Learning Rules With Numerical and Categorical Attributes from Linked Data Sources

Andre de Oliveira Melo

Saarland University

andresony@gmail.com

March 18, 2013

Overview

Semantic Web

Semantic Web

"provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries"

Semantic Web

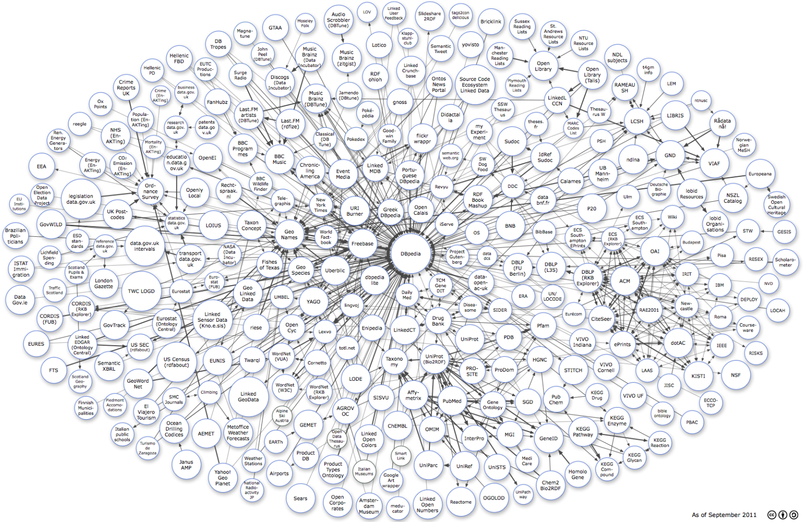
Semantic Web

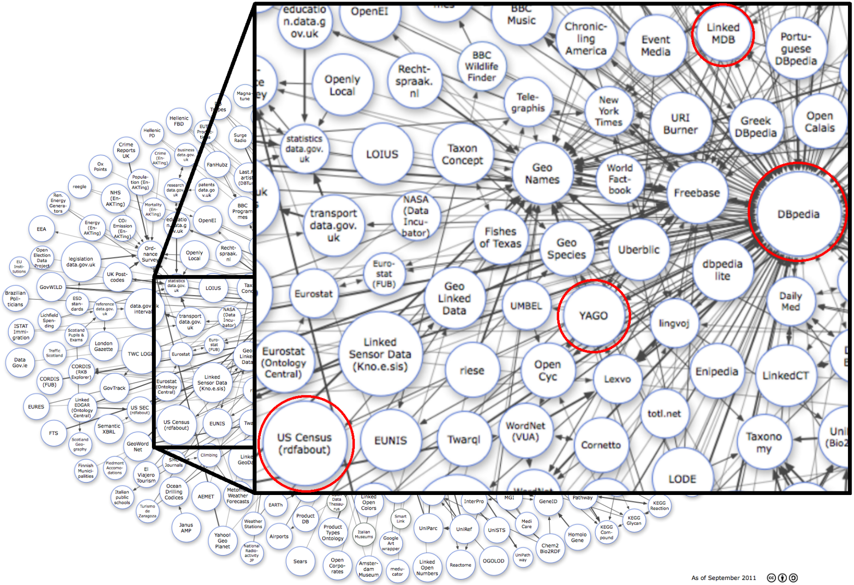
"provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries"

Linked Data

"collection of interrelated datasets on the Web"

"recommended best practises for exposing, sharing, and connecting pieces of data, information and knowledge on the Semantic Web"


