Carleton UNIVERSITY

The Causality/Repair Connection in Databases:

Causality-Programs

Leopoldo Bertossi

Carleton University
Ottawa, Canada

Database Causality and Goals

- Causality appears at the foundations of many scientific disciplines
- We want to represent and compute causality in order to deal with the *uncertainty* in data and knowledge
- In data management, we need to understand and compute why certain (query) results are obtained or not

  Why certain natural semantic conditions are not satisfied
- A DB system could provide *explanations*

  To understand/explore the data or reconsider the query

- Our current research is motivated by trying to understand *causality in data management* from different perspectives

- We have established interesting and fruitful connections among different forms of reasoning:

  - inferring causes from databases

  - database repairs and consistent query answering (CQA)

  - model-based diagnosis

  - Updates trough views in databases

- They all reflect some sort of uncertainty about the information at hand

  They all have related non-monotonic reasoning tasks

- A notion of causality-based explanation for a query result:

(Meliou et al., VLDB 2010)

  - A relational instance D and a boolean conjunctive $\mathcal{Q}$
  - A tuple $t \in D$ is a counterfactual cause for $\mathcal{Q}$ if $D \models \mathcal{Q}$ and $D \smallsetminus \{t\} \not\models \mathcal{Q}$
  - A tuple $t \in D$ is an actual cause for $\mathcal{Q}$ if there is a contingency set $\Gamma \subseteq D$, such that t is a counterfactual cause for $\mathcal{Q}$ in $D \smallsetminus \Gamma$

Based on (Halpern and Pearl, 2001, 2005)

- Responsibility reflects relative degree of causality of a tuple for a query result (Meliou et al., VLDB 2010)

  - The responsibility of an actual cause t for $\mathcal{Q}$:

$$\rho_{D}(t) \ := \ \frac{1}{|\Gamma| \ + \ 1}\ ,\ |\Gamma| = \text{size of smallest contingency set for } t$$

- Tuples with higher responsibility tend to provide more interesting explanations for query results

  Based on (Chockler and Halpern, 2004)

Example: Database D with relations R and S below

$\mathcal{Q}: \exists x \exists y (S(x) \wedge R(x, y) \wedge S(y))$ $\quad D \models \mathcal{Q}$

Causes for $\mathcal{Q}$ being true in D:

R	A	B
	a_4	a_3
	a_2	a_1
	a_3	a_3

S	A
	a_4
	a_2
	a_3

$S(a_3)$ is counterfactual cause for $\mathcal{Q}$: if $S(a_3)$ is removed from D, $\mathcal{Q}$ is no longer an answer; its responsibility is 1

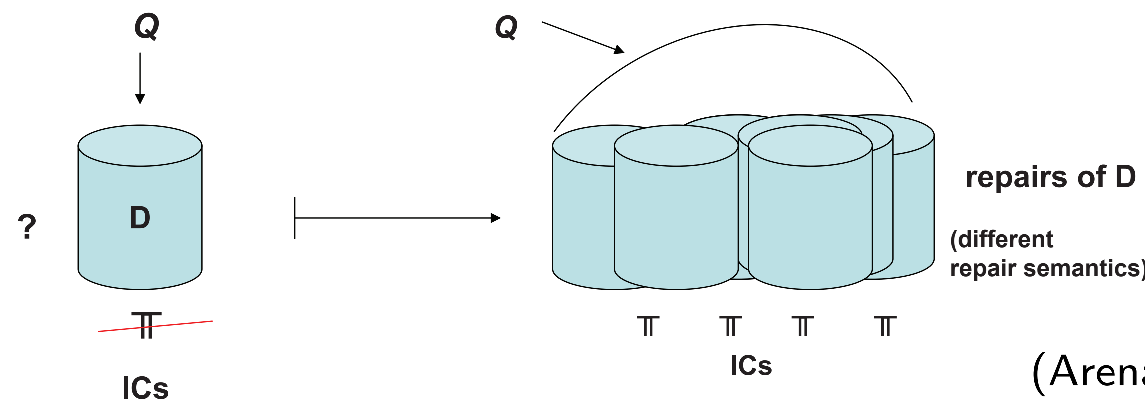
$R(a_4, a_3)$ is an actual cause for $\mathcal{Q}$ with contingency set $\{R(a_3, a_3)\}$

