JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

Online Estimation of Discrete Densities

Michael Geilke[1], Andreas Karwath[1], Eibe Frank[2], and Stefan Kramer[1]

[1] Johannes Gutenberg-Universität Mainz, Germany
[2] University of Waikato, New Zealand

December 10, 2013

Motivation

Online store

Motivation

Online store

Books

Laptops

⋮

Motivation

Online store

Books $b_{137}, b_{91}, b_{140}, b_{32}, b_{4214}, b_{490}, \ldots$

Laptops $l_{70}, l_2, l_{42}, l_7, l_{28}, \ldots$

$\vdots$

Motivation

Online store

Books $b_{137}, b_{91}, b_{140}, b_{32}, b_{4214}, b_{490}, \ldots$

Laptops $l_{70}, l_{2}, l_{42}, l_{7}, l_{28}, \ldots$

$\vdots$

$l := \boxed{\begin{array}{lllll} \text{Brand} & \text{Display} & \text{CPU} & \text{RAM} & \text{OS} \quad \ldots \end{array}}$

Motivation

Online store

Books $b_{137}, b_{91}, b_{140}, b_{32}, b_{4214}, b_{490}, \ldots$

Laptops $l_{70}, l_{2}, l_{42}, l_{7}, l_{28}, \ldots$

$\vdots$

$l_{28} :=$

Brand	Display	CPU	RAM	OS	...
Lenovo	15.4 inch	Intel i7	8 GB	Linux	...

Problem Statement (1)

Given:

- nominal variables $X_1, X_2, \ldots, X_n$
- an unknown discrete joint density $f(X_1, X_2, \ldots, X_n)$
- an infinite stream of data that is distributed according to f

Problem Statement (1)

Given:

- nominal variables $X_1, X_2, \ldots, X_n$
- an unknown discrete joint density $f(X_1, X_2, \ldots, X_n)$
- an infinite stream of data that is distributed according to f

Goal: Determine a density estimate $\hat{f}$ for f in an online fashion, i.e.,

- the algorithm is only provided the current example,
- its current density estimate,
- and a limited amount of memory.

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_3, X_4, X_5)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_3, X_4, X_5)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_4 | X_3 = b, X_5 = c)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_4 | X_3 = b, X_5 = c)$$

$$f(X_1, X_2, X_3, X_4, X_5)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_4 | X_3 = b, X_5 = c)$$

$$f(X_1, X_2, X_3, X_4, X_5)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_4 | X_3 = b, X_5 = c)$$

$$f\left(X_1, X_2, X_4 \middle| \begin{array}{l} p(X_3 = a) = 0.3 \\ p(X_3 = b) = 0.7 \end{array}, \begin{array}{l} p(X_5 = c) = 0.4 \\ p(X_5 = d) = 0.2 \\ p(X_5 = e) = 0.4 \end{array}\right)$$

Problem Statement (2)

Goals: estimators should

- work online
- be consistent
- enable inference tasks:

$$f(X_1, X_2, X_4 | X_3 = b, X_5 = c)$$

$$f\left(X_1, X_2, X_4 \middle| \begin{array}{l} p(X_3 = a) = 0.3 \\ p(X_3 = b) = 0.7 \end{array}, \begin{array}{l} p(X_5 = c) = 0.4 \\ p(X_5 = d) = 0.2 \\ p(X_5 = e) = 0.4 \end{array}\right)$$

$$f(X_1, X_2, X_3, X_4, X_5) > \theta$$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1)$$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1) \cdot f_2(X_2 \mid X_1)$$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1) \cdot f_2(X_2 \mid X_1) \cdot \ldots$$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1) \cdot f_2(X_2 \mid X_1) \cdot \ldots \cdot f_n(X_n \mid X_1, X_2, \ldots, X_{n-1})$$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1) \cdot f_2(X_2 \mid X_1) \cdot \ldots \cdot f_n(X_n \mid X_1, X_2, \ldots, X_{n-1})$$

Classifiers

Majority class	for $f_1(X_1)$
Hoeffding trees	for $f_i(X_i \mid X_1, X_2, \ldots, X_{i-1})$, $i \in [2; n]$

Modeling Discrete Densities using Classifier Chains (1)

Applying the product rule to $f(X_1, X_2, \ldots, X_n)$ yields

$$f_1(X_1) \cdot f_2(X_2 \mid X_1) \cdot \ldots \cdot f_n(X_n \mid X_1, X_2, \ldots, X_{n-1})$$

Classifiers

Majority class	for $f_1(X_1)$
Hoeffding trees	for $f_i(X_i \mid X_1, X_2, \ldots, X_{i-1})$, $i \in [2; n]$

Both allow us to estimate the density in an online fashion.

Modeling Discrete Densities using Classifier Chains (2)

Let $\pi : \{1, 2, \ldots, n\} \to \{1, 2, \ldots, n\}$ be a bijective mapping.

