
$\rightarrow$ $\boxed{B \mid C}$ Conjunction

SDDs have important properties:

- Polytime operations!

1 Sentential Decision Diagrams (SDDs)

$$f(A, B, C) = A \wedge (B \Rightarrow C) \xrightarrow{\text{Compilation}}$$



$\rightarrow$ 1 Disjunction

$\rightarrow$ $\boxed{B \mid C}$ Conjunction

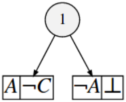
SDDs have important properties:

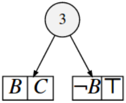
- Polytime operations!
- Enable Weighted Model Counting!

1 SDDs: Polytime operations

1 SDDs: Polytime operations

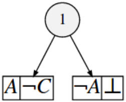
$A \wedge \neg C$ $\qquad$ $B \Rightarrow C$



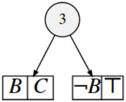


1 SDDs: Polytime operations

$$\left(\quad A \wedge \neg C \quad \wedge \quad B \Rightarrow C \quad \right) =$$



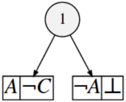
$\wedge$

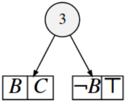


$=$

1 SDDs: Polytime operations

$$\left(\quad A \wedge \neg C \quad \wedge \quad B \Rightarrow C \quad \right) \quad = \quad A \wedge \neg B \wedge \neg C$$

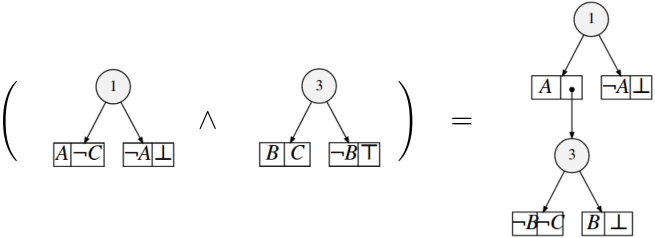
$\Bigg($ 

$\wedge$ 

$\Bigg) \quad =$

1 SDDs: Polytime operations

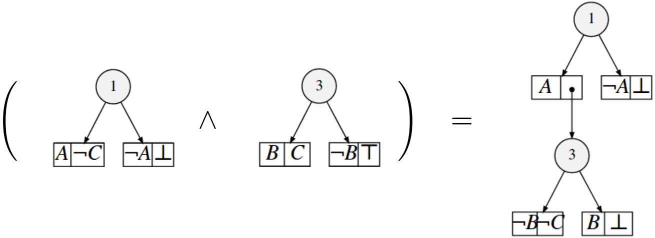
$$\left(\quad A \land \neg C \qquad \land \qquad B \Rightarrow C \quad \right) \quad = \quad A \land \neg B \land \neg C$$



KU LEUVEN

1 SDDs: Polytime operations

$$\left(\quad A \wedge \neg C \quad \wedge \quad B \Rightarrow C \quad \right) \quad = \quad A \wedge \neg B \wedge \neg C$$



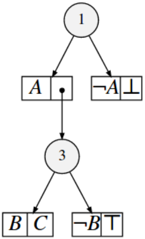
$$\boxed{\alpha \circ \beta \text{ can be done in } \mathcal{O}(|\alpha||\beta|)}$$

1 SDDs: Weighted Model Counting

$\boxed{\text{SDDs allow us to model count!}}$

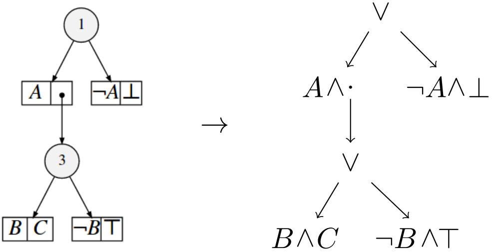
1 SDDs: Weighted Model Counting

SDDs allow us to model count!



1 SDDs: Weighted Model Counting

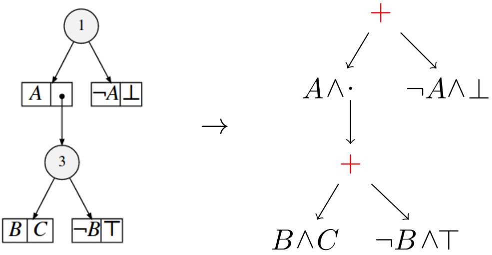
SDDs allow us to model count!



KU LEUVEN

1 SDDs: Weighted Model Counting

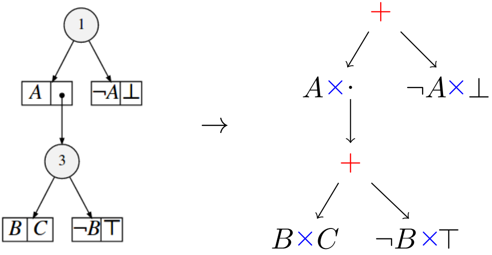
SDDs allow us to model count!



