Curriculum Learning of Bayesian Network Structures

Yanpeng Zhao\textsuperscript{1}, Yetian Chen\textsuperscript{2}, Kewei Tu\textsuperscript{3}, Jin Tian\textsuperscript{4}

\{zhaoyp\textsuperscript{1}, tukw\textsuperscript{3}\}@shanghaitech.edu.cn \{yetian\textsuperscript{2}, jti\textsuperscript{4}\}@iastate.edu

ACML, Nov 22th, 2015, HongKong
Bayesian Network

- A directed acyclic graph DAG where
  - nodes: random variables
  - directed edges: probability dependencies among variables

- An example

\[
\begin{array}{c|c|c}
E & F & 0.9 \\
T & 0.1 \\
\end{array}
\]

\[
\begin{array}{c|c|c|c|c}
P(A|B,E) & F & 0.8 & 0.2 & 0.15 & 0.1 \\
T & 0.2 & 0.8 & 0.85 & 0.9 \\
\end{array}
\]

\[
\begin{array}{c|c|c}
P(N|A) & F & 0.2 & 0.9 \\
T & 0.8 & 0.1 \\
\end{array}
\]
BN Structure Learning

▶ Why
  ▶ effective inference, causal modeling

▶ How
  ▶ construct the topology of the network using a structure searcher

▶ score the constructed network using a scoring function

\[
score(G : D) = \log P(G|D) \propto \log P(D|G) + \log P(G)
\]
**Q**: How can we find the relations between all variables?

<table>
<thead>
<tr>
<th>Variables</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>S, B, D, L, E, X, A, T</td>
<td>(0 0 0 1 1 1 0 0)</td>
</tr>
<tr>
<td></td>
<td>(0 0 0 0 0 0 0 1 1)</td>
</tr>
<tr>
<td></td>
<td>(1 0 0 1 1 1 0 0)</td>
</tr>
<tr>
<td></td>
<td>(0 1 0 0 1 1 1 0)</td>
</tr>
<tr>
<td></td>
<td>(0 0 1 1 1 0 0 0)</td>
</tr>
<tr>
<td></td>
<td>(1 1 0 0 0 0 1 1)</td>
</tr>
<tr>
<td></td>
<td>(0 1 0 1 1 1 0 0)</td>
</tr>
<tr>
<td></td>
<td>(0 1 0 0 1 1 0 0)</td>
</tr>
<tr>
<td></td>
<td>(1 0 1 1 0 1 1 0)</td>
</tr>
<tr>
<td></td>
<td>(0 1 0 1 1 1 0 0)</td>
</tr>
</tbody>
</table>
**BN Structure Learning**

**Q:** How can we find the relations between all variables?

Variables $S, B, D, L, E, X, A, T$ correspond to each column of the dataset respectively.

- Variables: $S, B, D, L, E, X, A, T$
- Instances:
  - $(0, 0, 0, 1, 1, 1, 0, 0)$
  - $(0, 0, 0, 0, 0, 0, 1, 1)$
  - $(1, 0, 0, 1, 1, 1, 0, 0)$
  - $(0, 1, 0, 0, 1, 1, 1, 0)$
  - $(0, 0, 1, 1, 1, 0, 0, 0)$
  - $(1, 1, 0, 0, 0, 0, 1, 1)$
  - $(0, 1, 0, 1, 1, 1, 0, 0)$
  - $(0, 1, 0, 0, 1, 1, 0, 0)$
  - $(1, 0, 1, 1, 0, 1, 1, 0)$
  - $(0, 1, 0, 1, 1, 1, 0, 0)$

What's the structure?
Curriculum Learning

Ideas

Guided learning helps training humans and animals

Start from simpler examples / easier tasks   (Piaget 1952, Skinner 1958)


1Yoshua Bengio et al. ICML 2009
Curriculum Learning

- A curriculum is a sequence of weighting schemes of the training data $\langle W_1, W_2, \ldots, W_n \rangle$
  - $W_1$ assigns more weight to easier samples
  - each next scheme assigns more weight to harder samples
  - $W_n$ assigns uniform weight to all samples