Modeling Discrete Densities using Classifier Chains (2)

Let $\pi : \{1, 2, \ldots, n\} \rightarrow \{1, 2, \ldots, n\}$ be a bijective mapping.

$$f(X_1, X_2, \ldots, X_n) =$$

Modeling Discrete Densities using Classifier Chains (2)

Let $\pi : \{1, 2, \ldots, n\} \rightarrow \{1, 2, \ldots, n\}$ be a bijective mapping.

$$f(X_1, X_2, \ldots, X_n) =$$

$$f_1(X_{\pi(1)}) \cdot f_2(X_{\pi(2)}|X_{\pi(1)}) \cdot \ldots \cdot f_n(X_{\pi(n)} \mid X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(n-1)})$$

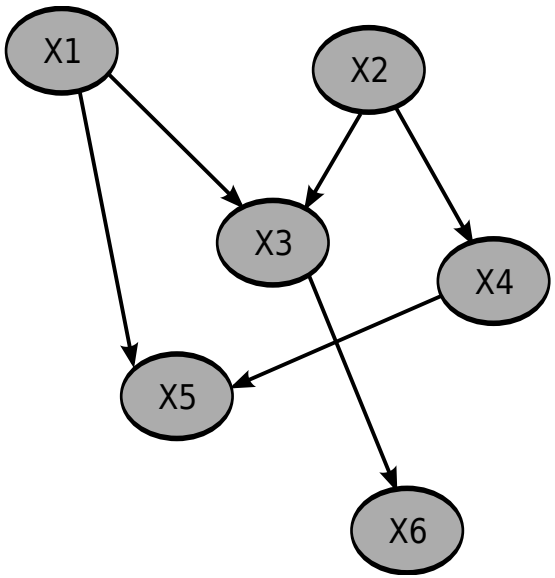
Modeling Discrete Densities using Classifier Chains (2)

Let $\pi : \{1, 2, \ldots, n\} \rightarrow \{1, 2, \ldots, n\}$ be a bijective mapping.

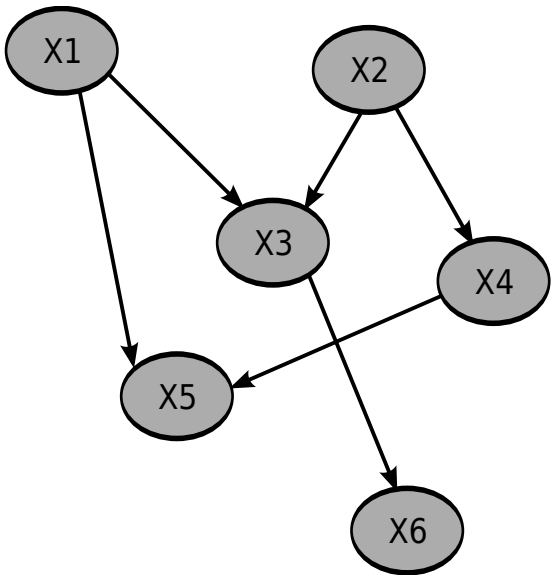
$$f(X_1, X_2, \ldots, X_n) =$$
$$f_1(X_{\pi(1)}) \cdot f_2(X_{\pi(2)}|X_{\pi(1)}) \cdot \ldots \cdot f_n(X_{\pi(n)} \mid X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(n-1)})$$

Although all such products represent the same joint density, the ordering may be important for the performance of our classifier chains.

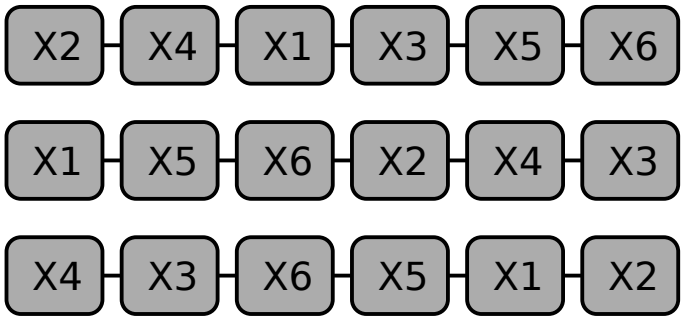
Ensembles of Classifier Chains (ECC)



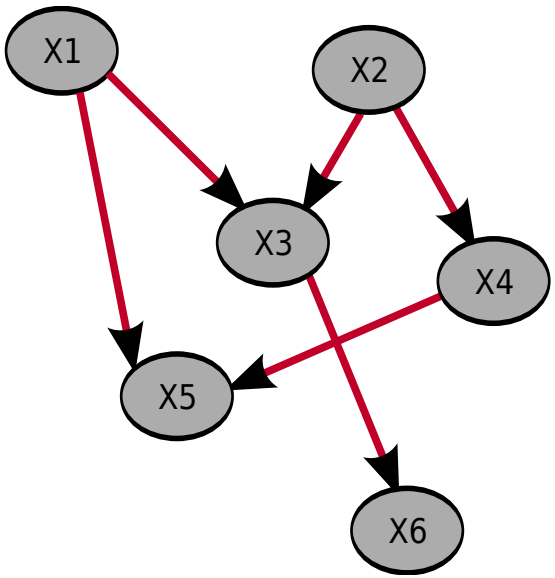
Ensembles of Classifier Chains (ECC)