- Replace 1 by $+$

1 SDDs: Weighted Model Counting

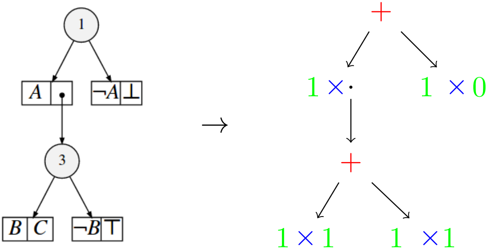
SDDs allow us to model count!



- Replace 1 by $+$
- Replace $\boxed{B \mid C}$ by $\times$

1 SDDs: Weighted Model Counting

$\boxed{\text{SDDs allow us to model count!}}$



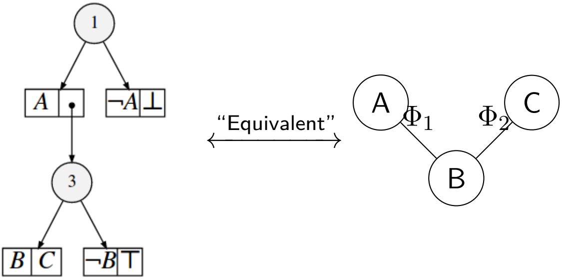
- Replace 1 by $+$
- Replace $\boxed{B \mid C}$ by $\times$
- Replace literals by 1 or 0 if $\perp$

1 SDDs as a distribution

- Since SDDs allow WMC, we can compute Z!

1 SDDs as a distribution

- Since SDDs allow WMC, we can compute Z!
- Since we can compute Z, we can compute the distribution!



1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms

1 Learning MLNs and SDDs

Learning MLNs and their SDDs from data $\mathcal{X}$:

1 Learning MLNs and SDDs

Learning MLNs and their SDDs from data $\mathcal{X}$:

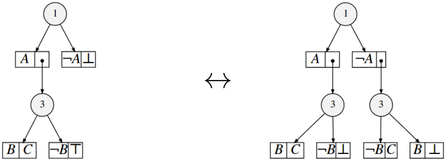
1. Feature learning: $f = (f_1, \ldots, f_m)$

1 Learning MLNs and SDDs

Learning MLNs and their SDDs from data $\mathcal{X}$:

1. Feature learning: $f = (f_1, \ldots, f_m)$

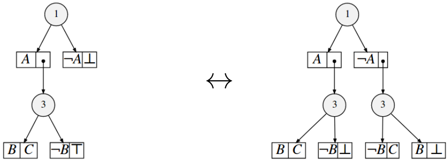
- Directly using the encoding



1 Learning MLNs and SDDs

Learning MLNs and their SDDs from data $\mathcal{X}$:

1. **Feature learning:** $f = (f_1, \ldots, f_m)$
   - Directly using the encoding

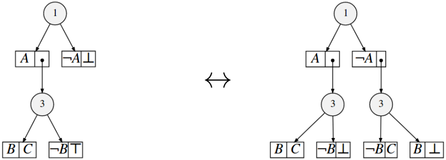


2. **Weight learning:** $w = (w_1, \ldots, w_m)$

1 Learning MLNs and SDDs

Learning MLNs and their SDDs from data $\mathcal{X}$:

1. **Feature learning:** $f = (f_1, \ldots, f_m)$
   - Directly using the encoding



2. **Weight learning:** $w = (w_1, \ldots, w_m)$
   - MLN should reflect given data

$$\langle MLN \rangle \approx \mathcal{X}_p \tag{4}$$

1 Outline

1. Background
   - Markov Logic Network (MLN)
   - Weighted Model Counting (WMC)
   - MLN encoding
   - Sentential Decision Diagrams (SDDs)
   - Learning MLNs and SDDs
   - Genetic Algorithms

1 Genetic Algorithms

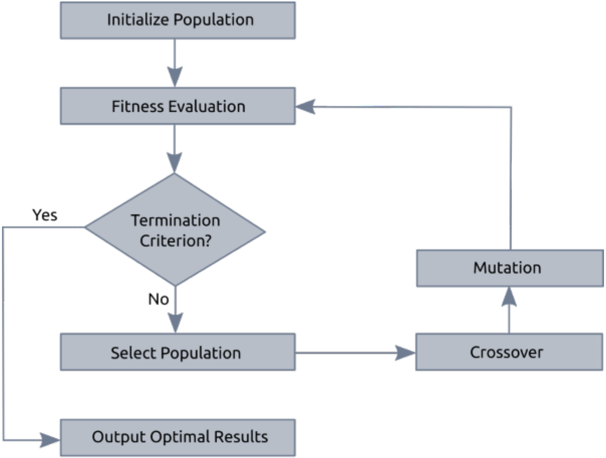


Figure 2: Basic structure of Genetic Algorithm

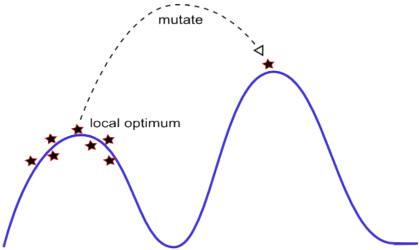
2 Outline

KU LEUVEN

2 Problem statement

This thesis:

Is it advantageous to learn MLNs and their SDDs using genetic algorithms?