- Advantages
  - faster convergence to a minimum
  - convergence to better local minimum

- Difficulties
  - how to define better curriculum strategies
Learn BN Structure via CL

Motivation

- Human learn in a more organized way, starting with more common samples that involve dependency relations between only a small subset of variables

At stage 1, learn a subnet $G_1$ over $\{S, B, D\}$ from scratch with the rest variables fixed at $(1 1 1 0 0)$
Learn BN Structure via CL

Motivation

- Only turn to less common data samples involving dependency relations with additional variables when some knowledge (i.e., a partial model) is obtained.

At stage 2, learn a larger subnet $G_2$ over \( \{S, B, D, L, E, X\} \) with $G_1$ as the start point of search while fix the rest variables at (0 0).
We define the curriculum as \((X_{(1)}, ..., X_{(n)})\), a sequence of selected subsets of the random variables \(X_{(i)}\), over which the corresponding subnet \(G_i\) is learnt.

Where \(X = (X_1, ..., X_n)\) is a variable set, \(X_{(i)} \subseteq X, X'_{(i)} = X \setminus X_{(i)}, X_{(i)} \subset X_{(i+1)}\). 

\((G_1, ..., G_m)\) is a sequence of intermediate learning targets.
Curriculum in BN Structure Learning

We define the curriculum as \( (X_1, \ldots, X_n) \), a sequence of selected subsets of the random variables \( X(i) \), over which the corresponding subnet \( G_i \) is learnt.

Where \( X = (X_1, \ldots, X_n) \) is a variable set, \( X(i) \subseteq X, X'(i) = X \setminus X(i), X(i) \subset X(i+1) \).

\((G_1, \ldots, G_m)\) is a sequence of intermediate learning targets.

Curriculum: \( \{X(1), X(2), X(3)\} \)

\( X(1) = \{S, B, D\}, X(2) = \{S, B, D, L, E, X\}, X(3) = \{S, B, D, L, E, X, A, T\} \)
Curriculum in BN Structure Learning

In terms of the sample weighting scheme $\langle W_1, W_2, \ldots, W_n \rangle$, at stage $i$, $W_i$ assigns '1' to those samples with $X'_{(i)} = x'_{(i)}$ (the fixed value) and '0' to the other samples.

At stage 1, learn a subnet $G_1$ over $X_{(i)} = \{S, B, D\}$ with $X'_{(i)} = \{L, E, X, A, T\}$ fixed at $(1 1 1 0 0)$. For the left dataset, $W_1 = (1 0 1 0 1 0 1 0 0 1)$

<table>
<thead>
<tr>
<th>Variables</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>(0 0 0 1 1 1 0 0)</td>
</tr>
<tr>
<td>B</td>
<td>(0 0 0 0 0 0 1 1)</td>
</tr>
<tr>
<td>C</td>
<td>(1 0 0 1 1 1 0 0)</td>
</tr>
<tr>
<td>D</td>
<td>(0 1 0 0 1 1 1 0)</td>
</tr>
<tr>
<td>E</td>
<td>(0 0 1 1 1 0 0 0)</td>
</tr>
<tr>
<td>F</td>
<td>(1 1 0 0 0 0 1 1)</td>
</tr>
<tr>
<td>G</td>
<td>(0 1 0 1 1 1 0 0)</td>
</tr>
<tr>
<td>H</td>
<td>(0 1 0 0 1 1 0 0)</td>
</tr>
<tr>
<td>I</td>
<td>(1 0 1 1 0 1 1 0)</td>
</tr>
<tr>
<td>J</td>
<td>(0 1 0 1 1 1 0 0)</td>
</tr>
</tbody>
</table>

\[ W_1 = (1 0 0 1 0 0 1 0 1 0) \]
Limitation

- We only used a small fraction of the dataset at each learning stage.

