Explanation Scores in Data Management

Leopoldo Bertossi[1]

Ontario DB-Day, Dec. 2023

[1] Prof. Emeritus, Carleton U., Ottawa

www.scs.carleton.ca/~bertossi

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?
- $\mathcal{Q}: \ \exists x \exists y (Store(x) \wedge Receives(x, y) \wedge Store(y))$

  The query is true in D: $D \models \mathcal{Q}$

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?
- $\mathcal{Q}:\ \exists x \exists y (Store(x) \land Receives(x, y) \land Store(y))$

  The query is true in D: $D \models \mathcal{Q}$
- What tuples "cause" the query to be true?

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?
- $\mathcal{Q}: \ \exists x \exists y (Store(x) \land Receives(x, y) \land Store(y))$

  The query is true in D: $D \models \mathcal{Q}$
- What tuples "cause" the query to be true?
- How strong are they as causes?

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?
- $\mathcal{Q}: \ \exists x \exists y (Store(x) \wedge Receives(x,y) \wedge Store(y))$

  The query is true in D: $D \models \mathcal{Q}$
- What tuples "cause" the query to be true?
- How strong are they as causes?
- We expect tuples $Receives(s_3, s_3)$ and $Receives(s_4, s_3)$ to be "causes"

Explanations in Databases

Receives	*R*.1	*R*.2
	s_2	s_1
	s_3	s_3
	s_4	s_3

Store	*S*.1
	s_2
	s_3
	s_4

- Query: Are there pairs of official stores in a receiving relationship?
- $\mathcal{Q}: \ \exists x \exists y (Store(x) \wedge Receives(x, y) \wedge Store(y))$

  The query is true in D: $D \models \mathcal{Q}$
- What tuples "cause" the query to be true?
- How strong are they as causes?
- We expect tuples $Receives(s_3, s_3)$ and $Receives(s_4, s_3)$ to be "causes"
- Explanations for a query result ...

- A DB system could provide *explanations*

- A DB system could provide *explanations*
- Explanations come in different forms

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution
- Both based on Counterfactual Interventions

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution
- Both based on Counterfactual Interventions
- Hypothetical changes of values in a causal model to detect other changes: *"What would happen if we change ..."?*

  By so doing identify actual causes

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution
- Both based on Counterfactual Interventions
- Hypothetical changes of values in a causal model to detect other changes: *"What would happen if we change ..."?*

  By so doing identify actual causes
- Do changes of feature values make the label change to "Yes"?

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution
- Both based on Counterfactual Interventions
- Hypothetical changes of values in a causal model to detect other changes: *"What would happen if we change ..."?*

  By so doing identify actual causes
- Do changes of feature values make the label change to "Yes"?
- We have investigated actual causality and responsibility in data management and ML-based classification

A Score-Based Approach: Responsibility

- Causality has been developed in AI for three decades or so
- In particular: Actual Causality
- Also the quantitative notion of Responsibility: a measure of causal contribution
- Both based on Counterfactual Interventions
- Hypothetical changes of values in a causal model to detect other changes: *"What would happen if we change ..."?*

  By so doing identify actual causes
- Do changes of feature values make the label change to "Yes"?
- We have investigated actual causality and responsibility in data management and ML-based classification
- Semantics, computational mechanisms, intrinsic complexity, logic-based specifications, reasoning, etc.

- There are other *local explanation scores*

  Also called “attribution scores”

- There are other *local explanation scores*

  Also called "attribution scores"

- Assign numbers to, e.g., database tuples or features values to capture their causal, or, more generally, explanatory strength

- There are other *local explanation scores*

  Also called "attribution scores"

- Assign numbers to, e.g., database tuples or features values to capture their causal, or, more generally, explanatory strength

- Some of them (in data management or ML)
  - Responsibility
  - The Causal Effect score
  - The Shapley value (as Shap in ML)