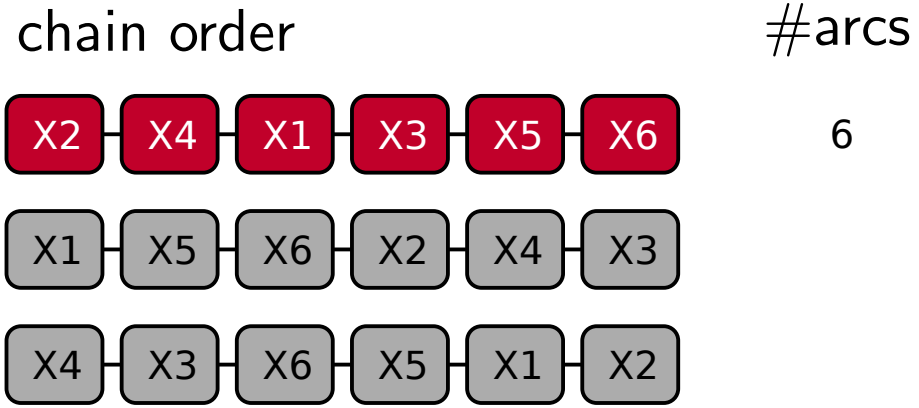


chain order

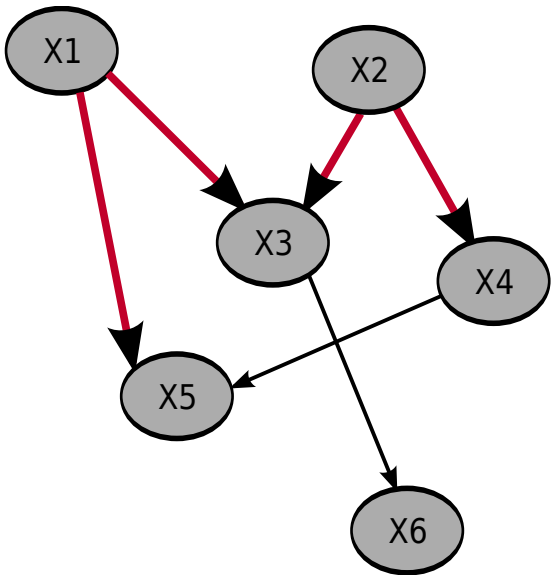


Ensembles of Classifier Chains (ECC)



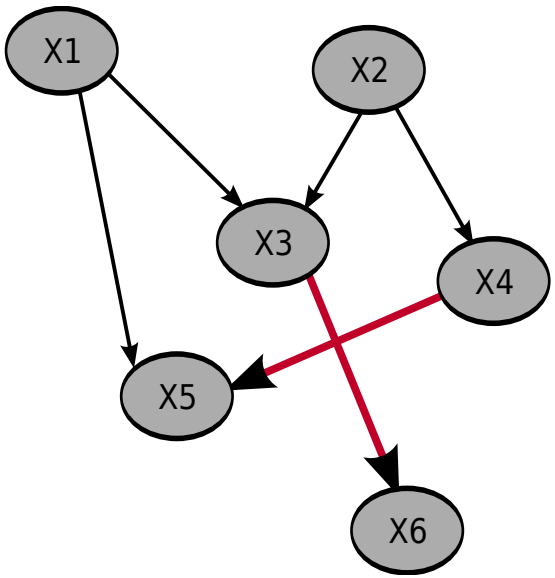


Ensembles of Classifier Chains (ECC)



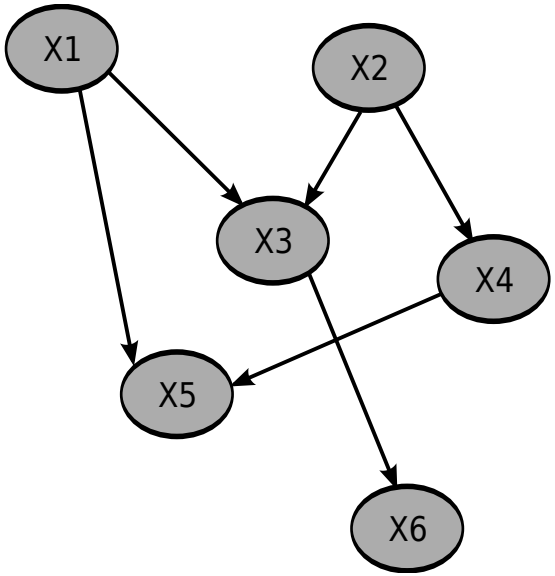
chain order	#arcs
X2 X4 X1 X3 X5 X6	6
X1 X5 X6 X2 X4 X3	4
X4 X3 X6 X5 X1 X2	

Ensembles of Classifier Chains (ECC)



chain order	#arcs
X2 X4 X1 X3 X5 X6	6
X1 X5 X6 X2 X4 X3	4
X4 X3 X6 X5 X1 X2	2

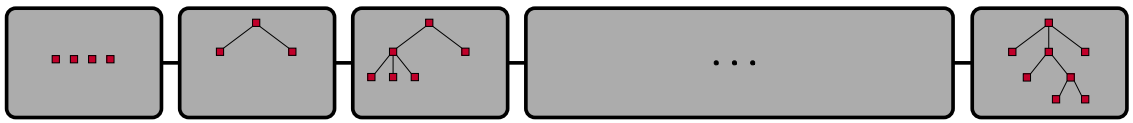
Ensembles of Classifier Chains (ECC)