LOV
Linked User Feedback
Slideshare 2RDF
tags2con delicious
Audio Scrobbler (DBTune)
Moseley Folk
GTAA
Magna-tune
Klapp-stuhl-club
Bricklink
Sussex Reading Lists
St. Andrews Resource Lists
NTU Resource Lists
DB Tropes
John Peel (DBTune)
Music Brainz (Data Incubator)
Music Brainz (zitgist)
Lotico
Semantic Tweet
yovisto
Man-chester Reading Lists
Open Library
Open Library (Talis)
NDL subjects
t4gm info
Hellenic FBD
EUTC Produc-tions
Hellenic PD
Surge Radio
Discogs (Data Incubator)
RDF ohloh
Linked Crunch-base
Ontos News Portal
Source Code Ecosystem Linked Data
Plymouth Reading Lists
RAMEAU SH
LEM
Crime Reports UK
Crime (En-AKTing)
business data.gov.uk
FanHubz
Music Brainz (DBTune)
Jamendo (DBtune)
gnoss
SSW Thesaur us
LinkedL CCN
Thesau-rus W
ntnusc
Ox Points
Popula-tion (En-AKTing)
Last.FM artists (DBTune)
Last.FM (rdfize)
Poké-pédia
Didactal ia
theses. fr
MARC Codes List
LCSH
LIBRIS
Rådata nå!
reegle
NHS (En-AKTing)
research data.gov.uk
patents data.go v.uk
Classical (DB Tune)
Pokedex
Good-win Family
flickr wrappr
my Experi-ment
Sudoc
PSH
Ren. Energy Genera-tors
Mortality (En-AKTing)
BBC Program mes
BBC Music
semantic web.org
SW Dog Food
IdRef Sudoc
GND
Norwe-gian MeSH
EEA
Energy (En-AKTing)
CO2 Emission (En-AKTing)
educatio n.data.g ov.uk
OpenEI
Chronic-ling America
Event Media
Linked MDB
Portu-guese DBpedia
UB Mann-heim
ndlna
VIAF
Europeana
Open Election Data Project
Ord-nance Survey
Openly Local
Recht-spraak. nl
BBC Wildlife Finder
Revyu
DDC
Calames
Deutsche Bio-graphie
lobid Resources
NSZL Catalog
Swedish Open Cultural Heritage
EU Insti-tutions
legislation data.gov.uk
UK Post-codes
Tele-graphis
New York Times
URI Burner
Greek DBpedia
Open Calais
RDF Book Mashup
data bnf.fr
P20
Ulm
ECS South-ampton
Wiki
lobid Organi-sations
STW
GESIS
Brazilian Poli-ticians
GovWILD
ESD stan-dards
statistics data.gov.uk
LOIUS
Taxon Concept
Geo Names
World Fact-book
Freebase
iServe
OS
BNB
BibBase
ECS South-ampton EPrints
ECS (RKB Explorer)
Budapest
Lichfield Spen-ding
reference data.gov.uk
data.gov.uk intervals
transport data.gov.uk
NASA (Data Incu-bator)
DBpedia
Project Guten-berg
data dcs
DBLP (FU Berlin)
DBLP (L3S)
OAI
Pisa
RESEX
Scholaro-meter
ISTAT Immi-gration
Scotland Pupils & Exams
Fishes of Texas
Geo Species
Uberblic
dbpedia lite
DBLP (RKB Explorer)
IRIT
Traffic Scotland
London Gazette
TWC LOGD
Eurostat
Euro-stat (FUB)
Geo Linked Data
UMBEL
TCM Gene DIT
data-open-ac-uk
ACM
IBM
NVD
Data Gov.ie
CORDIS (RKB Explorer)
Daily Med
ERA
UN/ LOCODE
DEPLOY
YAGO
lingvoj
Disea-some
SIDER
RAE2001
New-castle
LOCAH
CORDIS (FUB)
GovTrack
Eurostat (Ontology Central)
Linked Sensor Data (Kno.e.sis)
riese
Open Cyc
Lexvo
Enipedia
LinkedCT
Drug Bank
Eurécom
CiteSeer
Roma
Course-ware
EURES
Linked EDGAR (Ontology Central)
US SEC (rdfabout)
totl.net
UniProt
PDB
Pfam
VIVO Indiana
ePrints
dotAC
IEEE
RISKS
FTS
Semantic XBRL
US Census (rdfabout)
EUNIS
Twarql
WordNet (VUA)
Cornetto
Taxono my
UniProt (Bio2RDF)
PRO-SITE
ProDom
HGNC
STITCH
VIVO Cornell
LAAS
KISTI
NSF
Scotland Geo-graphy
GeoWord Net
Climbing
Linked GeoData
WordNet (W3C)
WordNet (RKB Explorer)
LODE
SISVU
Affy-metrix
PubMed
Gene Ontology
SGD
Pub Chem
KEGG Drug
VIVO UF
JISC
ECCO-TCP
Finnish Munici-palities
Piedmont Accomo-dations
El Viajero Tourism
SMC Journals
Alpine Ski Austria
AGROV OC
bible ontology
PBAC
Italian public schools
Turismo de Zaragoza
Ocean Drilling Codices
AEMET
Metoffice Weather Forecasts
GEMET
Open Data Thesau-rus
Linked Open Colors
ChEMBL
OMIM
InterPro
MGI
GeneID
KEGG Pathway
KEGG Enzyme
KEGG Reaction
EARTh
Product DB
Smart Link
KEGG Glycan
Janus AMP
Yahoo! Geo Planet
Weather Stations
Airports
Product Types Ontology
Italian Museums
Google Art wrapper
UniParc
UniRef
UniSTS
Medi Care
Homolo Gene
KEGG Com-pound
National Radio-activity JP
Sears
Open Corpo-rates
Amster-dam Museum
medu-cator
Linked Open Numbers
Reactome
OGOLOD
UniPath way
Chem2 Bio2RDF
As of September 2011


educatio n.data.g ov.uk
OpenEI
BBC Music
Chronic-ling America
Event Media
Linked MDB
Portu-guese DBpedia
Openly Local
Recht-spraak. nl
BBC Wildlife Finder
Tele-graphis
New York Times
URI Burner
Greek DBpedia
Open Calais
statistics data.gov. uk
LOIUS
Taxon Concept
Geo Names
World Fact-book
Freebase
DBpedia
transport data.gov. uk
NASA (Data Incu-bator)
Fishes of Texas
Geo Species
Uberblic
dbpedia lite
Euro-stat (FUB)
Eurostat
Geo Linked Data
UMBEL
YAGO
lingvoj
Daily Med
Eurostat (Ontology Central)
Linked Sensor Data (Kno.e.sis)
riese
Open Cyc
Lexvo
Enipedia
LinkedCT
totl.net
US Census (rdfabout)
EUNIS
Twarql
WordNet (VUA)
Cornetto
Taxono my
LODE
As of September 2011

Motivation

Learn Datalog rules from data:

$$\underbrace{livesIn(X, Y)}_{head} \text{ :- } \underbrace{isMarriedTo(X, Z), livesIn(Z, Y)}_{body}$$

Support and confidence thresholds

- Support: $supp(head \text{ :- } body) = supp(head \wedge body)$
- Confidence: $conf(head \text{ :- } body) = \dfrac{supp(head \wedge body)}{supp(body)}$

