Estimation-Based Search Space Traversal in PILP Environments

Joana Côrte-Real, Inês Dutra, and Ricardo Rocha
{**jcr**, *ines*, *ricroc*}*@dcc.fc.up.pt*

CRACS & INESC TEC
Faculty of Sciences
University of Porto

ILP 2016
4th September 2016

Overview

ILP and PILP

Probabilistic Inductive Logic Programming (PILP) extends ILP by:

- dealing with **probabilistic** facts and rules.
- learning probabilistic theories that can be used for **prediction**.

There are three PILP systems in the literature:

ProbFOIL+ (De Raedt et al., 2015)
SLIPCOVER (Bellodi and Riguzzi, 2015)
SkILL (Côrte-Real et al., 2015)

Motivation

PILP theory search space:

- uses algorithms similar to ILP to generate the logical part of the theories.
- therefore suffers from the same search space traversal efficiency issues as ILP.
- adds a level of complexity w.r.t. ILP because every theory must be probabilistically evaluated.

PILP **pruning strategies** can **reduce the number of candidate theories** based on their probabilistic values, which results in a **shorter execution time**.

What is estimation pruning?

Estimation pruning:

1. calculates the **interval** where the predictions of a combination of two theories may lie.
2. **estimates predictions** for the combination of theories (using a given *estimator*).
3. **excludes** estimated theories that are too specific (AND operation) or too general (OR operation).

Estimators

Estimators can be used to **determine the estimated predictions** of a theory within the possible interval of values.

	AND	OR
minimum	$max(0, A + B - 1)$	$max(A, B)$
maximum	$min(A, B)$	$min(A + B, 1)$
center	$\frac{1}{2}\,(min(A, B) + max(0, A + B - 1))$	$\frac{1}{2}\,(max(A, B) + min(A + B, 1))$
independence	$A \times B$	$A + B - A \times B$
exclusion	$max(0, A + B - 1)$	$min(A + B, 1)$

Hard and Soft Pruning Criteria

Pruning criteria are used to **decide if a theory should be evaluated** exactly based on its estimations.

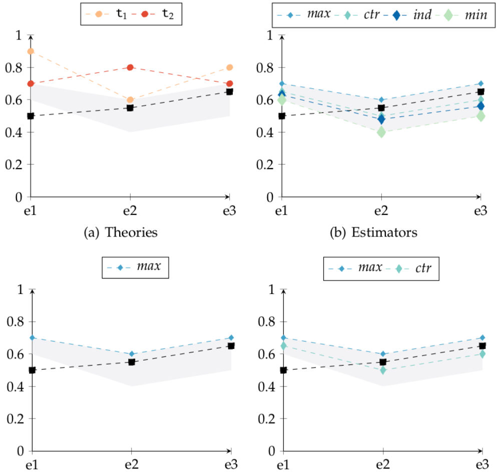
The **hard** pruning criterion excludes theories if they too specific (AND operation)/general (OR operation) for **any estimation**.

The **soft** pruning criterion excludes theories if **all estimations** are overall too specific (AND operation)/general (OR operation).

AND estimation pruning

Squares represent example values, **dots** represent theory predictions and **diamonds** represent estimations of a combination of theories.

Estimation pruning in the AND operation excludes combinations that are **too specific**, i.e. below the example values in Figures (c) and (d).

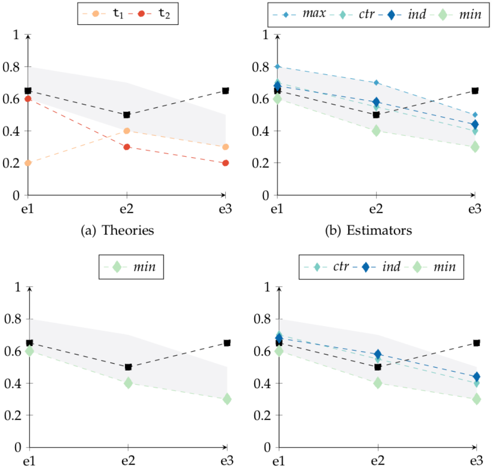


(a) Theories
(b) Estimators
(c) Hard Pruning
(d) Soft Pruning

OR estimation pruning

Squares represent example values, **dots** represent theory predictions and **diamonds** represent estimations of a combination of theories.