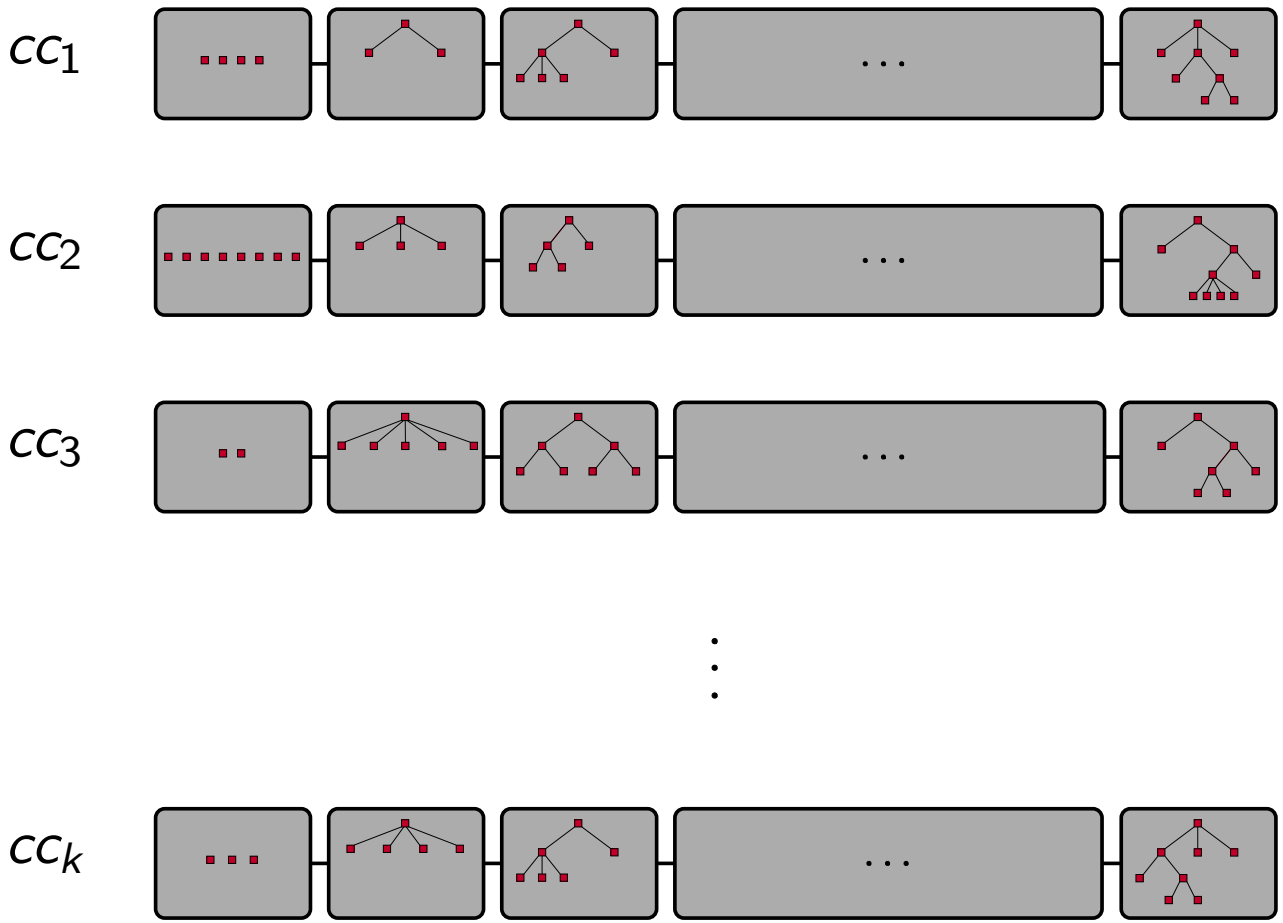


chain order	#arcs
X2 X4 X1 X3 X5 X6	6
X1 X5 X6 X2 X4 X3	4
X4 X3 X6 X5 X1 X2	2

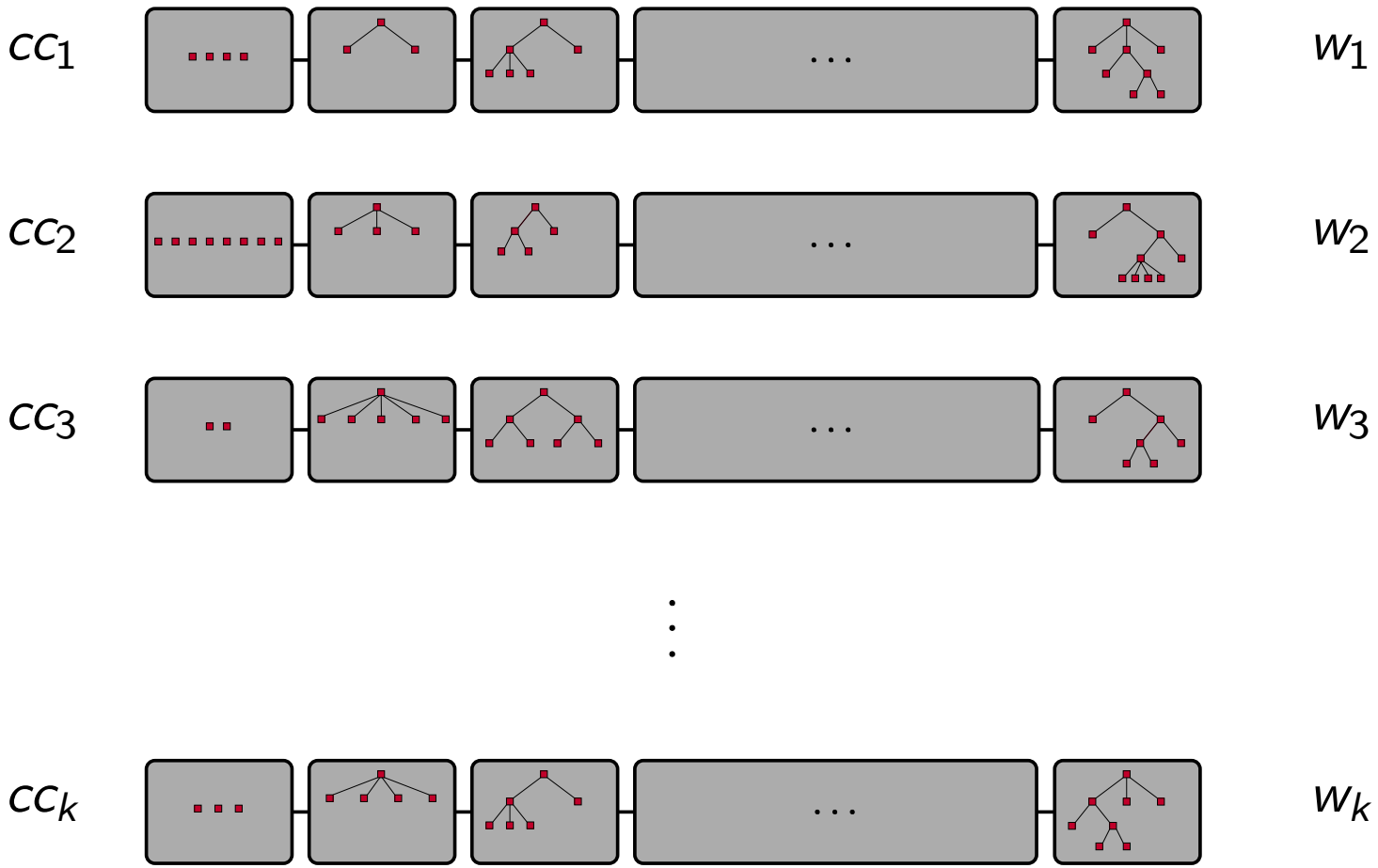
Hence, to increase robustness, we

- sample chains at random from the set of possible variable orderings,
- and average over the density estimates obtained.