Rules with constants

Refining rules with constants is relevant

$$speaks(X, Z) \text{ :- } livesIn(X, W)$$

Searching constants for Z and W we can learn:

$$speaks(X, englsih) \text{ :- } livesIn(X, australia)$$
$$speaks(X, spanish) \text{ :- } livesIn(X, argentina)$$
$$speaks(X, portuguese) \text{ :- } livesIn(X, brasil)$$

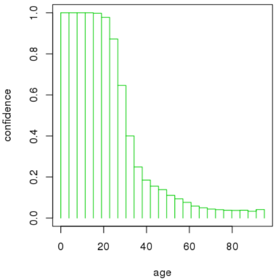
What about numerical constants?

$$speaks(X, english) \text{ :- } hasIncome(X, \$3.71Billion)$$
$$speaks(X, portuguese) \text{ :- } livesIn(X, W), hasPopulation(W, 193946886)$$

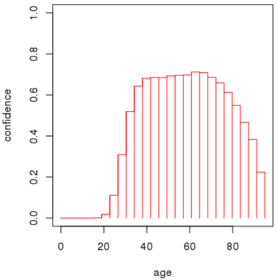
Refining rules with numerical intervals

maritalStatus(X, single) :- age(X,Y) [conf=0.40]
maritalStatus(X,married) :- age(X,Y) [conf=0.46]
maritalStatus(X,widowed) :- age(X,Y) [conf=0.06]

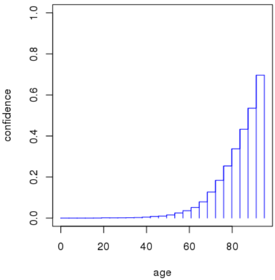
Single



Married



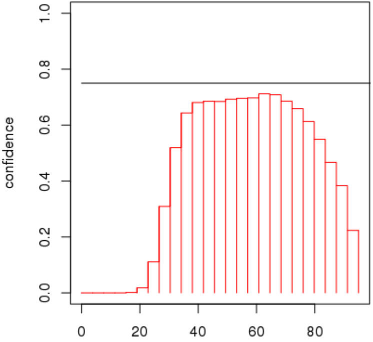
Widowed



Combine with categorical constants

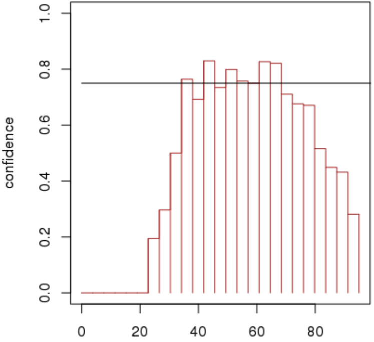
For $maritalStatus(X, married)$, $minConf = 0.75$ is not satisfied.
Refine by State?

USA



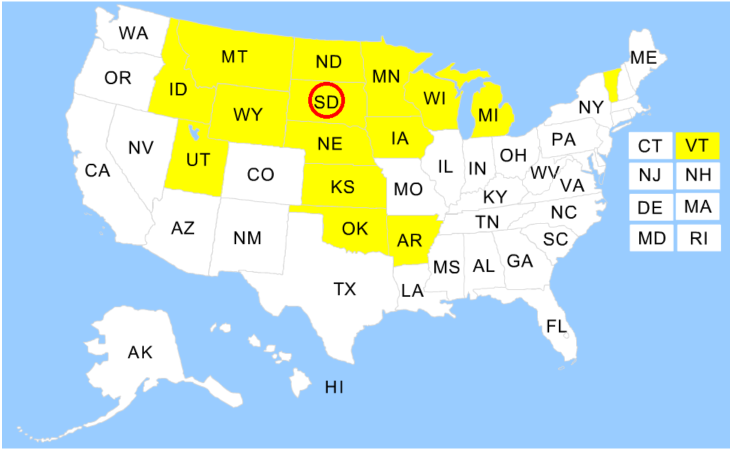
$\Rightarrow$

South Dakota



Combine with categorical constants

We can find intervals for Y that satisfy $minConf = 0.75$ for *maritalStatus(X,widowed) :- livesIn(X,sd),age(X,Y)* and...



Base-rule and Refined-rule

- **Base-rule**: Numerical argument with no constant
  r_1 : $marritalStatus(X, single)$:- $livesIn(X, sd), age(X, Y)$
  [conf=0.49,supp=2368]
- **Refined-rule**: Base-rule with restricted numerical variable
  r_2 : $marritalStatus(X, single)$:- $livesIn(X, sd), age(X, Y), Y \in [33, 67]$
  [conf=0.77,supp=1092]

We are interested in refinements that bring a significant confidence gain:

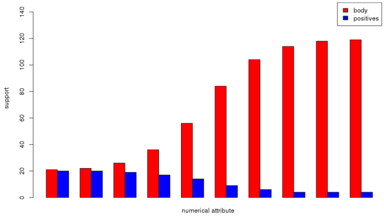
$$gain_{r_{ref}, r_{base}} = \frac{conf(r_{ref})}{conf(r_{base})} \tag{1}$$

For our example: $gain_{r_2, r_1} = \frac{0.77}{0.49} = 1.57$

What base-rules have refined-rules with significant confidence gain?