3 Outline

KU LEUVEN

3 Outline

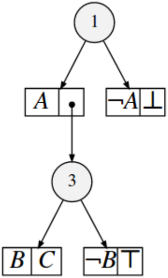
3. GeSDD
   - Search space
   - feature space
   - feature evaluation
   - Weight Learning
   - Structure Learning

KU LEUVEN

3 GeSDD: search space

3 GeSDD: search space

Earlier:

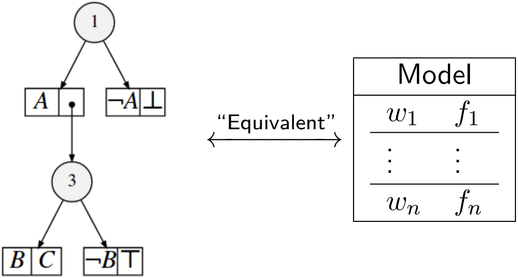


“Equivalent” $\longleftrightarrow$

Model	
w_1	f_1
$\vdots$	$\vdots$
w_n	f_n

3 GeSDD: search space

Earlier:



Learn MLN and its SDD simultaneously:

$$I = \{(M, \alpha) | encoding(M) = \alpha, |M| < k_f\} \quad (5)$$

3 Outline

3. GeSDD
   - Search space
   - feature space
   - feature evaluation
   - Weight Learning
   - Structure Learning

3 GeSDD: feature space

$$
\text{Model} = \begin{array}{|cc|}
\hline
\multicolumn{2}{|c|}{\text{Model}} \\
\hline
w_1 & f_1 \\
\hline
\vdots & \vdots \\
\hline
w_n & f_n \\
\hline
\end{array}
$$

3 GeSDD: feature space

$$\text{Model} = \begin{array}{|c|}\hline \text{Model} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}$$

Question: What is the form of f_i?

- $f = l_1 \lor ... \lor l_m$
- $f = l_1 \land ... \land l_m$
- $f = f'_1 \circ ... \circ f'_m$

KU LEUVEN

3 GeSDD: feature space

$$\text{Model} = \begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}$$

Question: What is the form of f_i?

- $f = l_1 \vee ... \vee l_m$
- $f = l_1 \wedge ... \wedge l_m$
- $f = f'_1 \circ ... \circ f'_m$

In words: conjunctions of: literals or negations of conjuncted literals

3 Outline

3. GeSDD
   - Search space
   - feature space
   - feature evaluation
   - Weight Learning
   - Structure Learning

KU LEUVEN

3 GeSDD: feature evaluation

Given a feature $f = f_1 \wedge \cdots \wedge f_l$ and data $\mathcal{X}$:

3 GeSDD: feature evaluation

Given a feature $f = f_1 \wedge \cdots \wedge f_l$ and data $\mathcal{X}$:

- How "Good" or "Informative" is feature f?

3 GeSDD: feature evaluation

Given a feature $f = f_1 \wedge \cdots \wedge f_l$ and data $\mathcal{X}$:

- How "Good" or "Informative" is feature f?
- Several measures exist

3 GeSDD: feature evaluation

Given a feature $f = f_1 \wedge \cdots \wedge f_l$ and data $\mathcal{X}$:

- How "Good" or "Informative" is feature f?
- Several measures exist

GeSDD uses average pairwise mutual information:

$$AMI(f, \mathcal{X}) = \frac{2}{l(l-1)} \sum_{i=1}^{l} \sum_{j=i+1}^{l} I(f_i; f_j) \tag{6}$$

3 Outline

3. GeSDD
    - Search space
    - feature space
    - feature evaluation
    - Weight Learning
    - Structure Learning

3 GeSDD: Weight Learning

3 GeSDD: Weight Learning

Learning weights: maximum liklihood

$$
\begin{aligned}
&\underset{W}{\arg\max} \sum_{i=1}^{n} \sum_{j=1}^{k} w_j f_j(x_i) - n \log(Z) \\
&\underset{W}{\arg\max} \sum_{j=1}^{k} w_j \text{count}(f_j) - n \log(Z)
\end{aligned}
\tag{7}
$$

3 GeSDD: Weight Learning

Learning weights: maximum liklihood

$$\begin{aligned} &\operatorname*{arg\,max}_W \sum_{i=1}^{n} \sum_{j=1}^{k} w_j f_j(x_i) - n\log(Z) \\ &\operatorname*{arg\,max}_W \sum_{j=1}^{k} w_j \text{count}(f_j) - n\log(Z) \end{aligned} \tag{7}$$

No closed form: gradient methods!

$$\frac{\partial f(W)}{\partial w_j} = \text{count}(f_j) - n\frac{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x})w_j) f_j(\bar{x})}{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x})w_j)} \tag{8}$$

3 GeSDD: Weight Learning

Learning weights: maximum liklihood

$$\begin{aligned} &\text{arg}\max_W \sum_{i=1}^{n} \sum_{j=1}^{k} w_j f_j(x_i) - n \log(Z) \\ &\text{arg}\max_W \sum_{j=1}^{k} w_j \text{count}(f_j) - n \log(Z) \end{aligned} \tag{7}$$

No closed form: gradient methods!