If $R(a_3, a_3)$ is removed from D, $\mathcal{Q}$ is still true, but further removing $R(a_4, a_3)$ makes $\mathcal{Q}$ false

The responsibility of $R(a_4, a_3)$ is $\frac{1}{2}$: its smallest contingency sets have size 1

$R(a_3, a_3)$ and $S(a_4)$ are actual causes, with responsibility $\frac{1}{2}$

Database Repairs



(Arenas et al., PODS 99)

Example: Denial constraints (DC) and inconsistent DBs (also FDs)

$$\neg\exists x\exists y(P(x)\wedge Q(x,y))$$
$$\neg\exists x\exists y(P(x)\wedge R(x,y))$$

P	A
	a
	e

Q	A	B
	a	b

R	A	C
	a	c

Minimal subset-repairs (S-repairs): (maximal consistent sub-instance)

$D_1=\{P(e),Q(a,b),R(a,b)\}$ $\qquad$ $D_2=\{P(e),P(a)\}$

Minimal cardinality-repairs (C-repairs): (maximum consistent sub-instance) D_1

The Repair/Causality Connection

- Assume BCQ: $\mathcal{Q}\colon \exists \bar{x}(P_1(\bar{x}_1) \wedge \cdots \wedge P_m(\bar{x}_m))$

  $\mathcal{Q}$ is (unexpectedly) true in D (or we expected $D \models \neg\mathcal{Q}$)

  What are the causes for $\mathcal{Q}$ to be true?

- We can obtain actual causes and contingency sets from database repairs

  $\neg\mathcal{Q}$ logically equivalent to DC:

$$\kappa(\mathcal{Q})\colon\ \neg\exists \bar{x}(P_1(\bar{x}_1) \wedge \cdots \wedge P_m(\bar{x}_m))$$

  Also written: $\leftarrow P_1(\bar{x}_1), \ldots, P_m(\bar{x}_m)$

- If $\mathcal{Q}$ not expected to hold, we may consider D to be inconsistent wrt. $\kappa(\mathcal{Q})$

- Repairs of D wrt $\kappa(\mathcal{Q})$ may be considered ...

- First the class of *S-repairs* wrt. DCs

  S-repairs are S-maximally consistent subsets of D (no proper subset is a repair)

  $\longrightarrow$ causes and responsibilities (Salimi, Bertossi, ICDT'15)

- Collect the classes of set-differences between D and those S- or C-repairs that do not contain tuple t, and are obtained by removing a subset of D

$$
\begin{aligned}
\mathit{Diff}^{s}(D, \kappa(\mathcal{Q}), t) &= \{D \smallsetminus D' \mid D' \text{ is S-repair}, \ t \in (D \smallsetminus D') \subseteq D\} \\
\mathit{Diff}^{c}(D, \kappa(\mathcal{Q}), t) &= \{D \smallsetminus D' \mid D' \text{ is C-repair}, \ t \in (D \smallsetminus D') \subseteq D\}
\end{aligned}
$$

Proposition: (causality)

$t \in D$ is an actual cause for $\mathcal{Q}$ iff $\mathit{Diff}^{s}(D, \kappa(\mathcal{Q}), t) \neq \emptyset$

Proposition: (responsibility)

(a) $\mathit{Diff}^{s}(D, \kappa(\mathcal{Q}), t) = \emptyset$ iff $\rho(t) = 0$

(b) Otherwise, $\rho(t) = \frac{1}{|s|}$ for a maximum-cardinality $s \in \mathit{Diff}^{s}(D, \kappa(\mathcal{Q}), t)$

Corollary: (most responsible actual cause)

t is a MRAC for $\mathcal{Q}$ iff $\mathit{Diff}^{c}(D, \kappa(\mathcal{Q}), t) \neq \emptyset$

So, maximum responsibility tuples are associated to C-repairs

- Similarly database repairs wrt. sets of DCs can be obtained from causes for unions of BCQs (UBCQs)