- Satisfy support threshold
- Do not necessarily satisfy confidence threshold
- Divergent body and positives (body$\wedge$head) probability distributions

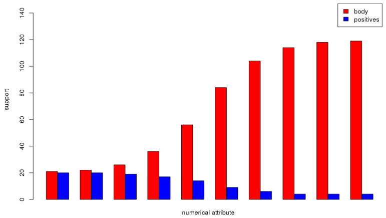
Frequency histograms



What base-rules have refined-rules with significant confidence gain?

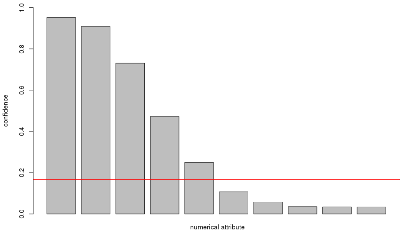
- Satisfy support threshold
- Do not necessarily satisfy confidence threshold
- Divergent body and positives (body$\wedge$head) probability distributions

Frequency histograms



$\Rightarrow$

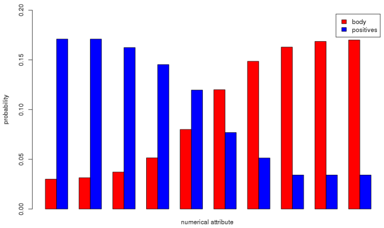
Confidence distribution



What base-rules have refined-rules with significant confidence gain?

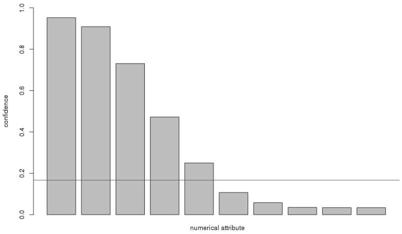
- Satisfy support threshold
- Do not necessarily satisfy confidence threshold
- Divergent body and positives (body$\wedge$head) probability distributions

Body and Positives distributions



$\Rightarrow$

Confidence distribution



Motivation

Problem?

- Search space grows exponentially with the number of predicates and constants
- Querying support and confidence distributions is very expensive

Idea:

- Analyze combinations of numerical and categorical properties
- Measure their level of interestingness
- Extend top-down ILP to detect and suggest interesting combinations

Logic Programming Concepts

- Literal: predicate symbol with bracketed n-tuple, e.g:
  $L = livesIn(X, Y)$
- Clause: a disjunction of literals (negated or not), e.g:
  $c = (L_1 \vee L_2 \vee \ldots \vee \neg L_{m-1} \vee \neg L_m)$
- Safe Datalog Rule: every variable in the head appear in a non-negated literal in the body, negated literal variables in the body should appear in some positive literal in the body, e.g.:
  $speaks(X, Y)$:- $wasBornIn(X, Z), hasOfficialLanguage(Z, Y)$
- Hypothesis: a set of clauses $\mathcal{H}$
  - Completeness: $\mathcal{H}$ covers all positive examples
  - Consistency: $\mathcal{H}$ covers no negative examples

Inductive Logic Programming (ILP)

Inductive Logic Programming: Finds a hypothesis $\mathcal{H}$ that covers all positive, and no negative examples

$$positiveExamples + negativeExamples + backgroundKnowledge \rightarrow hypothesis$$

Training Examples	**Background Knowledge**
daughter(mary,ann) +	parent(ann,mary)
daughter(eve,tom) +	parent(ann, tom)
daughter(tom,ann) -	parent(tom,eve)
daughter(eve,ann) -	parent(tom,ian)
	female(ann)
	female(mary)
	female(eve)

$$\mathcal{H} = daughter(X, Y) \text{ :- } female(X), parent(Y, X)$$

Inductive Logic Programming (ILP)

Approaches

- Bottom-up: Start with least general $\mathcal{H}$ then perform generalizations
- **Top-down**: Start with most general $\mathcal{H}$ then perform specializations
  - Specialization loop: adds literals to a clause and ensures consistency
  - Covering loop: adds clauses to the hypothesis and ensures completeness
  - Apriori-style pruning

What about large, noisy, and incomplete LOD datasets such as YAGO and DBpedia?:

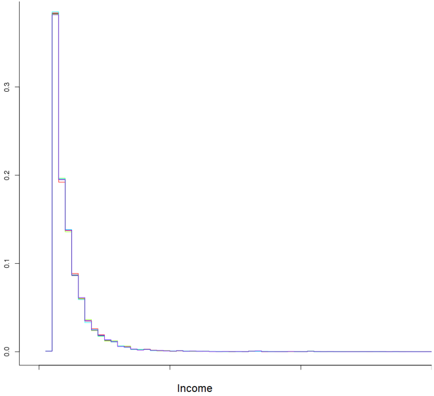
- Sample data to reduce size
- Restrict the number of literals in a clause
- Tolerate a certain level of inconsistency and incompleteness

$$\text{Expected Accuracy: } A(c) = P(e \in \mathcal{E}^+ | c) = \frac{n^+(c)}{n^+(c) + n^-(c)}$$

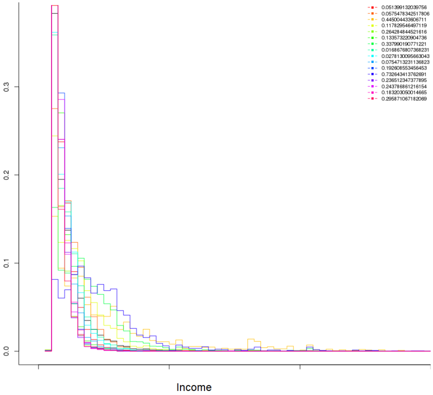
Correlation between Literals

Let's say we want to refine a clause with $hasIncome(X, Y)$ with an interval for Y. Refine by $quarterOfBirth$ or $hasEducation$?