$$\frac{\partial f(W)}{\partial w_j} = \text{count}(f_j) - n \frac{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x}) w_j) f_j(\bar{x})}{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x}) w_j)} \tag{8}$$

3 GeSDD: Weight Learning

Learning weights: maximum liklihood

$$\begin{aligned}
&\arg\max_W \sum_{i=1}^{n}\sum_{j=1}^{k} w_j f_j(x_i) - n\log(Z) \\
&\arg\max_W \sum_{j=1}^{k} w_j \text{count}(f_j) - n\log(Z)
\end{aligned} \tag{7}$$

No closed form: gradient methods!

$$\frac{\partial f(W)}{\partial w_j} = \text{count}(f_j) - n\frac{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x})w_j) f_j(\bar{x})}{\sum_{\bar{x}} \exp(\sum_{j=1}^{k} f_j(\bar{x})w_j)} \tag{8}$$

3 Outline

KU LEUVEN

3 GeSDD: Structure Learning

KU LEUVEN

3 GeSDD: Structure Learning

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: Fitness

What should be considered:

3 GeSDD: Fitness

What should be considered:

- Good fit

3 GeSDD: Fitness

What should be considered:

- Good fit
- Tractability

3 GeSDD: Fitness

What should be considered:

- Good fit
- Tractability

$$f((M,\alpha)) = \max\left(LL(M) - LL(M_E) - \beta|\alpha|, 0\right) \tag{9}$$

Interpretation:

- $|.|$: SDD size
- $LL(M) - LL(M_E)$: gain in LL compared to empty model
- β determines tractability

3 GeSDD: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: genetic algorithm

1 Fitness

2 Selection

3 Mutations

4 Cross-over

5 Other heuristics

3 GeSDD: selection

Which models cross over?

3 GeSDD: selection

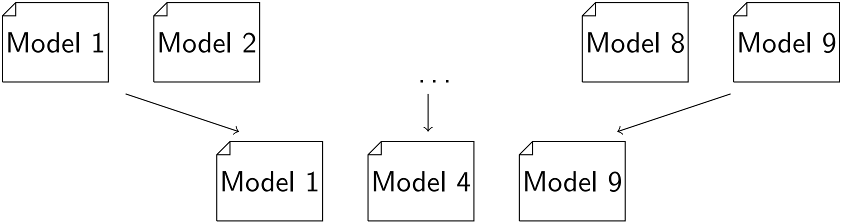
Which models cross over?

- Very simple: Tournament select

3 GeSDD: selection

Which models cross over?

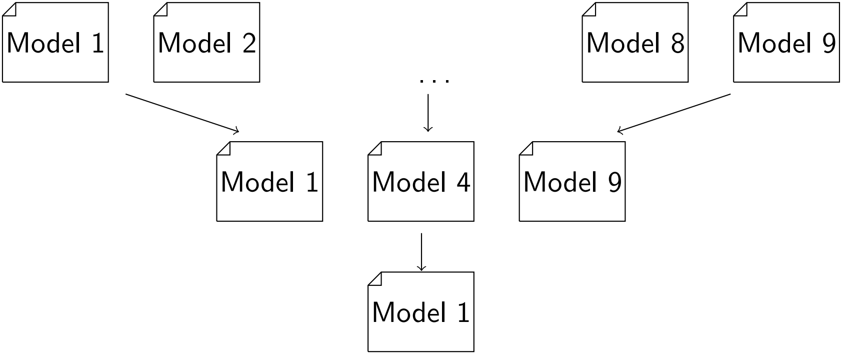
- Very simple: Tournament select



3 GeSDD: selection

Which models cross over?

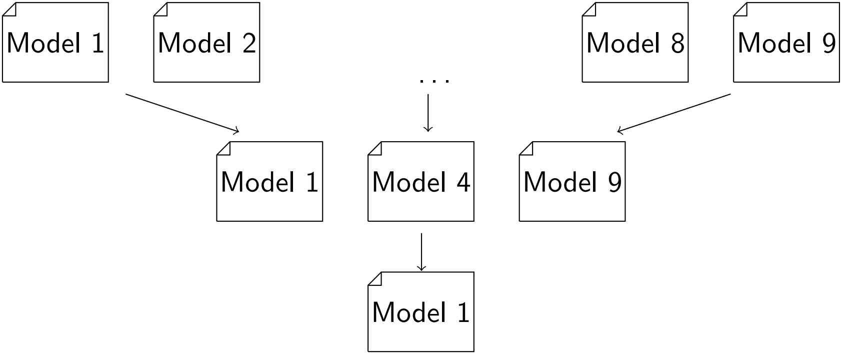
- Very simple: Tournament select



3 GeSDD: selection

Which models cross over?