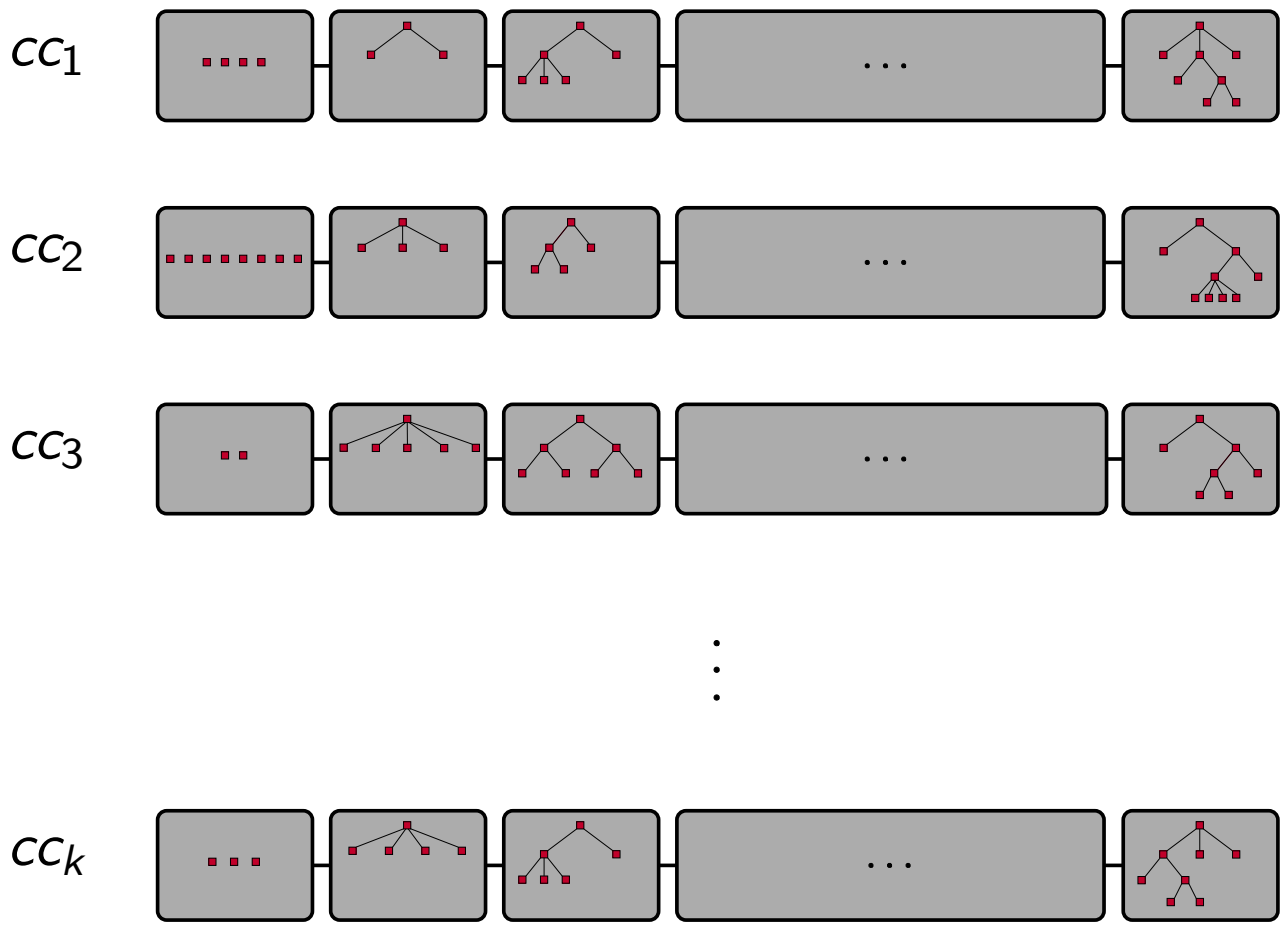
cc_1 


cc_1
cc_2
cc_3
cc_k

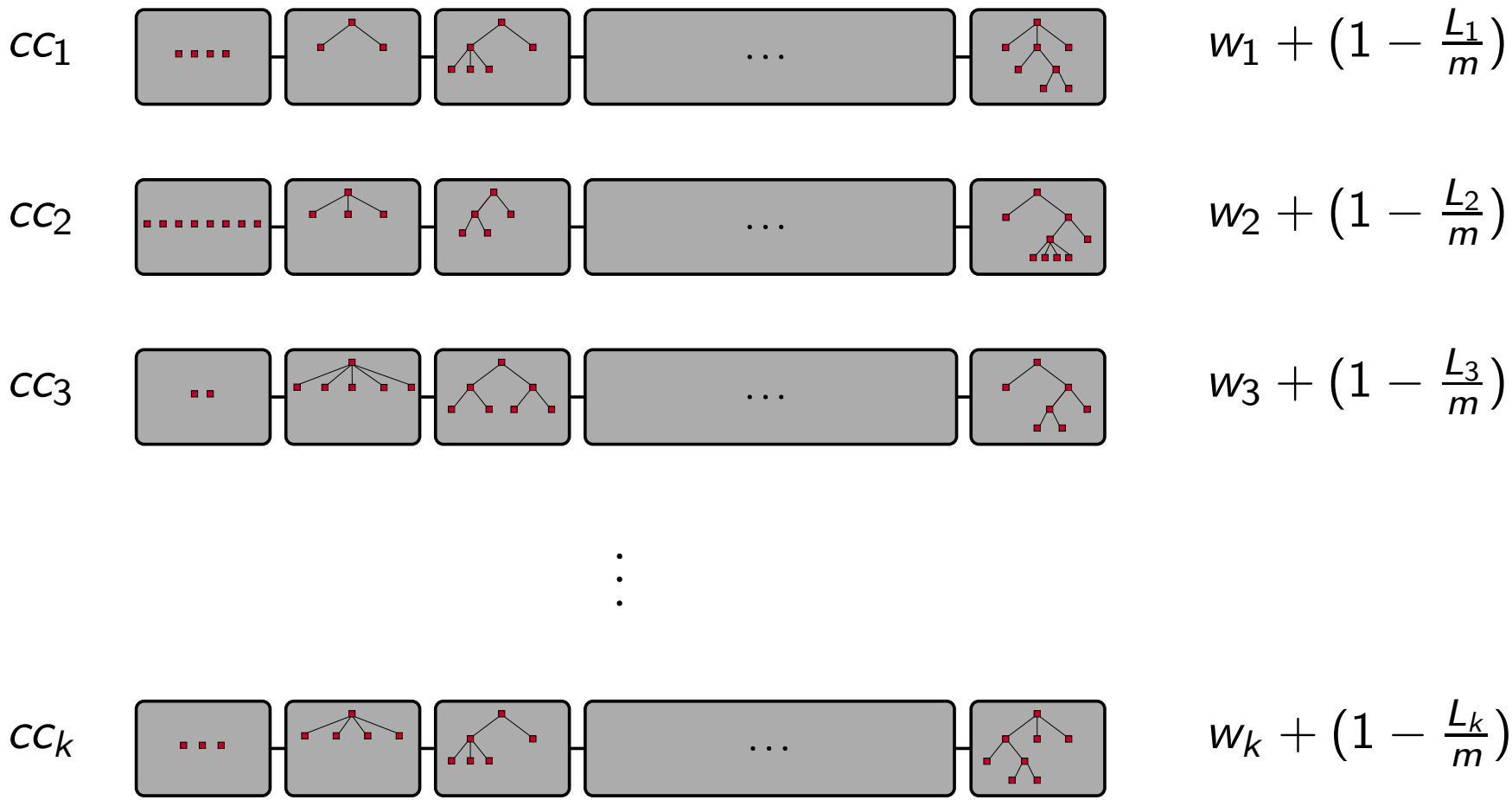
Ensembles of Weighted Classifier Chains (EWCC)



Ensembles of Weighted Classifier Chains (EWCC)



Ensembles of Weighted Classifier Chains (EWCC)



Evaluation (1)

Chain-based estimators:

- with a single classifier chain
- with an ensemble of (weighted) classifier chains

Evaluation (1)

Chain-based estimators:

- with a single classifier chain
- with an ensemble of (weighted) classifier chains

Bayesian network estimators:

Structure learners:

- 4 constraint-based algorithms
- 2 score-based algorithms
- 2 hybrid algorithms
- 4 local discovery algorithms

Parameter estimation:

- maximum likelihood
- Bayesian a posteriori

Evaluation (2)

