Incorporating causal reasoning in cost-effective action mining

Hendrik Blockeel
Dept. of Computer Science, KU Leuven

Based on P. Shamsinejadbabaki, M.H. Saraee, H. Blockeel, *Causality-based cost-effective action mining*, submitted, 2012

COSTS workshop @ ICDM 2012

DTAI
Machine Learning

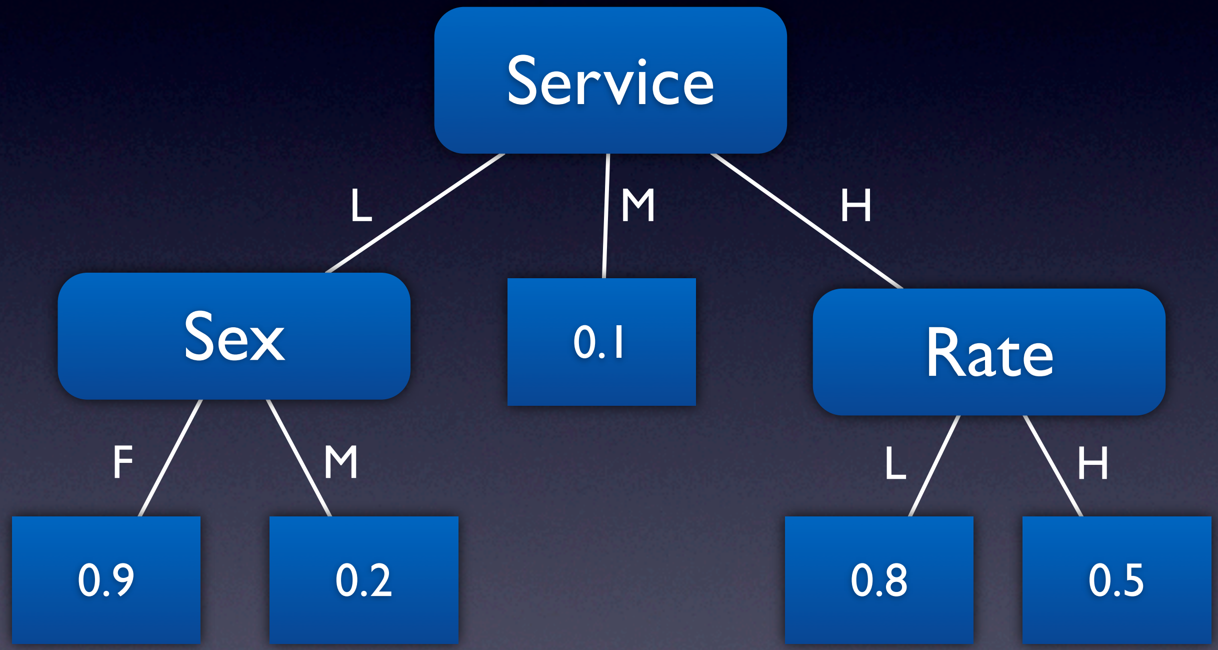
Overview

- Context: Action rule mining
- A weakness of current approaches: correlation $\neq$ causality
- Incorporating causal reasoning in action rule mining: A concrete approach
- Experimental validation
- Conclusions

Common setting: learning predictive models

- Goal: predict value of target attribute from some other attributes
  - transductive: make a prediction
  - inductive: learn a predictive model
- Much work on this: models in the form of if-then-rules, decision trees, ANNs, SVMs, probabilistic models, ...

Predictive model: decision tree



tree indicates loyalty for customers in different groups

can be used to predict loyalty for new customer

(example from Yang et al., ICDM 2003)

Predictive model: association rule

- Association rule:

IF bread & cheese THEN wine (14%)

- Suppose wine is bought by 6% of total population, but 14% of B&C subpopulation; then this rule tells us: *people who buy bread & cheese are more likely to buy wine*

Action mining

Action mining is not concerned with the question:

Given some values for non-target attributes, what's the most likely value for the target?

but, instead, with the question:

Given a desirable value v for the target, how should we change the non-target attributes to make the target equal to v?

(e.g., how can we make a customer more profitable?)