- Very simple: Tournament select



$$P(\text{select } M_i | f) \sim f_i \tag{10}$$

KU LEUVEN

3 GeSDD: genetic algorithm

1 Fitness
2 Selection
3 Mutations
4 Cross-over
5 Other heuristics

3 GeSDD: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: mutation

Mutation 1: Adding features

3 GeSDD: mutation

Mutation 1: Adding features

$$M_a\left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}, \alpha\right)$$

3 GeSDD: mutation

Mutation 1: Adding features

$$M_a\left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}, \alpha\right) = \left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model'}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline w_{n+1} & f_{n+1} \\ \hline w_{n+2} & f_{n+2} \\ \hline \end{array}, \alpha'\right)$$

3 GeSDD: mutation

Mutation 1: Adding features

$$M_a\left(\begin{array}{|c|}\hline \text{Model}\\ \hline \begin{array}{cc} w_1 & f_1\\ \hline \vdots & \vdots \\ \hline w_n & f_n\end{array}\\ \hline\end{array}, \alpha\right) = \left(\begin{array}{|c|}\hline \text{Model'}\\ \hline \begin{array}{cc} w_1 & f_1\\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline w_{n+1} & f_{n+1} \\ \hline w_{n+2} & f_{n+2}\end{array}\\ \hline\end{array}, \alpha'\right)$$

Technical:

- $\alpha' = \alpha \wedge \beta$ with β is the encoding of new rules
- No re-compilation!

3 GeSDD: mutation

1. **How many features to add?**

3 GeSDD: mutation

1. How many features to add?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{11}$$

3 GeSDD: mutation

1. **How many features to add?**

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{11}$$

2. **How to generate a single feature?**

3 GeSDD: mutation

1. How many features to add?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{11}$$

2. How to generate a single feature?

1. Generate C_n candicate set $C = (c_1, \ldots, c_{C_n})$
2. Evaluate each feature c_i using AMI: $AMI(c_i)$
3. Sample a feature from the candidate set with $P_i \sim AMI(c_i)$

3 GeSDD: mutation

1. How many features to add?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{11}$$

2. How to generate a single feature?

1. Generate C_n candicate set $C = (c_1, \ldots, c_{C_n})$
2. Evaluate each feature c_i using AMI: $AMI(c_i)$
3. Sample a feature from the candidate set with $P_i \sim AMI(c_i)$

3. How to generate candidates?

- Approach based on J. Van Haaren et al.

3 GeSDD: mutation

Mutation 2: Removing random features

3 GeSDD: mutation

Mutation 2: Removing random features

$$M_r\left(\begin{array}{|cc|} \hline \multicolumn{2}{|c|}{\text{Model'}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline w_{n+1} & f_{n+1} \\ \hline w_{n+2} & f_{n+2} \\ \hline \end{array}, \alpha \right)$$

3 GeSDD: mutation

Mutation 2: Removing random features

$$M_r\left(\begin{array}{|cc|} \hline \multicolumn{2}{|c|}{\text{Model'}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline w_{n+1} & f_{n+1} \\ \hline w_{n+2} & f_{n+2} \\ \hline \end{array}, \alpha \right) = \left(\begin{array}{|cc|} \hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}, \alpha' \right)$$

3 GeSDD: mutation

Mutation 2: Removing random features

$$M_r\left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model'}} \\ \hline w_1 & f_1 \\ \vdots & \vdots \\ w_n & f_n \\ w_{n+1} & f_{n+1} \\ w_{n+2} & f_{n+2} \\ \hline \end{array}, \alpha\right) = \left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \vdots & \vdots \\ w_n & f_n \\ \hline \end{array}, \alpha'\right)$$

Technical:

- $\alpha' = \alpha | M'$ with $M' \in M$ a subset
- No re-compilation!

KU LEUVEN

3 GeSDD: mutation

1. How many features to remove?

3 GeSDD: mutation

1. How many features to remove?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{12}$$

3 GeSDD: mutation

1. **How many features to remove?**

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{12}$$

2. **How to select features to remove?**

KU LEUVEN

3 GeSDD: mutation

1. How many features to remove?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{12}$$

2. How to select features to remove?

$$P_i \sim \exp(w_i) \tag{13}$$

3 GeSDD: mutation

Mutation 3: feature altering

3 GeSDD: mutation

Mutation 3: feature altering

$$M_{alt}\left(\begin{array}{|c|} \hline \text{Model} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array} , \alpha \right)$$

3 GeSDD: mutation

Mutation 3: feature altering

$$
M_{alt}\left(\begin{array}{|c|}\hline \text{Model} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha\right) = \left(\begin{array}{|c|}\hline \text{Model'} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i' \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha'\right)
$$

3 GeSDD: mutation

Mutation 3: feature altering

$$M_{alt}\left(\begin{array}{|c|}\hline \text{Model} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha\right) = \left(\begin{array}{|c|}\hline \text{Model'} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i' \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha'\right)$$

Technical:

- No re-compilation!