\[
\begin{align*}
(0 & 0 0 1 1 1 0 0) \\
(0 & 0 0 0 0 0 1 1) \\
(1 & 0 0 1 1 1 0 0) \\
(0 & 1 0 0 1 1 1 0) \\
(0 & 0 1 1 1 0 0 0) \\
(1 & 1 0 0 0 0 0 1) \\
(0 & 1 0 1 1 1 0 0) \\
(0 & 1 0 0 1 1 0 0) \\
(1 & 0 1 1 0 1 1 0) \\
(0 & 1 0 1 1 1 0 0)
\end{align*}
\]

At stage 1, learn a subnet $G_1$ over $X_{(i)} = \{S, B, D\}$. Here we only used the samples with $X'_{(i)} = \{L, E, X, A, T\}$ fixed at (1 1 1 0 0), the samples with a strikeout IS NOT used.
Limitation

- We only used a small fraction of the dataset at each learning stage

At stage 1, learn a subnet $G_1$ over $X(i) = \{S, B, D\}$. Here we only used the samples with $X'(i) = \{L, E, X, A, T\}$ fixed at (1 1 1 0 0), the samples with a strikeout is NOT used.

**Q:** Whether we can use all the samples at a learning stage?
Solution

An important observation

When we fix $X'_i$ to different values, our learning target is actually the same DAG structure $G_i$ but with different parameters (CPDs).

At stage 2, learn a subnet $G_1$ over $X_{(2)} = \{S, B, D, L, E, X\}$. $X'_i$ can take value from \{(0, 0), (1, 1), (1, 0)\}.
Solution

Let $X'_i$ take value from $\{x'_i,1, \ldots, x'_i,q\}$, the set of data segments $D_i = \{D_{i,1}, \ldots, D_{i,q}\}$ by grouping samples based on the values of $X'_i$.

▶ Assumption

▶ $D_{i,1}, \ldots, D_{i,q}$ are generated by the same $G_i$ but with independent CPDs
Let $X'_{(i)}$ take value from $\{x'_{(i),1}, ..., x'_{(i),q}\}$, the set of data segments $D_i = \{D_{i,1}, ..., D_{i,q}\}$ by grouping samples based on the values of $X'_{(i)}$.

▶ Assumption

▶ $D_{i,1}, ..., D_{i,q}$ are generated by the same $G_i$ but with independent CPDs

▶ Bayesian score function

$$\log P(G_i, D_i) = C + \sum_{j=1}^{q} \log P(G_i, D_{i,j})$$
Over-fitting

- Over-fitting occurs when sample size is small or there are many learning stages

- How to avoid

\[
penalty(G_i : D_i) = \left( \frac{a}{SS} + \frac{V(G_i)}{b} \right) E(G_i),
\]

$SS$: sample size; $V(G_i)$: number of the variables in $G_i$, $E(G_i)$: number of edges in $G_i$; $a, b$: positive constants.

- The final score function

\[
score(G_i : D_i) = \sum_{j=1}^{q} \log P(G_i, D_{i,j}) - penalty(G_i : D_i)
\]
How to Make a Curriculum

\[ S, B, D, L, E, X \]
\[ \mathcal{X}_{(2)} \]
\[ = \]
\[ \mathcal{S}, B, D \]
\[ \mathcal{X}_{(1)} \]
\[ + \]
\[ L, E, X \]
\[ \text{unfixed} \]
How to Make a Curriculum

$$S, B, D, L, E, X \quad \begin{cases} \quad \text{unfixed} \end{cases} \quad X_{(1)} = \quad S, B, D \quad + \quad L, E, X$$

Q: Which variables shall be unfixed (added into $X_{(i-1)}$)?
**How to Make a Curriculum**

### Question

\[
\begin{align*}
S, B, D, L, E, X_{(2)} & = S, B, D_{(1)} + L, E, X_{\text{unfixed}} \\
\end{align*}
\]

**Q:** Which variables shall be unfixed (added into \(X_{(i-1)}\))?

**Answer:** unfix the variables that are most likely to have connections with the current set of variables \(X_{(i-1)}\).
Theorems

**Theorem 1.** For any $i, j, k$ s.t. $1 \leq i < j < k \leq n$, we have

\[ d_H(G_i, G_k) \geq d_H(G_j, G_k) \]

where $d_H(G_i, G_j)$ is the structural Hamming distance (SHD) between the structures of two BNs $G_i$ and $G_j$.