Example

- Database D with relations R and S below

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

R	A	B
	a_4	a_3
	a_2	a_1
	a_3	a_3

S	A
	a_4
	a_2
	a_3

Example

- Database D with relations R and S below

$$\mathcal{Q}:\ \exists x \exists y (S(x) \land R(x,y) \land S(y))$$

Here: $D \models \mathcal{Q}$

- Causes for $\mathcal{Q}$ to be 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

Example

- Database D with relations R and S below

$$\mathcal{Q}: \ \exists x \exists y (S(x) \land R(x, y) \land S(y))$$

Here: $D \models \mathcal{Q}$

- Causes for $\mathcal{Q}$ to be true in D?
- $S(a_3)$ is counterfactual cause for $\mathcal{Q}$: If $S(a_3)$ is removed from D, $\mathcal{Q}$ is no longer an answer

R	A	B
	a_4	a_3
	a_2	a_1
	a_3	a_3

S	A
	a_4
	a_2
	a_3

Example

- Database D with relations R and S below

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

- Causes for $\mathcal{Q}$ to be 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: $\frac{1}{1 \ + \ \text{min. \# addit. changes}} \ = \ \frac{1}{1+|\emptyset|} \ = \ 1$

Example

- Database D with relations R and S below

$$\mathcal{Q}:\ \exists x \exists y (S(x) \land R(x,y) \land S(y))$$

Here: $D \models \mathcal{Q}$

- Causes for $\mathcal{Q}$ to be 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: $\frac{1}{1 \,+\, \text{min. \# addit. changes}} \;=\; \frac{1}{1+|\emptyset|} \;=\; 1$

- $R(a_4, a_3)$: actual cause 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

Example

- Database D with relations R and S below

$$\mathcal{Q}: \ \exists x \exists y (S(x) \land R(x,y) \land S(y)) \qquad \text{Here: } \ D \models \mathcal{Q}$$

- Causes for $\mathcal{Q}$ to be 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: $\frac{1}{1 \, + \, \text{min. \# addit. changes}} \ = \ \frac{1}{1+|\emptyset|} \ = \ 1$

- $R(a_4, a_3)$: actual cause 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

- Responsibility of $R(a_4, a_3)$: $= \ \frac{1}{1+1} \ = \ \frac{1}{2}$

Example

- Database D with relations R and S below

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

- Causes for $\mathcal{Q}$ to be 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: $\frac{1}{1\ +\ \text{min. \# addit. changes}} = \frac{1}{1+|\emptyset|} = 1$

- $R(a_4, a_3)$: actual cause 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

- Responsibility of $R(a_4, a_3)$: $= \frac{1}{1+1} = \frac{1}{2}$

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

Coalition Games and the Shapley Value

- Initial motivation: By how much a database tuple contributes to the inconsistency of a DB? To the violation of ICs

Coalition Games and the Shapley Value

- Initial motivation: By how much a database tuple contributes to the inconsistency of a DB? To the violation of ICs
- Similar ideas can be applied to the contribution to query results

Coalition Games and the Shapley Value

- Initial motivation: By how much a database tuple contributes to the inconsistency of a DB? To the violation of ICs
- Similar ideas can be applied to the contribution to query results
- Usually *several tuples together* are necessary to violate an IC or produce a query result

Coalition Games and the Shapley Value

- Initial motivation: By how much a database tuple contributes to the inconsistency of a DB? To the violation of ICs
- Similar ideas can be applied to the contribution to query results
- Usually *several tuples together* are necessary to violate an IC or produce a query result
- Like players in a coalition game, some may contribute more than others

Coalition Games and the Shapley Value

- Initial motivation: By how much a database tuple contributes to the inconsistency of a DB? To the violation of ICs
- Similar ideas can be applied to the contribution to query results
- Usually *several tuples together* are necessary to violate an IC or produce a query result
- Like players in a coalition game, some may contribute more than others
- Apply standard measures used in game theory: the Shapley value of tuple

- Consider a set of players D, and a wealth-distribution (game) function $\mathcal{G}: \mathcal{P}(D) \longrightarrow \mathbb{R}$ ($\mathcal{P}(D)$ the power set of D)