3 GeSDD: mutation

Mutation 3: feature altering

$$M_{alt}\left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}, \alpha\right) = \left(\begin{array}{|cc|}\hline \multicolumn{2}{|c|}{\text{Model'}} \\ \hline w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i' \\ \hline \vdots & \vdots \\ \hline w_n & f_n \\ \hline \end{array}, \alpha'\right)$$

Technical:

- No re-compilation!
- $\alpha' = \alpha | M' \wedge \beta$

3 GeSDD: mutation

Mutation 3: feature altering

$$M_{alt}\left(\begin{array}{|c|}\hline \text{Model}\\\hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array}\\\hline\end{array}, \alpha\right) = \left(\begin{array}{|c|}\hline \text{Model'}\\\hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i' \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array}\\\hline\end{array}, \alpha'\right)$$

Technical:

- No re-compilation!
- $\alpha' = \alpha|M' \wedge \beta$

3 GeSDD: mutation

Mutation 3: feature altering

$$M_{alt}\left(\begin{array}{|c|} \hline \text{Model} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline w_i & f_i \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha \right) = \left(\begin{array}{|c|} \hline \text{Model'} \\ \hline \begin{array}{cc} w_1 & f_1 \\ \hline \vdots & \vdots \\ \hline \boxed{w_i \quad f_i'} \\ \hline \vdots & \vdots \\ \hline w_n & f_n \end{array} \\ \hline \end{array}, \alpha' \right)$$

Technical:

- No re-compilation!
- $\alpha' = \alpha | M' \wedge \beta$

3 GeSDD: mutation

1. How many features to change?

3 GeSDD: mutation

1. How many features to change?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{14}$$

3 GeSDD: mutation

1. **How many features to change?**

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{14}$$

2. **How to transform a single feature?**

3 GeSDD: mutation

1. How many features to change?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{14}$$

2. How to transform a single feature?

1. Generate C_n candicate set $C = (c_1, \dots, c_{C_n})$
2. Evaluate each feature c_i using AMI: $AMI(c_i)$
3. Sample a feature from the candidate set with $P_i \sim AMI(c_i)$

3 GeSDD: mutation

1. How many features to change?

$$l \sim \mathcal{U}(1, \lceil \gamma * |M| \rceil) \tag{14}$$

2. How to transform a single feature?

1. Generate C_n candicate set $C = (c_1, \ldots, c_{C_n})$
2. Evaluate each feature c_i using AMI: $AMI(c_i)$
3. Sample a feature from the candidate set with $P_i \sim AMI(c_i)$

3. How to transform features into candidates?

- GeSDD employs 4 strategies.

3 GeSDD: mutation

Mutation 3: feature altering

3 GeSDD: mutation

Mutation 3: feature altering

Method	f	$t(f)$

KU LEUVEN

3 GeSDD: mutation

Mutation 3: feature altering

Method	f		$t(f)$
feature expansion	$l_1 \wedge \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_k \wedge l_{k+1}$

3 GeSDD: mutation

Mutation 3: feature altering

Method	f		$t(f)$
feature expansion	$l_1 \wedge \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_k \wedge l_{k+1}$
feature schrinking	$l_1 \wedge \ldots \wedge l_{k-1} \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_{k-1}$

3 GeSDD: mutation

Mutation 3: feature altering

Method	f		$t(f)$
feature expansion	$l_1 \wedge \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_k \wedge l_{k+1}$
feature schrinking	$l_1 \wedge \ldots \wedge l_{k-1} \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_{k-1}$
individual negation	$l_1 \wedge \ldots \wedge l_i \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge \neg l_i \ldots \wedge l_k$

3 GeSDD: mutation

Mutation 3: feature altering

Method	f		$t(f)$
feature expansion	$l_1 \wedge \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_k \wedge l_{k+1}$
feature schrinking	$l_1 \wedge \ldots \wedge l_{k-1} \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge l_{k-1}$
individual negation	$l_1 \wedge \ldots \wedge l_i \ldots \wedge l_k$	$\rightarrow$	$l_1 \wedge \ldots \wedge \neg l_i \ldots \wedge l_k$
group negation	$l_1 \wedge l_2 \wedge l_3 \ldots \wedge l_k$	$\rightarrow$	$\neg(l_1 \wedge l_2 \wedge l_3) \ldots \wedge l_k$

3 GeSDD: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: cross-over

Main Idea: sharing “good” features!

3 GeSDD: cross-over

Main Idea: sharing “good” features!

Model 1	
4	f_1
2	f_2
0.3	f_3

Model 2	
3.2	f_4
1	f_5
2.2	f_6

3 GeSDD: cross-over

Main Idea: sharing “good” features!

Model 1	
4	f_1
2	f_2
0.3	f_3

Model 2	
3.2	f_4
1	f_5
2.2	f_6

Model 3	
4	f_1
2	f_2
0.3	f_3
3.2	f_4

Model 4	
3.2	f_4
1	f_5
2.2	f_6
4	f_1

3 GeSDD: cross-over

1. **How many features to share?**

3 GeSDD: cross-over

1. How many features to share?

$$l_1 \sim \mathcal{U}(1, \lceil \gamma * |M_1| \rceil)$$
$$l_2 \sim \mathcal{U}(1, \lceil \gamma * |M_2| \rceil) \tag{15}$$

3 GeSDD: cross-over

1. How many features to share?

$$
\begin{aligned}
l_1 &\sim \mathcal{U}(1, \lceil \gamma * |M_1| \rceil) \\
l_2 &\sim \mathcal{U}(1, \lceil \gamma * |M_2| \rceil)
\end{aligned}
\tag{15}
$$