Term coined & initial work done by Z. Ras & others

Setting: "cost-effective action mining"

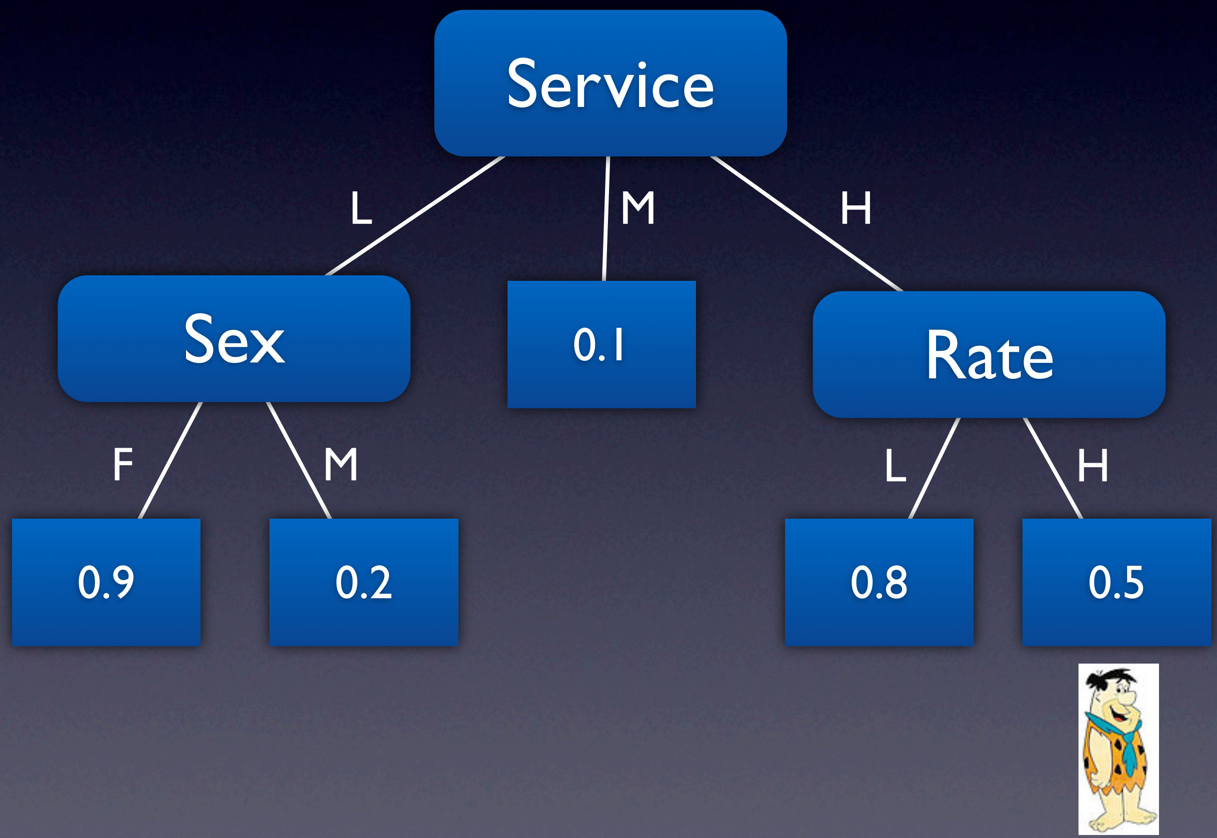
- We are given:
  - A set of attributes A_i with domains D_i, and cost functions C_i: $D_i \times D_i \rightarrow \mathbb{R}$
  - A "target attribute" T with domain D_T and profit function P: $V \rightarrow \mathbb{R}$
- An action A is a set of externally induced changes $a_i \rightarrow a_i'$ of attribute values ("interventions")
- The cost of an action is the sum of the costs of the changes: $C(A) = \sum_{(a_i \rightarrow a_i') \in A} C_i(a_i, a_i')$

- Changing one attribute may have an effect on other attributes or on the target
- Let t be the original (pre-action) value of the target, and t' the new value
- The <u>profit</u> of an action A is $P(t')-P(t)$
- The <u>net profit</u> of A is $NP(A)=P(t')-P(t)-C(A)$
  - this assumes t' is known
- The <u>expected net profit</u> of A is $ENP(A)=\boldsymbol{E}(P(t'))-P(t)-C(A)$
  - t' not known

Action (rule) mining

- Given the C_i and P functions and a dataset $D \subseteq D_1 \times \ldots \times D_n \times D_T$
- Find:
  - For a given instance x, the action with highest ENP ["action mining", transductive]
  - A set of rules that predict for any instance x the action with highest ENP ["action rule mining", inductive]

Is it straightforward?