- Consider a set of players D, and a wealth-distribution (game) function $\mathcal{G}: \mathcal{P}(D) \longrightarrow \mathbb{R}$ ($\mathcal{P}(D)$ the power set of D)
- The Shapley value of player p among a set of players D:

$$Shapley(D, \mathcal{G}, p) := \sum_{S \subseteq D \setminus \{p\}} \frac{|S|!(|D| - |S| - 1)!}{|D|!} (\mathcal{G}(S \cup \{p\}) - \mathcal{G}(S))$$

- Consider a set of players D, and a wealth-distribution (game) function $\mathcal{G}: \quad \mathcal{P}(D) \longrightarrow \mathbb{R}$ ($\mathcal{P}(D)$ the power set of D)
- The Shapley value of player p among a set of players D:

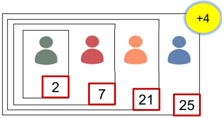
$$Shapley(D, \mathcal{G}, p) := \sum_{S \subseteq D \setminus \{p\}} \frac{|S|!(|D| - |S| - 1)!}{|D|!} (\mathcal{G}(S \cup \{p\}) - \mathcal{G}(S))$$

- $|S|!(|D| - |S| - 1)!$ is number of permutations of D with all players in S coming first, then p, and then all the others

- Consider a set of players D, and a wealth-distribution (game) function $\mathcal{G}: \mathcal{P}(D) \longrightarrow \mathbb{R}$ ($\mathcal{P}(D)$ the power set of D)
- The Shapley value of player p among a set of players D:

$$Shapley(D, \mathcal{G}, p) := \sum_{S \subseteq D \setminus \{p\}} \frac{|S|!(|D| - |S| - 1)!}{|D|!} (\mathcal{G}(S \cup \{p\}) - \mathcal{G}(S))$$

- $|S|!(|D| - |S| - 1)!$ is number of permutations of D with all players in S coming first, then p, and then all the others
- Expected contribution of player p under all possible additions of p to a partial random sequence of players followed by a random sequence of the rest of the players



- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function

- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function

- The Shapley value is a established measure of contribution by players to the wealth function

- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function
- The Shapley value is a established measure of contribution by players to the wealth function
- It emerges as the only measure that enjoys certain desired properties

- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function
- The Shapley value is a established measure of contribution by players to the wealth function
- It emerges as the only measure that enjoys certain desired properties
- For each game one defines an appropriate wealth or game function

- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function
- The Shapley value is a established measure of contribution by players to the wealth function
- It emerges as the only measure that enjoys certain desired properties
- For each game one defines an appropriate wealth or game function
- Shapley difficult to compute: #P-hard in general

- Database tuples and feature values can be seen as players in a coalition game

  Each of them contributing to a shared wealth function

- The Shapley value is a established measure of contribution by players to the wealth function

- It emerges as the only measure that enjoys certain desired properties

- For each game one defines an appropriate wealth or game function

- Shapley difficult to compute: #P-hard in general

- Evidence of difficulty: $\#SAT$ is #P-hard

  About counting satisfying assignments for propositional formulas

  At least as difficult as SAT

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}\colon\ \exists x \exists y (Store(x) \land Receives(x,y) \land Store(y))$

  It takes values 0 or 1 in a database

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}: \ \exists x \exists y(\mathit{Store}(x) \land \mathit{Receives}(x,y) \land \mathit{Store}(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}: \ \exists x \exists y(\mathit{Store}(x) \wedge \mathit{Receives}(x,y) \wedge \mathit{Store}(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}: \ \exists x \exists y (\mathit{Store}(x) \wedge \mathit{Receives}(x, y) \wedge \mathit{Store}(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

$$\mathit{Shapley}(D, \mathcal{Q}, \tau) := \sum_{S \subseteq D \setminus \{\tau\}} \frac{|S|!(|D|-|S|-1)!}{|D|!}(\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}\colon\ \exists x \exists y (\mathit{Store}(x) \wedge \mathit{Receives}(x,y) \wedge \mathit{Store}(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

$$\mathit{Shapley}(D, \mathcal{Q}, \tau) := \sum_{S \subseteq D \setminus \{\tau\}} \frac{|S|!(|D|-|S|-1)!}{|D|!}(\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$