2. Which features to swap?

3 GeSDD: cross-over

1. **How many features to share?**

$$l_1 \sim \mathcal{U}(1, \lceil \gamma * |M_1| \rceil)$$
$$l_2 \sim \mathcal{U}(1, \lceil \gamma * |M_2| \rceil) \quad (15)$$

2. **Which features to swap?**

$$P(f_i|W) \sim \exp(w_i) \quad (16)$$

3 GeSDD: cross-over

1. How many features to share?

$$l_1 \sim \mathcal{U}(1, \lceil \gamma * |M_1| \rceil)$$
$$l_2 \sim \mathcal{U}(1, \lceil \gamma * |M_2| \rceil) \tag{15}$$

2. Which features to swap?

$$P(f_i|W) \sim \exp(w_i) \tag{16}$$

Model 1	
4	f_1
2	f_2
0.3	f_3

$\rightarrow$

f_i	$P(f_i\|W)$
f_1	0.86
f_2	0.12
f_3	0.02

3 Structure learning: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 Structure learning: genetic algorithm

1. Fitness
2. Selection
3. Mutations
4. Cross-over
5. Other heuristics

3 GeSDD: MLN trimming

Consider:

3 GeSDD: MLN trimming

Consider:

Model	
5	f_1
0.05	f_2

3 GeSDD: MLN trimming

Consider:

Model	
5	f_1
0.05	f_2

$$P(x) \sim \exp(5f_1(x) + 0.05f_2(x))$$

3 GeSDD: MLN trimming

Consider:

Model	
5	f_1
0.05	f_2

$$P(x) \sim \exp(5 f_1(x) + 0.05 f_2(x))$$

$$LL\left(\begin{array}{|ll|} \hline \multicolumn{2}{|c|}{\text{Model 1}} \\ \hline 5 & f_1 \\ 0.05 & f_2 \\ \hline \end{array} \right) \approx LL\left(\begin{array}{|ll|} \hline \multicolumn{2}{|c|}{\text{Model 2}} \\ \hline 5 & f_1 \\ \hline \end{array} \right)$$

3 GeSDD: MLN trimming

Consider:

Model	
5	f_1
0.05	f_2

$$P(x) \sim \exp(5f_1(x) + 0.05f_2(x))$$

$$LL\left(\begin{array}{|ll|}\hline \multicolumn{2}{|c|}{\text{Model 1}} \\ \hline 5 & f_1 \\ 0.05 & f_2 \\ \hline \end{array}\right) \approx LL\left(\begin{array}{|ll|}\hline \multicolumn{2}{|c|}{\text{Model 2}} \\ \hline 5 & f_1 \\ \hline \end{array}\right)$$

$$Trim((M, \alpha)) = \Big(M \setminus M', \alpha|M'\Big) \tag{17}$$
$$\text{where } M' = \{(f, w) \in M \,|\, |w| < \tau\}$$

KU LEUVEN

3 GeSDD: SDD minimization

Minimization:

3 GeSDD: SDD minimization

Minimization:

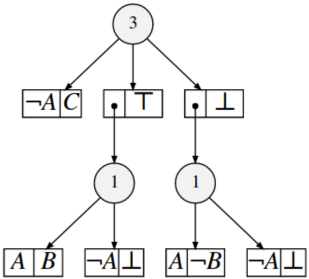
$$f(A, B, C) = (A \wedge B) \vee (\neg A \wedge C) \tag{18}$$

3 GeSDD: SDD minimization

Minimization:

$$f(A, B, C) = (A \wedge B) \vee (\neg A \wedge C) \tag{18}$$

$SDD(f) =$

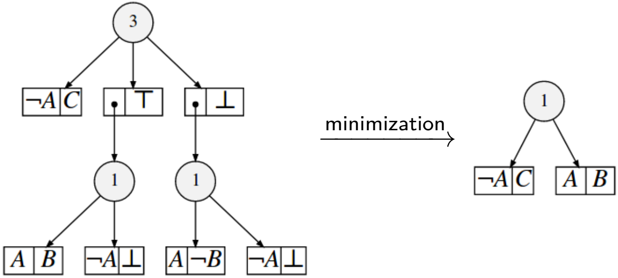


3 GeSDD: SDD minimization

Minimization:

$$f(A, B, C) = (A \wedge B) \vee (\neg A \wedge C) \tag{18}$$

$SDD(f) =$



4 Outline

KU LEUVEN

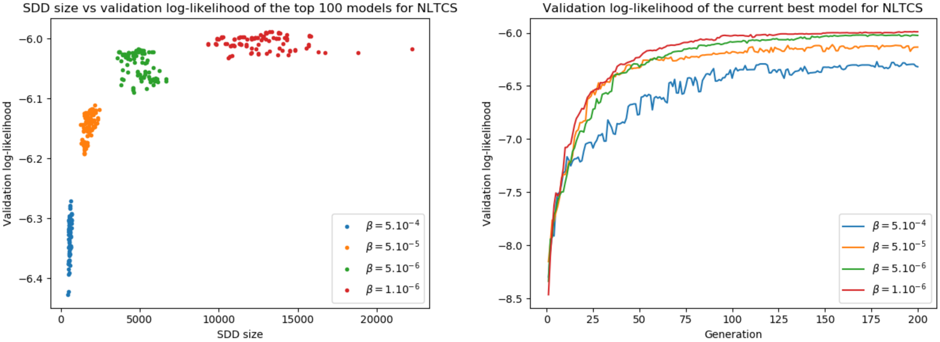
4 Results: brief overview

Overview of:

- Effects of β in fitness function?
- Effects of τ heuristic?
- Effects of minimization heuristic?
- Compare with LearnSDD

4 Effects of β

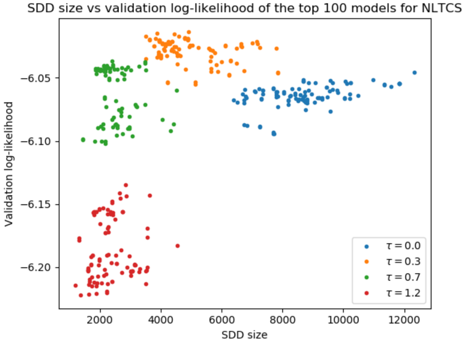
$$f((M,\alpha)) = \max(LL(M) - LL(M_E) - \beta|\alpha|, 0) \qquad (19)$$

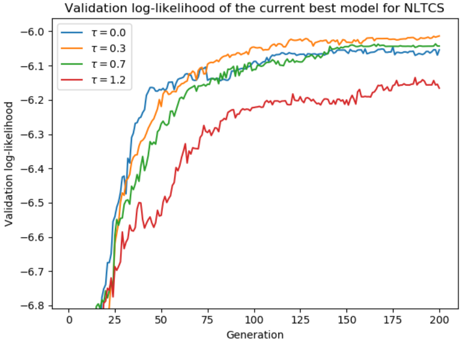


KU LEUVEN

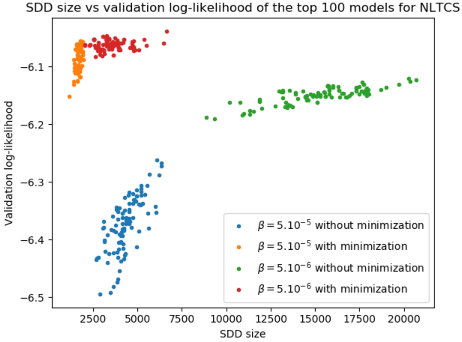
4 Effects of τ

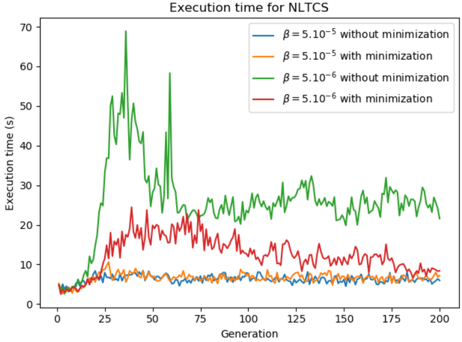
$$Trim((M,\alpha)) = \Big(M \setminus M', \alpha|M'\Big) \quad \text{where } M' = \{(f,w) \in M \,|\, |w| < \tau\} \tag{20}$$



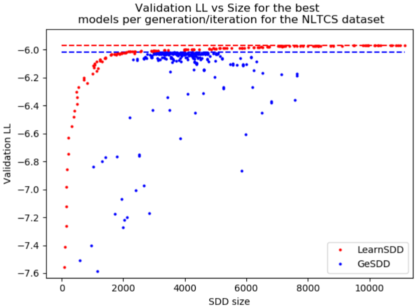


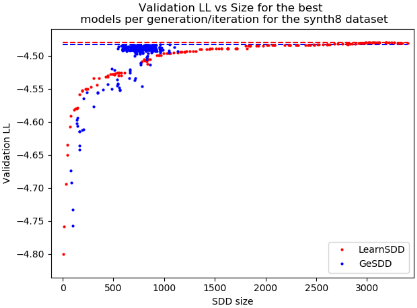
4 Effects of minimization





4 Comparison LearnSDD





KU LEUVEN

5 Outline

KU LEUVEN

5 Future work

Conclusion: Perhaps a more "directed" random search is needed.

Drawbacks:

- CPU/memory intensive
- Only Boolean variables (efficiently)
- Many parameter settings to explore

Extensions:

- Bootstrapping with other learners
- Combine greedy search with random pertubations
- Adaptive fitness function

Thank you for your attention! Questions?

Jessa Bekker, Jesse Davis, Arthur Choi, Adnan Darwiche, and Guy Van den Broeck.
Tractable learning for complex probability queries.
In *Advances in Neural Information Processing Systems 28 (NIPS)*, December 2015.

Mark Chavira and Adnan Darwiche.
On probabilistic inference by weighted model counting.
Artificial Intelligence, 172(6):772 – 799, 2008.

Gregory F. Cooper.
The computational complexity of probabilistic inference using bayesian belief networks.
Artificial Intelligence, 42(2):393 – 405, 1990.

Adnan Darwiche.
Sdd: A new canonical representation of propositional knowledge bases.
In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*,

Adnan Darwiche and Pierre Marquis.
A knowledge compilation map.
CoRR, abs/1106.1819, 2011.

KU LEUVEN