Fred has high service level, high rate; can we make him more loyal?

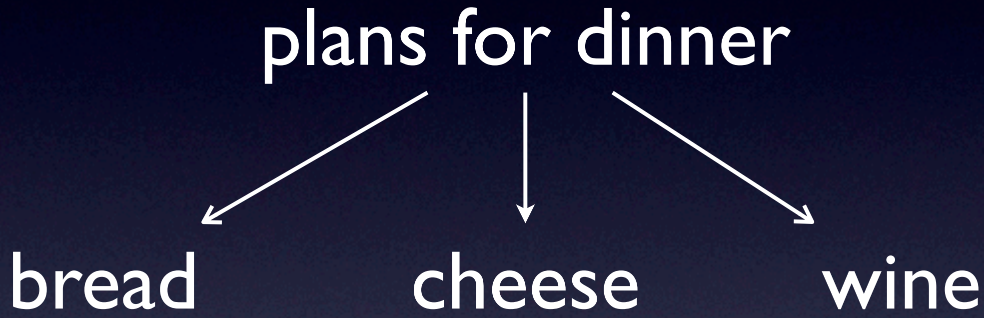
Is it straightforward?

IF bread & cheese THEN wine

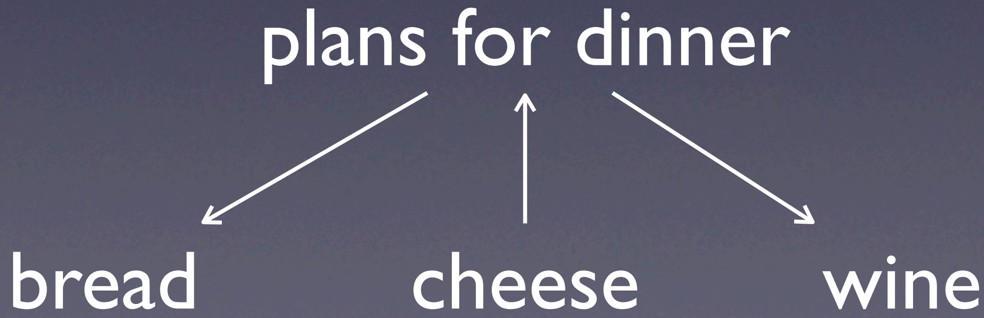
- Suppose many people buy bread, but few buy cheese; and we want to sell more wine (high profit). Can we achieve that by giving them cheese for free?

It is not straightforward

- The real question is: will changing a value **cause** the target value to change?
- Causal information is necessary!
- Existing methods (e.g., Yang et al.) implicitly assume
  - each A_i causally affects T
  - no A_i causally affects any A_j, $j \neq i$



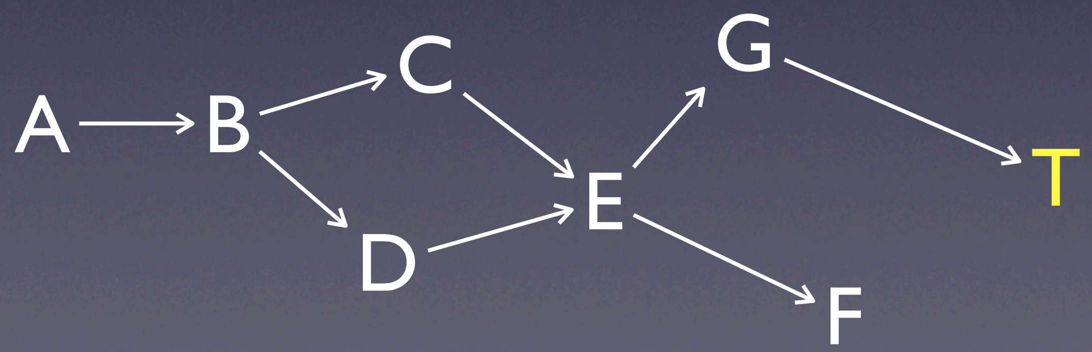
Setting 1:
dinner plans affect
bought products



Setting 2:
promotion affects
dinner plans

Incorporating causal information

- Causal information can be represented as a causal network
- Case 1: causal network is available
- Case 2: causal network is not available