- Most computational problems related to repairs, in particular C-repairs, are provably hard (Lopatenko, Bertossi, ICDT'07)

- Techniques and results for the latter ca be leveraged

  This leads to high-complexity results for causality and responsibility

Some Data Complexity Results for Causality

- Causality problem (CP): Computing/deciding actual causes

  (a) In polynomial time for CQs (Meliou et al. 2010)

  (b) It also holds for UCQs

- Responsibility problem: deciding membership of

  $$RDP(\mathcal{Q}) = \{(D,t,v) \mid \; t \in D, v \in \{0\} \cup \{\tfrac{1}{k} \mid k \in \mathbb{N}^+\}, \text{ and } D \models \mathcal{Q} \text{ and } \rho_{_D}(t) > v\}$$

  NP-complete for UCQs (Salimi, Bertossi, ICDT'15)

  In general, responsibility-related computations are hard

Answer Set Programs for Database Repairs

- ASPs can be used to specify, compute and query S- and C-repairs

  Examples for S-repairs follow

  Example: Schema $R(A,B,C,D)$ and FD $A,B \rightarrow C$

  $$\neg \exists x y z_1 z_2 v w (R(x,y,z_1,v) \wedge R(x,y,z_2,w) \wedge z_1 \neq z_2)$$

  Repair program contains the rules: (with global tuple ids)

  $$R'(t_1,x,y,z_1,v,\mathsf{d}) \vee R'(t_2,x,y,z_2,w,\mathsf{d}) \leftarrow R(t_1,x,y,z_1,v), R(t_2,x,y,z_2,w), z_1 \neq z_2$$

  $$R'(t,x,y,z,v,\mathsf{s}) \leftarrow R(t,x,y,z,v),\ \mathit{not}\ R'(t,x,y,z,v,\mathsf{d})$$

- A stable model M of the program determines a repair D' of D:

  $D' := \{P(\bar{c}) \mid P'(t,\bar{c},\mathsf{s}) \in M\}$ (and every repair obtained in this way)

Example: (cont., page 6) $\kappa(\mathcal{Q})$: $\neg \exists x \exists y (S(x) \land R(x,y) \land S(y))$

Repair-ASP contains the facts (with tids):

$R(1, a_4, a_3), R(2, a_2, a_1), R(3, a_3, a_3),\ S(4, a_4),\ S(5, a_2), S(6, a_3)$

Rules:

$$\begin{array}{rcl} S'(t_1, x, \mathsf{d}) \lor R'(t_2, x, y, \mathsf{d}) \lor S'(t_3, y, \mathsf{d}) & \leftarrow & S(t_1, x), R(t_2, x, y), S(t_3, y), \\ S'(t, x, \mathsf{s}) & \leftarrow & S(t, x),\ \mathit{not}\ S'(t, x, \mathsf{d}). \quad \text{etc.} \end{array}$$

Repair D_1 represented by model

$$M_1 = \{R'(1, a_4, a_3, \mathsf{s}), R'(2, a_2, a_1, \mathsf{s}), R'(3, a_3, a_3, \mathsf{s}), S'(4, a_4, \mathsf{s}), S'(5, a_2, \mathsf{s}), S'(6, a_3, \mathsf{d}), \ldots\}$$

- For DCs and FDs repair programs can be made non-disjunctive

  May be non-stratified, as in the case of FDs and DCs with self-joins

  Certain query answering (QA) becomes NP-complete in data

Specifying Causes with Repair-ASPs

- Repair programs (and ASPs) provide right expressive power and complexity for causality-related computations

- Causes for the query (represented by tids) obtained by QA on the repair program Π: for each DB predicate P in a DC

  $\Pi \models_{brave} P(t, \bar{x}_i, \mathsf{d})$ (true in *some* model of Π)

Example: (cont., page 6) Causes? Define:

$$Ans(t) \leftarrow R'(t, x, y, \mathsf{d})$$
$$Ans(t) \leftarrow S'(t, x, \mathsf{d})$$

QA: $\Pi \models_{brave} Ans(t)$?