- Quantifies the contribution of tuple τ to query result

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}:\ \exists x \exists y (Store(x) \wedge Receives(x,y) \wedge Store(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

$$Shapley(D, \mathcal{Q}, \tau) := \sum_{S \subseteq D \setminus \{\tau\}} \frac{|S|!(|D|-|S|-1)!}{|D|!}(\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$

- Quantifies the contribution of tuple τ to query result
- All possible permutations of subinstances of D

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}\colon\ \exists x \exists y (Store(x) \wedge Receives(x, y) \wedge Store(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

$$Shapley(D, \mathcal{Q}, \tau) := \sum_{S \subseteq D \setminus \{\tau\}} \frac{|S|!(|D|-|S|-1)!}{|D|!}(\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$

- Quantifies the contribution of tuple τ to query result
- All possible permutations of subinstances of D
- Average of differences between having τ or not

Shapley Values as Scores in DBs

- Database tuples can be seen as players in a coalition game
- Query $\mathcal{Q}\colon\ \exists x \exists y (\mathit{Store}(x) \wedge \mathit{Receives}(x,y) \wedge \mathit{Store}(y))$

  It takes values 0 or 1 in a database
- Game function becomes the value of the query
- A set of tuples make it true or not, with some possibly contributing more than others to making it true

$$\mathit{Shapley}(D, \mathcal{Q}, \tau) := \sum_{S \subseteq D \setminus \{\tau\}} \frac{|S|!(|D|-|S|-1)!}{|D|!}(\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$

- Quantifies the contribution of tuple τ to query result
- All possible permutations of subinstances of D
- Average of differences between having τ or not
- Counterfactuals implicitly involved and aggregated

- We investigated algorithmic, complexity and approximation problems

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard
- Extended to aggregate queries

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard
- Extended to aggregate queries
- It has been applied to measure contribution of tuples to inconsistency of a database

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard
- Extended to aggregate queries
- It has been applied to measure contribution of tuples to inconsistency of a database
- Related and popular score: Banzhaf Power Index (order ignored)

  $Banzhaf(D, \mathcal{Q}, \tau) := \frac{1}{2^{|D|-1}} \cdot \sum_{S \subseteq (D \setminus \{\tau\})} (\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard
- Extended to aggregate queries
- It has been applied to measure contribution of tuples to inconsistency of a database
- Related and popular score: Banzhaf Power Index (order ignored)

  $$\mathit{Banzhaf}(D, \mathcal{Q}, \tau) := \frac{1}{2^{|D|-1}} \cdot \sum_{S \subseteq (D \setminus \{\tau\})} (\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$
- Banzhaf also difficult to compute: #P-hard in general

- We investigated algorithmic, complexity and approximation problems
- A dichotomy theorem for Boolean CQs without self-joins

  Syntactic characterization: : PTIME vs. #P-hard
- Extended to aggregate queries
- It has been applied to measure contribution of tuples to inconsistency of a database
- Related and popular score: Banzhaf Power Index (order ignored)

  $$Banzhaf(D, \mathcal{Q}, \tau) := \frac{1}{2^{|D|-1}} \cdot \sum_{S \subseteq (D \setminus \{\tau\})} (\mathcal{Q}(S \cup \{\tau\}) - \mathcal{Q}(S))$$
- Banzhaf also difficult to compute: #P-hard in general
- We proved "Causal Effect" coincides with the Banzhaf Index!

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

2. Some explanation scores appeal to a probability distribution over the data

   For example, the *Causal-Effect* score

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

2. Some explanation scores appeal to a probability distribution over the data

   For example, the *Causal-Effect* score

   Only the case of the uniform distribution has been investigated

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

2. Some explanation scores appeal to a probability distribution over the data

   For example, the *Causal-Effect* score

   Only the case of the uniform distribution has been investigated

   What about other distributions?