CREAM

- Causal Relationships based Economical Action Mining
- Given a causal network, and a set of interventions (= an action), we can compute the effect on T (standard algorithm)
- Our task: find the action that results in maximal ENP
- Straightforward approach: try many different actions, see how they affect target

CREAM algorithm

Algorithm 1 The CREAM algorithm for learning cost-effective action sets from causal networks (greedy version).

procedure CREAM (T, O, $\mathcal{C}$, p_g, $\mathcal{CN}$)
    Input:target attribute T,
        object data O,
        cost data $\mathcal{C}$,
        profit p_g,
        underlying causal network $\mathcal{CN}$,
    Output: one action set for each object $o \in O$

  $\mathbf{O}^- \leftarrow \{o \in O | Pr(T(o) \neq t_g) > 0\}$
  for each o **in** $\mathbf{O}^-$ **do**
    $\mathbf{I} \leftarrow$ *findCandidateActions(o, $\mathcal{CN}$)*
    $\Gamma \leftarrow$ empty action set
    repeat
      $\alpha_{max} \leftarrow \arg\max_{\alpha \in \mathbf{I}} np(\Gamma \cup \{\alpha\}, o)$
      if $np(\Gamma \cup \{\alpha_{max}\}, o) > np(\Gamma, o)$ **then**
        $\Gamma \leftarrow \Gamma \cup \{\alpha_{max}\}$
        $\mathbf{I} \leftarrow \mathbf{I} - \{\alpha_{max}\}$
    until $\mathbf{I} = \emptyset$ **or** $np(\Gamma \cup \{\alpha_{max}\}, o) \leq np(\Gamma, o)$
    assign Γ to o

Here greedy (hill-climbing) construction of action;
an exhaustive version was also implemented.
Will compare CREAM(GS) vs. CREAM(ES).

Case 2: no causal information

- CREAM assume a causal network is given
- Usually, this is not the case
- Can we *learn* the causal network from the data? If yes, problem solved...

ICE-CREAM

- "IC-Enabled CREAM"
- IC, Inductive Causation, is an algorithm for learning causal networks from data (Verma & Pearl, 1991)
- *Is it actually possible to learn causal relationships by just observing data (not intervening)?*
  - Statistics textbooks: "correlation can be determined from observations alone, but causation cannot"
  - Pearl (1990-...): In some cases (and under mild assumptions), we *can* determine causation from observations!

Inferring causation: the basic idea

Suppose there is evidence that A and B are directly dependent, and B and C too, but no direct connection between A and C (could be based on pre-existing knowledge, or observations of dependencies)

$A — B — C$

No direct link between A and C; all information flow goes through B

$A \longrightarrow B \longrightarrow C$

$A \longleftarrow B \longleftarrow C$

$A \longleftarrow B \longrightarrow C$

$A \longrightarrow B \longleftarrow C$

4 different causal connections possible

Inferring causation: the basic idea

Find a number of cases with the same value for B...

$A \longrightarrow B \longrightarrow C$

- A and C correlate
- Fixing B removes correlation

$A \longleftarrow B \longleftarrow C$

- A and C correlate
- Fixing B removes correlation

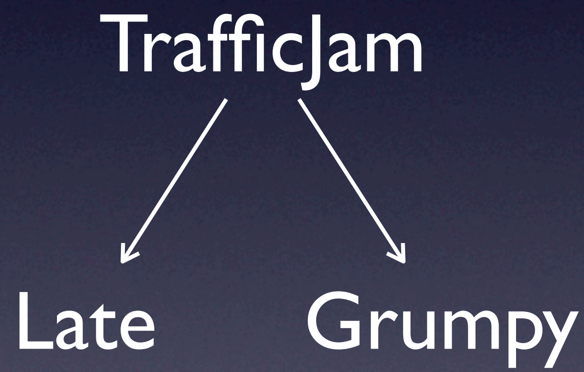
$A \longleftarrow B \longrightarrow C$

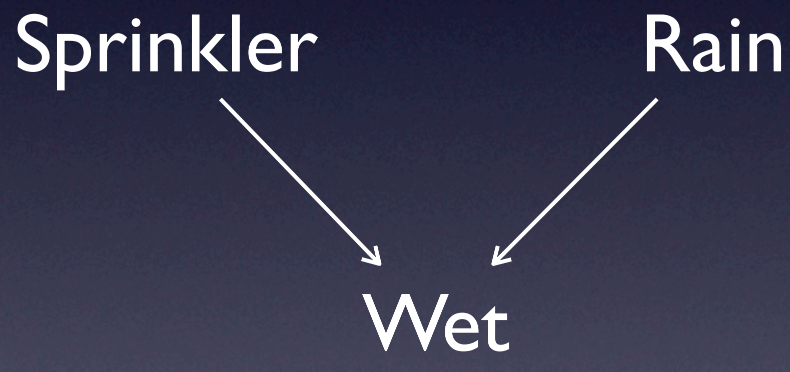
- A and C correlate
- Fixing B removes correlation