- For responsibility we need contingency sets associated to causes

  New predicate $CauCon(t,t')$: t is an actual cause, and t' accompanies t as a member of the former's contingency set

  (as captured by repair at hand or the corresponding model)

  For each pair of predicates P_i, P_j in DC $\kappa(\mathcal{Q})$, the rule

  $$CauCon(t,t') \leftarrow P_i'(t,\bar{x}_i,\mathsf{d}),\ P_j'(t',\bar{x}_j,\mathsf{d}),\ t \neq t'$$

  Inequality only when P_i and P_j are the same predicate

  Needed even without FDs or DCs with self-joins

Example: (cont.) Π extended with rules to compute causes with contingency sets

$$CauCon(t,t') \leftarrow S'(t,x,\mathsf{d}), R'(t',u,v,\mathsf{d})$$
$$CauCon(t,t') \leftarrow S'(t,x,\mathsf{d}), S'(t',u,\mathsf{d}), t \neq t'$$
$$CauCon(t,t') \leftarrow R'(t,x,y,\mathsf{d}), S'(t',u,\mathsf{d})$$
$$CauCon(t,t') \leftarrow R'(t,x,y,\mathsf{d}), R'(t',u,v,\mathsf{d}), t \neq t'$$

From model M_2 corresponding to repair D_2: $CauCon(1,3)$ and $CauCon(3,1)$ from repair difference $D \smallsetminus D_2 = \{R(a_4,a_3), R(a_3,a_3)\}$

- Contingency sets computed with extensions of ASP with set- and numerical aggregation, e.g. DLV

$$\begin{aligned}
preCon(t,\{t'\}) &\leftarrow CauCon(t,t') \\
preCon(t,\#union(C,\{t''\})) &\leftarrow CauCon(t,t''), preCon(t,C), not\ \#member(t'',C) \\
Con(t,C) &\leftarrow preCon(t,C), not\, aux(t,C) \qquad \text{(maximal sets)} \\
aux(t,C) &\leftarrow CauCon(t,t'), \#member(t',C)
\end{aligned}$$

Responsibility Computation

- Computation of a cause's responsibility within a model/repair:
(maybe not be overall maximum)

$\rho(t, M) := 1/(1 + |d(t, M)|)$, with $d(t, M) := \{CauCon(t, t') \in M\}$

Computed by:

$$pre\text{-}rho(t, n) \leftarrow \#count\{t' : CauCon(t, t')\} = n$$
$$rho(t, m) \leftarrow m * (pre\text{-}rho(t, m) + 1) = 1$$

- Obtaining the responsibility (a global maximum) means aggregation (optimization) over all models, not within individual models

Off-line computation: $\rho(t) = \max\{\rho(t, M) \mid M \text{ model}\}$

- Not needed: each C-repair gives such a global maximum

- C-repairs can be specified with ASPs with weak constraints

Example: (cont.) Add to the program

$$\Leftarrow R(t,\bar{x}), R'(t,\bar{x},\mathsf{d})$$
$$\Leftarrow S(t,\bar{x}), S'(t,\bar{x},\mathsf{d})$$

Only the models that minimize the number violations of the weak program constraints (here # of deleted tuples) are kept

- Local responsibilities above become global responsibilities

Conclusions

- Several classes of repairs have been investigated in the literature
- There is rather general notion of preferred (or prioritized) repair

  (Chomicki et al., 2012)

  Minimization criterion corresponds to a *priority relationship* $\preceq$ on instances
- Optimization criteria for preferences among models of ASP programs have been considered (Gebser et al., TPLP 2011)
- Can we use prioritized ASPs to compute prioritized repairs and causes?

  It should: our repair/causality connection is quite generic

- What about causality for Datalog queries?

  For Datalog queries, cause computation can be NP-complete

  Through a connection to Datalog abduction

  (Bertossi, Salimi, IJAR 17)

- What is the notion of repair wrt. Datalog constraints?

  E.g. navigational constraints on graph databases

- Can repairs wrt. Datalog constraints be specified by ASPs?

  And causes computed with those programs?

EXTRA SLIDES