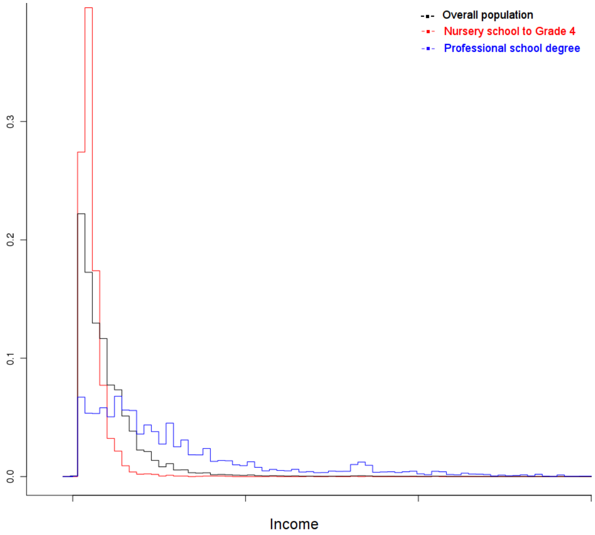
$hasIncome(X, Y), quarterOfBirth(X, Z)$



$hasIncome(X, Y), hasEducation(X, Z)$



Correlation between Literals



Educational Levels
Nursery school to grade 4
Grade 5 or grade 6
Grade 7 or grade 8
Grade 9
Grade 10
Grade 11
Grade 12 no diploma
High school graduate
Some college ($<$ 1 year)
Some college ($\geq$ 1 year)
Associate's degree
Bachelor's degree
Master's degree
Professional school degree
Doctorate degree

Interestingness Measure

How to measure the interestingness of adding a literal l to a clause c?

- Extract the frequency histograms of $\{c\}$ and $\{c \wedge l\}$ over a numerical attribute Y
- Normalize the histograms to obtain their probability distributions, and measure their divergence (e.g., with Kullback-Leibler)

But, divergence alone isn't a good idea because:

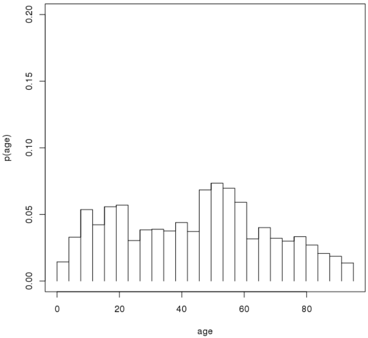
- Lower support histograms are more likely to have a divergent distribution (*sampling error*)
- Rules with high support are still interesting

Then combine both measures: divergence*support

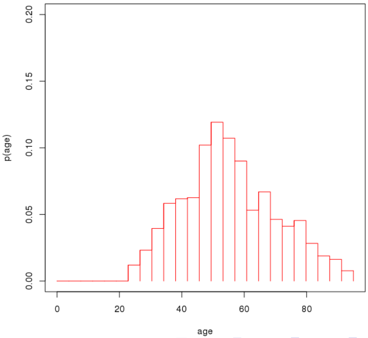
Interestingness Measure

$$\underbrace{maritalStatus(X, married)}_{l=head} \text{ :- } \underbrace{livesIn(X, sd), age(X, Y)}_{c=body}$$

age(X,Y),livesIn(X,sd)



age(X,Y),livesIn(X,sd),marriedStatus(X,married)



Correlation Lattice

- Build a lattice similar to an *itemset lattice*
- Numerical property $r(X, Y)$ as root
- The "items" are literals that can be joined with the root's non-numerical variable X
- Root's numerical attribute Y is discretized in k buckets $\{b_1, \ldots, b_k\}$
- Each node x has a frequency histogram $h(x) =< h_1(x), \ldots, h_k(x) >$ from its clause support distribution

  where $h_i(x) = supp(x|Y \in b_i)$ and $|h(x)|_1 = supp(x)$

Correlation Lattice

$$hasIncome(X, Y)$$

Correlation Lattice

$hasIncome(X, Y)$

$\ldots$

$hasIncome(X, Y),$
$quarterOfBirth(X, q1)$

$hasIncome(X, Y),$
$quarterOfBirth(X, q4)$

Correlation Lattice

$hasIncome(X, Y)$

$\ldots$ $\ldots$

$hasIncome(X, Y)$, $quarterOfBirth(X, q1)$

$hasIncome(X, Y)$, $quarterOfBirth(X, q4)$

$hasIncome(X, Y)$, $hasEducation(X, nursery)$

$hasIncome(X, Y)$, $hasEducation(X, phd)$

Correlation Lattice

$hasIncome(X, Y)$

. . .   .