Discrete joint densities: Generated from Bayesian networks

- between 4 to 8 nodes, 100 networks for each node count
- 10^3, 10^4, 10^5, or 10^6 instances

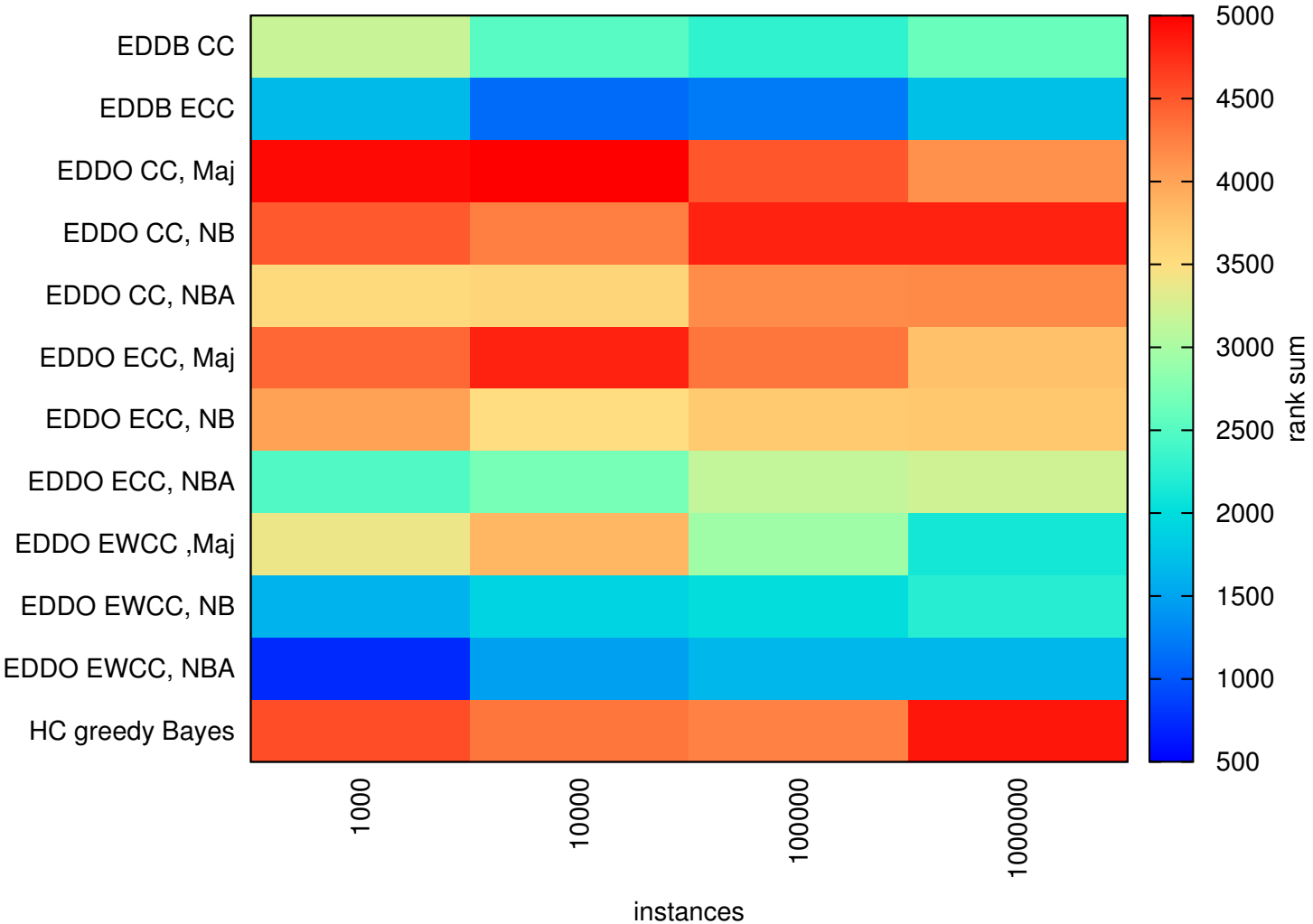
Performance measured by Kullback-Leibler divergence:

$$\sum_i \hat{f}(i) \cdot \ln \frac{\hat{f}(i)}{f(i)}$$

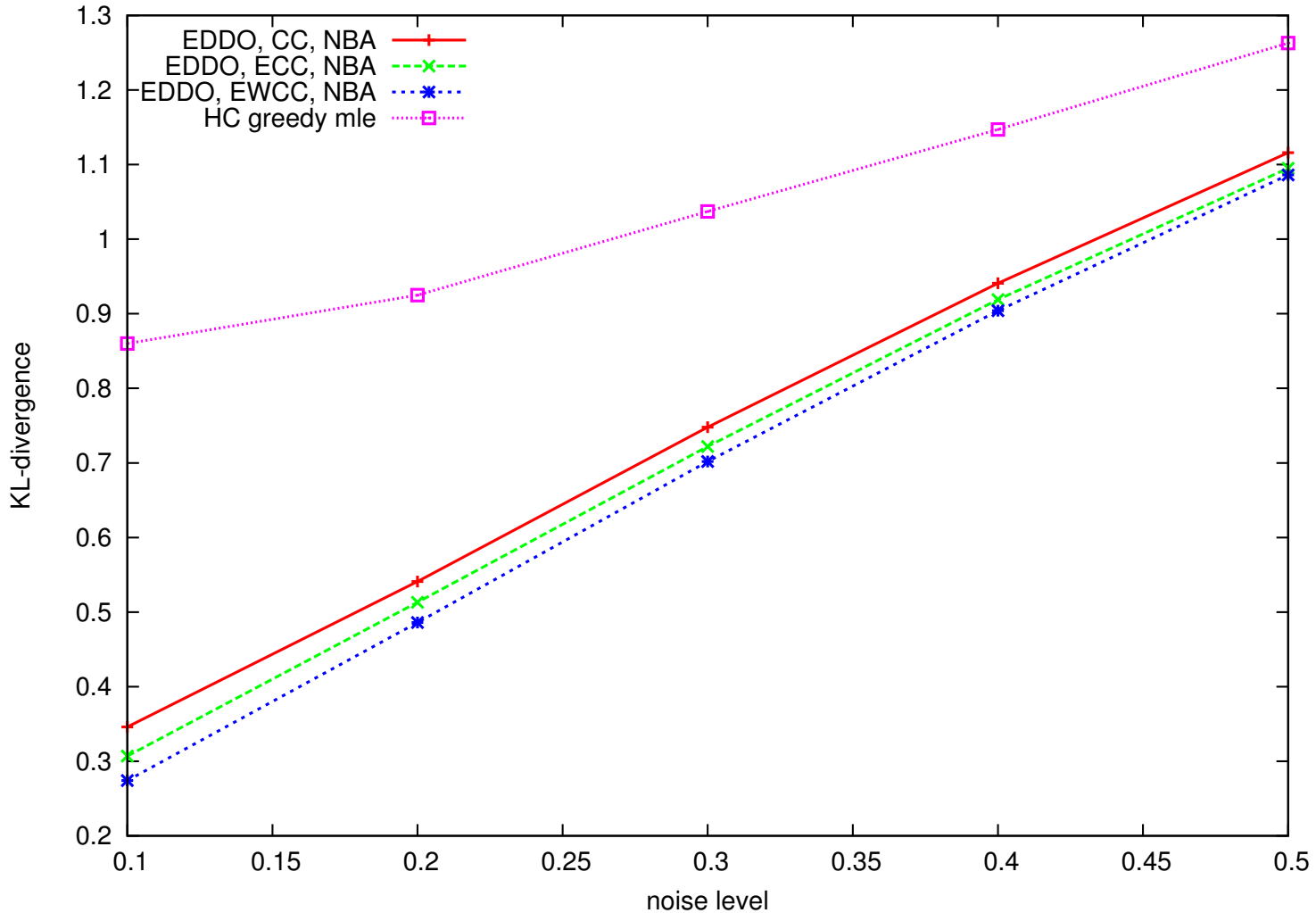
and the average log-likelihood:

$$\frac{1}{n} \cdot \sum_{i=1}^{n} \hat{f}(i)$$

Noise-free Synthetic Data



Noisy Data


EDDO, CC, NBA
EDDO, ECC, NBA
EDDO, EWCC, NBA
HC greedy mle
KL-divergence
noise level
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5

Further Experiments

Concept drift

- window sizes: 25k, 50k, and 75k
- estimators required on average less than 10^6 instances to reach the same KL divergence as before the concept drift

Further Experiments

Concept drift

- window sizes: 25k, 50k, and 75k
- estimators required on average less than 10^6 instances to reach the same KL divergence as before the concept drift

Real-world data

Data set	Instances	Attributes
US Census	2,458,285	68
Covertype (discretized)	581,012	54

EDDO always has a **substantially larger** average log-likelihood when compared to the Bayesian network estimators.

Conclusions and Future Work

Online density estimators:

- for discrete joint densities
- are consistent (proof in the paper)
- enable inference
- perform well on (noisy) synthetic and real-world data
- can cope with concept drifts

Conclusions and Future Work

Online density estimators:

- for discrete joint densities
- are consistent (proof in the paper)
- enable inference
- perform well on (noisy) synthetic and real-world data
- can cope with concept drifts

Future Work:

- broader range of discrete joint densities
- continuous joint densities and conditional densities
- more sophisticated inference algorithms

Thank you for your attention!

Additional slides

Consistency

Proposition

Let f be a discrete joint density with

$$f(X_1, \ldots, X_n) = f_1(X_1) \cdot \ldots \cdot f_n(X_n \mid X_1, \ldots, X_{n-1}).$$

Further, let $\hat{f}$ be an estimator employing a single classifier chain, and let $\hat{f}_i$ be the estimate of the ith classifier in the classifier chain. If the number of instances tends to infinity and $KL(f_i, \hat{f}_i) \to 0$, for all $i \in [1; n]$, then $KL(f, \hat{f}) \to 0$.

Similar result for estimators employing an ensemble of (weighted) classifier chains.

Ensembles of Weighted Classifier Chains

Further improvement: For every instance,

- measure probability p_i for each classsifier chain cc_i
- compute log-likelihood $L := (-\log_2 p_1, \ldots, -\log_2 p_k)$
- $m := \max\{L_i \mid i \in [1; k]\}$
- update weight w_i of classifier chain cc_i:
  $w_i := w_i + (1 - \frac{L_i}{m})$
- normalize the vector $(w_1, \ldots, w_n)$

$\rightarrow$ ensembles of **weighted** classifier chains.