**Theorem 2.** For any $i, j, k$ s.t. $1 \leq i < j < k \leq n$, we have

\[ d_{TV}(G_i, G_k) \geq d_{TV}(G_j, G_k) \]

where $d_{TV}(G_i, G_j)$ is the total variation distance between the two distributions defined by the two BNs $G_i$ and $G_j$. 
Datasets

<table>
<thead>
<tr>
<th>Network</th>
<th>Num. vars</th>
<th>Num. edges</th>
<th>Max in/out-degree</th>
<th>Cardinality range</th>
<th>Average cardinality</th>
</tr>
</thead>
<tbody>
<tr>
<td>alarm</td>
<td>37</td>
<td>46</td>
<td>4/5</td>
<td>2-4</td>
<td>2.84</td>
</tr>
<tr>
<td>andes</td>
<td>223</td>
<td>338</td>
<td>6/12</td>
<td>2-2</td>
<td>2.00</td>
</tr>
<tr>
<td>asia</td>
<td>8</td>
<td>8</td>
<td>2/2</td>
<td>2-2</td>
<td>2.00</td>
</tr>
<tr>
<td>child</td>
<td>20</td>
<td>25</td>
<td>2/7</td>
<td>2-6</td>
<td>3.00</td>
</tr>
<tr>
<td>hailfinder</td>
<td>56</td>
<td>66</td>
<td>4/16</td>
<td>2-11</td>
<td>3.98</td>
</tr>
<tr>
<td>hepar2</td>
<td>70</td>
<td>123</td>
<td>6/17</td>
<td>2-4</td>
<td>2.31</td>
</tr>
<tr>
<td>insurance</td>
<td>27</td>
<td>52</td>
<td>3/7</td>
<td>2-5</td>
<td>3.30</td>
</tr>
<tr>
<td>sachs</td>
<td>11</td>
<td>17</td>
<td>3/6</td>
<td>3-3</td>
<td>3.00</td>
</tr>
<tr>
<td>water</td>
<td>32</td>
<td>66</td>
<td>5/3</td>
<td>3-4</td>
<td>3.63</td>
</tr>
<tr>
<td>win95pts</td>
<td>76</td>
<td>112</td>
<td>7/10</td>
<td>2-2</td>
<td>2.00</td>
</tr>
</tbody>
</table>

Cardinality denotes the number of values that a variable can take.

\(^2\)http://www.bnlearn.com/bnrepository/
### Comparison With MMHC

#### Table: Comparisons under different metrics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Algorithm</th>
<th>Sample Size (SS)</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>100</td>
<td>500</td>
<td>1000</td>
<td>5000</td>
<td>10000</td>
<td>50000</td>
</tr>
<tr>
<td>BDeu</td>
<td>CL</td>
<td>1(0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MMHC</td>
<td>0.89(10)</td>
<td>1.06(0)</td>
<td>1.02(1)</td>
<td>1.01(2)</td>
<td>1.02(0)</td>
<td>1.01(2)</td>
</tr>
<tr>
<td>BIC</td>
<td>CL</td>
<td>1(0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MMHC</td>
<td>0.88(10)</td>
<td>1.07(1)</td>
<td>1.02(1)</td>
<td>1.02(4)</td>
<td>1.02(2)</td>
<td>1.01(2)</td>
</tr>
<tr>
<td>KL</td>
<td>CL</td>
<td>1(0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MMHC</td>
<td>1.71(10)</td>
<td>0.82(0)</td>
<td>0.96(1)</td>
<td>0.96(2)</td>
<td>0.97(0)</td>
<td>0.97(0)</td>
</tr>
<tr>
<td>SHD</td>
<td>CL</td>
<td>1(7)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MMHC</td>
<td>1.06(3)</td>
<td>1.26(1)</td>
<td>1.29(3)</td>
<td>1.07(2)</td>
<td>1.21(1)</td>
<td>1.24(3)</td>
</tr>
</tbody>
</table>