Estimation pruning in the OR operation excludes combinations that are **too general**, i.e. above the example values in Figures (c) and (d).



(a) Theories
(b) Estimators
(c) Hard Pruning
(d) Soft Pruning

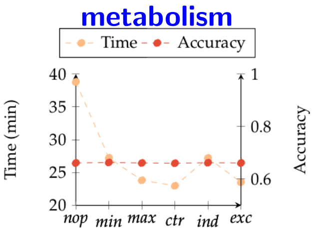
Datasets

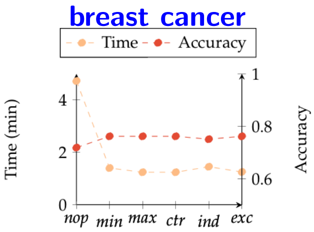
Three datasets were used in the experimental section:

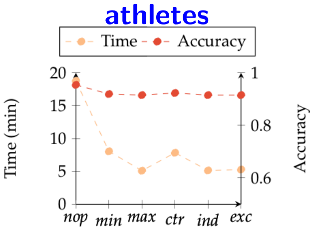
- **metabolism** adaptation of the dataset originally from the 2001 KDD Cup Challenge
- **breast cancer** data from 130 biopsies dating from January 2006 to December 2011
- **athletes** subset of facts regarding athletes and the sports they play collected by the neverending language learner NELL

Dataset	Exs	PBK		Folds		Size train		Size test	
met	230	7000	(46%)	30	n	160	(70%)	70	(30%)
bc	130	13400	(3%)	130	lo	129	(99%)	1	(1%)
ath	721	4294	(100%)	30	n	505	(70%)	216	(30%)

Time and Accuracy Results







- Estimation pruning:
  - **reduced** execution time
  - **maintained** or **increased** accuracy
- Estimators *maximum* and *exclusion* are faster

Pruning Rules and Theories (athletes dataset)

The greatest reduction in evaluated theories corresponds to the **HH setting**, and is consistent with the setting that presents the greatest speedups.

The three fastest estimators (in average) are also the estimators that prune away most theories.

Est	xx	Sx	Hx	xS	xH	SS	HH
min	2414/1989	164/968	164/968	2414/1981	2414/604	164/913	164/361
max	2414/1989	164/968	164/968	2414/69	2414/0	164/243	164/0
ctr	2414/1989	164/968	164/968	2414/1974	2414/381	164/907	164/128
ind	2414/1989	164/968	164/968	2414/69	2414/0	164/243	164/0
exc	2414/1989	164/968	164/968	2414/69	2414/0	164/243	164/0

Summing up

- We proposed **five PILP estimators** reduce the overhead caused by evaluation of candidate probabilistic theories.
- The estimators were implemented in the estimation pruning stage of the SkILL system, but can be **generalized to any PILP engine**.
- Candidate theories can be selected using **two pruning criteria**: soft and hard.
- Experiments using different pruning combinations were performed on **three real-world datasets**.

Discussion and Future Work

Results show that:

1. All estimators **maintain predictive quality** and **reduce execution time**.
2. The **HH pruning setting** showed the greatest speedups and also the greatest reduction in the number of probabilistic evaluations performed.
3. Estimators *maximum* and *exclusion* are overall faster.

Future work includes adding an estimator that divides the estimation interval according to a user-defined distance and dynamically adapting the estimator setting during runtime.

Thank you

Joana Côrte-Real

Bellodi, E. and Riguzzi, F. (2015). Structure learning of probabilistic logic programs by searching the clause space. *Theory and Practice of Logic Programming*, 15(02):169–212.

Côrte-Real, J., Mantadelis, T., Dutra, I., Rocha, R., and Burnside, E. (2015). SkILL - a Stochastic Inductive Logic Learner. In *Proceedings of the 14th International Conference on Machine Learning and Applications*, pages 555–558. IEEE.

De Raedt, L., Dries, A., Thon, I., Van den Broeck, G., and Verbeke, M. (2015). Inducing Probabilistic Relational Rules from Probabilistic Examples. In *International Joint Conference on Artificial Intelligence*, pages 1835–1843. AAAI Press.