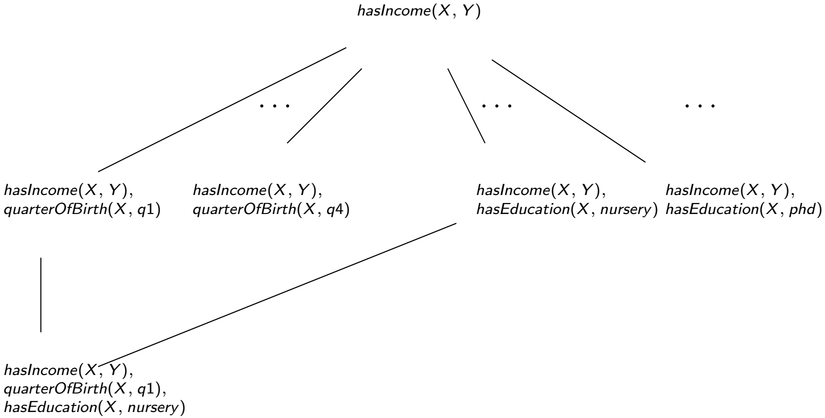
$hasIncome(X, Y)$, $quarterOfBirth(X, q1)$

$hasIncome(X, Y)$, $quarterOfBirth(X, q4)$

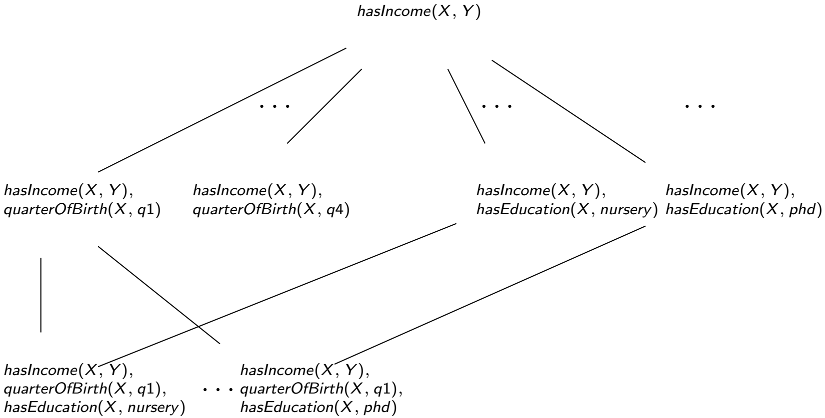
$hasIncome(X, Y)$, $hasEducation(X, nursery)$

$hasIncome(X, Y)$, $hasEducation(X, phd)$

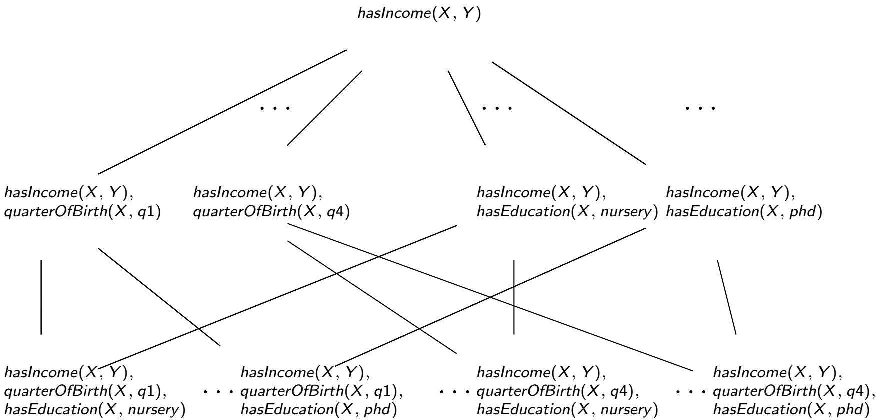
Correlation Lattice



Correlation Lattice



Correlation Lattice



Correlation Lattice

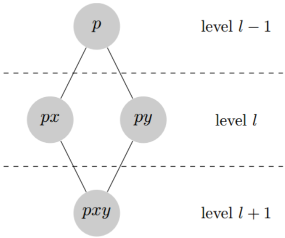
- Number of nodes in a lattice with ℓ levels n properties and m constants per property:

$$\sum_{i=1}^{\ell} \binom{nm}{i} \tag{2}$$

- Too expensive, we need to reduce size
  - Prune by support (safe)
  - Restrict ℓ to the maximum clause size allowed in the core-ILP
  - Restrict the literals added to the lattice in order to reduce n and m
  - Prune by interestingness or independence (heuristics)

Independence checks

- Checks if a pair of nodes joining nodes are independent given their common parent



(where p is a clause, x and y are literals, s.t. $x \neq y$ and $x, y \notin p$)

- Estimate $\hat{h}(pxy)$ assuming independence of x and y given p
- Query actual $h(pxy)$ and perform a Pearson's chi-squared test

  $H_0 = x$ and y are independent given p

  $H_1 = x$ and y are dependent given p

Independence checks

$hasIncome(X, Y)$
$h =< 6, 10, 8, 4, 2, 1 >$

$hasIncome(X, Y), marritalStatus(X, single)$
$h =< 4, 5, 4, 3, 1, 0 >$

$hasIncome(X, Y), employmentStatus(X, employed)$
$h =< 3, 6, 6, 4, 2, 1 >$

$hasIncome(X, Y), marritalStatus(X, single), employmentStatus(X, employed)$
$h =< 2, 4, 2, 4, 1, 0 >$
$\hat{h} =< 2, 3, 3, 3, 1, 0 >$

$$\chi^2 = \sum_{i=1}^{k} \frac{(h_i - \hat{h}_i)^2}{\hat{h}_i} = 1 \quad \Rightarrow \quad p\text{-}value = 0.96$$