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

2. Some explanation scores appeal to a probability distribution over the data

   For example, the *Causal-Effect* score

   Only the case of the uniform distribution has been investigated

   What about other distributions?

   If we impose or use explicit and additional domain semantics or domain knowledge?

Some Research Directions

1. We have investigated the *Resp* score in the presence of ICs

   How does the Shapley score behave under ICs (or additional metadata)?

2. Some explanation scores appeal to a probability distribution over the data

   For example, the *Causal-Effect* score

   Only the case of the uniform distribution has been investigated

   What about other distributions?

   If we impose or use explicit and additional domain semantics or domain knowledge?

   Can we modify the score's definition and computation accordingly?

   Or the probability distribution?

3. Shapley values satisfy desirable properties for general coalition game theory

3. Shapley values satisfy desirable properties for general coalition game theory

   Existing scores have been criticized or under-explored in terms of general properties

3. Shapley values satisfy desirable properties for general coalition game theory

   Existing scores have been criticized or under-explored in terms of general properties

   Specific general and expected properties for Explanations Scores (in data management)?

3. Shapley values satisfy desirable properties for general coalition game theory

   Existing scores have been criticized or under-explored in terms of general properties

   Specific general and expected properties for Explanations Scores (in data management)?

4. Features (in ML and in general) may be hierarchically ordered according to categorical dimensions

   $$\text{address} \rightarrow \text{neighborhood} \rightarrow city \rightarrow \cdots$$

3. Shapley values satisfy desirable properties for general coalition game theory

   Existing scores have been criticized or under-explored in terms of general properties

   Specific general and expected properties for Explanations Scores (in data management)?

4. Features (in ML and in general) may be hierarchically ordered according to categorical dimensions

   $$\text{address} \rightarrow \text{neighborhood} \rightarrow \textit{city} \rightarrow \cdots$$

   We may want to define and compute explanations (scores) at different levels of abstraction

   How to do this in a systematic way, possibly reusing results at different levels?

   Multi-dimensional explanations?

5. There is a need for principled and sensible algorithms for explanation score aggregation

   At the individual level as in Item 4. or at the group level, e.g. categories of instances

5. There is a need for principled and sensible algorithms for explanation score aggregation

   At the individual level as in Item 4. or at the group level, e.g. categories of instances

   Hopefully guided by a declarative and flexible specifications (about what to aggregate and at which level)

References (some publications for this presentation)

- L. Bertossi, L. and B. Salimi. "From Causes for Database Queries to Repairs and Model-Based Diagnosis and Back". *Theory of Computing Systems*, 2017, 61(1):191-232.

- L. Bertossi and B. Salimi. "Causes for Query Answers from Databases: Datalog Abduction, View-Updates, and Integrity Constraints". *International Journal of Approximate Reasoning*, 2017, 90:226-252.

- L. Bertossi. "Specifying and Computing Causes for Query Answers in Databases via Database Repairs and Repair Programs". *Knowledge and Information Systems*, 2021, 63(1):199-231.

- L. Bertossi. "From Database Repairs to Causality in Databases and Beyond". *Transactions on Large-Scale Data- and Knowledge-Centered Systems LIV* (TLDKS), Springer LNCS 14160, 2023, pp. 119-131.

- E. Livshits, L. Bertossi, B. Kimelfeld and M. Sebag. "The Shapley Value of Tuples in Query Answering". *Logical Methods in Computer Science*, 17(3):22.1-22.33.

- E. Livshits, L. Bertossi, B. Kimelfeld, M. Sebag. "Query Games in Databases". *ACM Sigmod Record*, 2021, 50(1):78-85.

- L. Bertossi, B. Kimelfeld, E. Livshits and M. Monet. "The Shapley Value in Database Management". ACM Sigmod Record, 2023, 52(2):6-17.

- Salimi, B., Bertossi, L., Suciu, D. and Van den Broeck, G. "Quantifying Causal Effects on Query Answering in Databases". Proc. 8th USENIX Workshop on the Theory and Practice of Provenance (TaPP'16).