$A \longrightarrow B \longleftarrow C$

- A and C *do not* correlate
- Fixing B introduces correlation

Examples



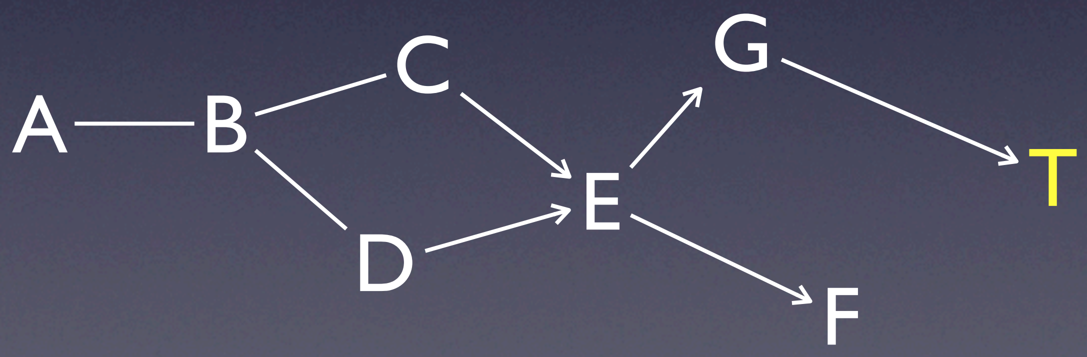


Late $\not\perp\!\!\!\perp$ Grumpy
Late $\perp\!\!\!\perp$ Grumpy | TrafficJam

Sprinkler $\perp\!\!\!\perp$ Rain
Sprinkler $\not\perp\!\!\!\perp$ Rain | Wet

Partial causal networks

- For some edges in a network, the direction can be determined; for others it cannot
- This gives only partial causal information
- How can we deal with that?



What is the effect of A on T?

- The question cannot be answered with certainty: not enough information
- But we need to do something with it...
- Our solution: make different guesses of the complete network, perform inference in these, combine results.
- Ugly, but no better solution available.

ICE-CREAM algorithm

- Run IC (actually a variant by Margaritis, 2003) to derive a partial causal network
- For any action A, estimate ENP(A) as follows:
  - repeat n times: create a random complete network CN consistent with the partial one; compute ENP in CN
  - return the average of all ENPs thus computed
- Otherwise, same as CREAM

Experiments

- Experiments on some "real" (pre-existing) and artificial (created for this purpose) datasets
- For all these datasets, we know the real causal model
- Thus, we can compare:
  - methods that ignore causality (e.g., Yang et al.'s)
  - methods that use the causal network (CREAM)
  - methods that use the estimated, partial causal network (ICE-CREAM)

Results

Average ENP of actions suggested by the method:

Network	CREAM(ES)	CREAM(GS)	ICE-CREAM(ES)	ICE-CREAM(GS)	Yang
ChestClinic	0.58	0.58	0.49	0.49	0.41
Fire	0.81	0.81	0.81	0.81	0.80
usa2000	0.75	0.71	0.66	0.59	0.56
Headache *	0.73	0.72	0.71	0.71	0.22
Alarm *	0.56	0.56	0.54	0.54	0.11
Hailfinder *	0.89	0.90	0.80	0.79	0.63
sample7	0.35	0.35	0.34	0.34	0.25
sample15 *	0.39	0.39	0.36	0.36	0.23
sample30 *	0.35	0.37	0.28	0.30	0.14
sample45 *	0.40	0.39	0.35	0.34	0.17

Conclusions

- Traditional methods for action rule learning make strong assumptions about causality
- Better results are possible by taking real causation into account (CREAM)
- It is possible to (incompletely) learn causal relationships from data (IC)
- Incorporating limited information about causality can give much more accurate action rules