Results are collected under the metrics of BDeu, BIC, KL and SHD. The bold numbers represent better performance.

---

3 Ioannis Tsamardinos et al. ML 2006
Verification of Theorem 1

Changes of SHD from the target BN during curriculum learning with $SS = 5000$ on the alarm and hailfinder networks.
Conclusion

- We proposed a curriculum learning algorithm for BN structure learning
- We tailored the bayesian scoring function for our algorithm
- We proved two theorems that show theoretical properties of our algorithm
- We empirically showed that our algorithm outperformed the state-of-the-art MMHC algorithm in learning BN structures
Thanks
Q-1. How to estimate the likeliness of connections?

- By measuring the strength of the dependency (e.g., using mutual information) with the current set of variables

At stage $i$, compute pairwise mutual information $MI(X, Y)$ between any node $X$ in $X_{(i-1)}$ and node $Y$ in $X \setminus X_{(i-1)}$. Then for any node $Y$ in $X \setminus X_{(i-1)}$, compute the average pairwise mutual information by

$$\text{AveMI}(Y, X_{(i-1)}) = \frac{\sum_{X \in X_{(i-1)}} MI(X, Y) / |X_{(i-1)}|}{|X_{(i-1)}|}$$

As to variable $A$, compute $MI$ between $A$ and each variable in $\{S, B, D, L, E, X\}$, then average $MIs$. 

As shown in the diagram, the mutual information between variables can be calculated using their dependency relationships.
**Q-2.** How to configure the curriculum?

1. first pick variable $Y_1$ with the largest $\text{AveMI}(Y_1, X_{(i-1)})$

2. then pick the second variable $Y_2$ with the largest $\text{AveMI}(Y_2, X_{(i-1)} \cup \{Y_1\})$

3. so on and so forth

First compute every variable's $\text{AveMI}$ with all of the rest ones, then pick the variable $S$ with largest $\text{AveMI}$ and add it into a list $L$. The rest are selected in the sequential way as described above. $L$ changes as this

$$(S) \rightarrow (S, B) \rightarrow (S, B, D) \rightarrow \cdots$$
Q-3. How many variables would be added at a stage?

- That is, from stage $i - 1$ to $i$, which variables $X_{(i-1,i)}$ should we select to produce $X_{(i)} = X_{(i-1)} \cup X_{(i-1,i)}$?

- The number of variables selected, $|X_{(i-1,i)}|$, is called the step size.

Recall that we have got $L = (S, B, D, L, E, X, A, T)$, and every variable is most likely to have connections with the variables before it. Given step size 2, our curriculum could be

$\{\{S, B\}, \{S, B, D, L\}, \{S, B, D, L, E, X\}, \{S, B, D, L, E, X, A, T\}\}$

- Intuitively, the smaller step size, the more cautious and less time-efficient the algorithm is.
**Algorithm 1: Curriculum Learning of BN Structures**

**Input:** Variable Set $X$, Training Data $D$, Curriculum $(X_{(1)}, \ldots, X_{(m)})$. $G_0$ is initialized to a network containing variables in $X_{(1)}$ with no edge.

**for** $i = 1 \ldots m$ **do**

| Generate the set of data segments $D_i = \{D_{i,1}, \ldots, D_{i,q}\}$ based on the values of $X \setminus X_{(i)}$
| $G_i \leftarrow search(D_i, X_{(i)}, G_{i-1})$

**end**

**Return:** $G_m$. 