Independence checks

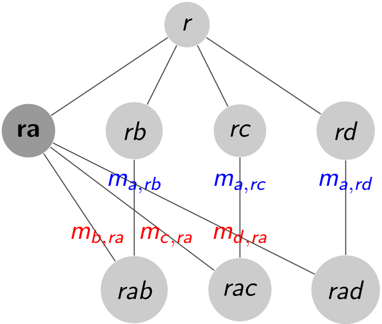
- If there's not enough evidence of dependence, we assume independence, then:

  x :- $p, y \equiv x$:- p

  y :- $p, x \equiv y$:- p
- The lower the p-value (greater χ^2), the greater the evidence that x and y are dependent given p, therefore the more interesting it is to join the nodes py and px
- As heuristics, we can set a maximum *p-value* threshold to prune independent nodes

Refinement Suggestions

In the ILP refinement loop, the clauses have a fixed head while the body is refined. Assuming we have *a* as head literal, *r* as root and *b*, *c*, *d* as possible new literals:

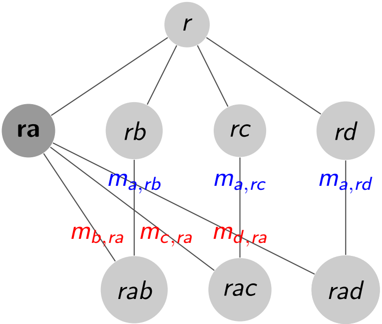


$$a \;\Big|\; \begin{array}{l} b\ [m_{a,rb}] \\ c\ [m_{a,rc}] \\ d\ [m_{a,rd}] \end{array}$$

What literal is more interesting to add to the clause a:-r?

Refinement Suggestions

In the ILP refinement loop, the clauses have a fixed head while the body is refined. Assuming we have a as head literal, r as root and b, c, d as possible new literals:



a	b $[m_{a,rb}]$
	c $[m_{a,rc}]$
	d $[m_{a,rd}]$

What literal is more interesting to add to the clause a:-r?

$argmax_{i \in \{b,c,d\}} m_{a,ri}$

Search in the Lattice

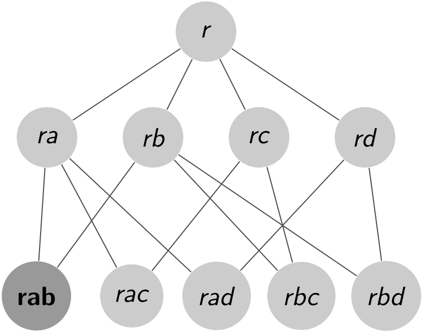
What has to be done?

- Search the node with body literals
- For each child of such node check head literal can be further added, if so collect the new literal and the interestingness value of adding the head
- Sort the possible new literals by interestingness

Alternative?

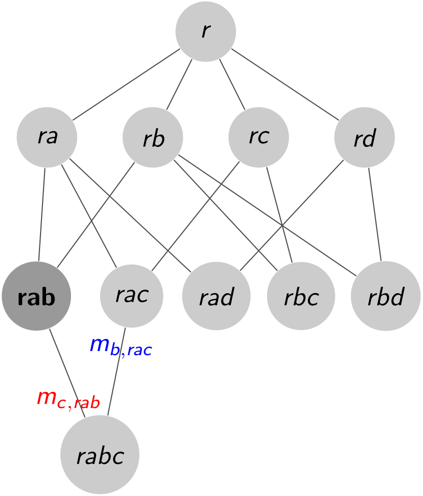
- Create mapping in every node with the possible head literals as key and sorted literals to be added to body as value
- Only add entry if head and new literal not independent given body

Refinement Suggestions



Suggestions Map

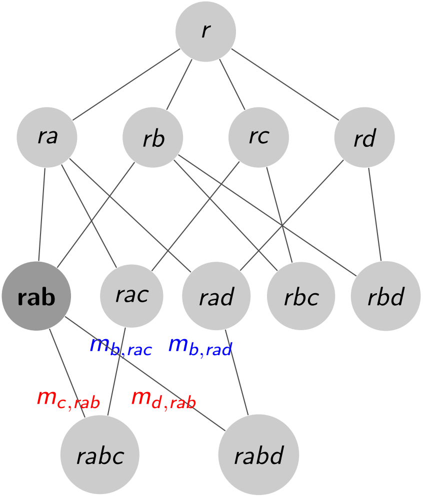
Refinement Suggestions



Suggestions Map

b	$c\ [m_{b,rac}]$

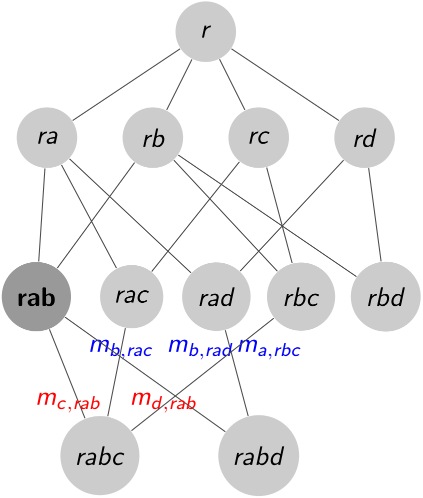
Refinement Suggestions



Suggestions Map

b	$c\ [m_{b,rac}]$ $d\ [m_{b,rad}]$

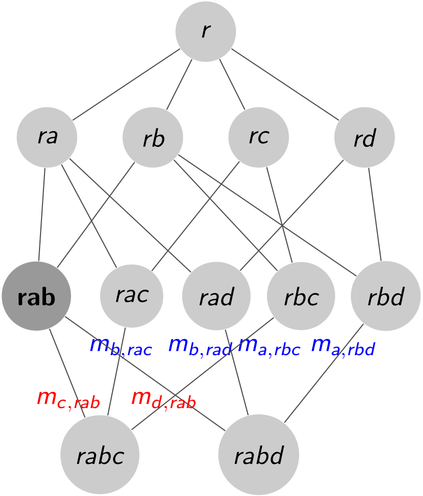
Refinement Suggestions



Suggestions Map

b	$c\ [m_{b,rac}]$ $d\ [m_{b,rad}]$
a	$c\ [m_{a,rbc}]$

Refinement Suggestions



Suggestions Map

b	$c\ [m_{b,rac}]$ $d\ [m_{b,rad}]$
a	$c\ [m_{a,rbc}]$ $d\ [m_{a,rbd}]$

Incorporating the Lattice in the Core-ILP

In the refinement step, we detect clauses with body containing a lattice root

- If clause satisfies support threshold and does not satisfy confidence threshold
- Then search in lattice for body literals and head
- Check the interestingness of adding the head to the body and analyze whether to search for numerical intervals
- Query the lattice for suggestions of interesting literals to be added to the clause

Experiments

Overall Settings:

- We compare 4 interestingness measures:
  1. ● *supp*: Support Only
  2. ■ *klsupp*: KL-divergence*Support
  3. ● *kldiv*: KL-divergence Only
  4. $\star$ *jssupp*: JS-divergence*Support
- Thresholds:
  - $minConf = 0.75$
  - $minSupp = 25$
  - $minGain = 1.25$

1^{st} Experiment: evaluation of the Correlation Lattice

- All data joined by person only (anonymized)
- All properties categorical (categories as literals)
- Create a lattice for *hasIncome* property

2^{nd} Experiment: evaluation of the ILP extension

- All data joined by person only (anonymized)
- All properties categorical (categories as literals)

1^{st} Experiment

Figure: Build time per lattice size

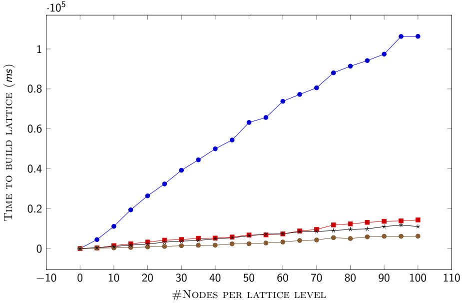
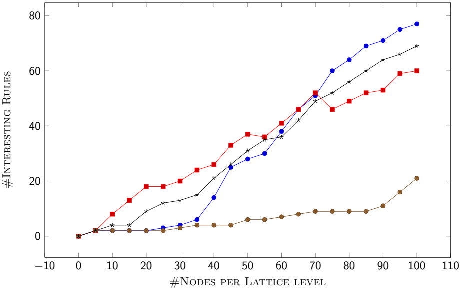


Figure: Interesting rules per lattice size



Legend: [● *supp* ■ *klsupp* ● *kldiv* ⋆ *jssupp*]

2^{nd} Experiment

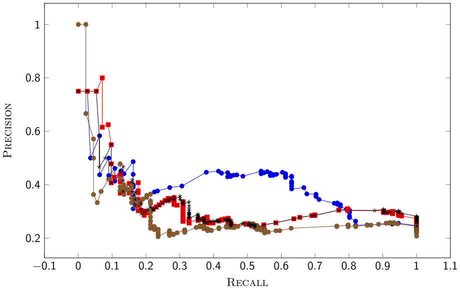


Figure: Precision-Recall graph from interestingness predictions (rules with *runtime* attribute)

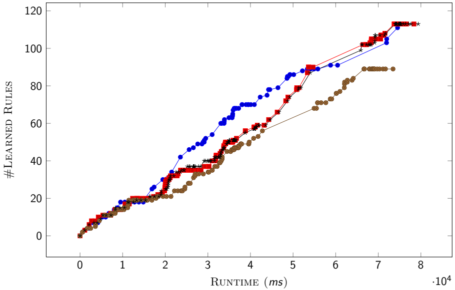


Figure: Interesting rules per runtime (rules with attribute *runtime*)

Legend: [● *supp* ■ *klsupp* ● *kldiv* ⋆ *jssupp*]

1^{st} Experiment

Figure: Precision-Recall graph from interestingness predictions (rules with *budget* attribute)

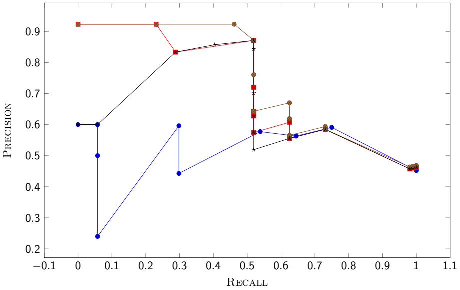
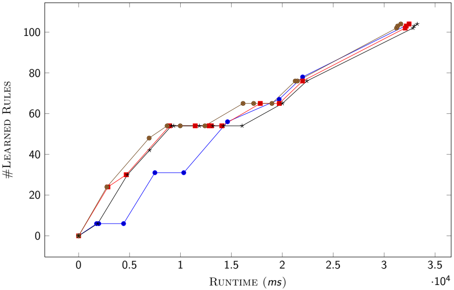


Figure: Interesting rules per runtime (rules with *budget* attribute)



Legend: [● *supp* ■ *klsupp* ● *kldiv* ⋆ *jssupp*]

Thank you