Technical Perspective for: Query Games in Databases

Dan Suciu
University of Washington

When a data analyst runs some query to analyze her data, she often wants to ask some follow-up questions, about the result of the query. *Why*-questions take many shapes, and occur in many scenarios. Why is a particular tuple in the answer? Why is it *not* in the answer? Why is this graph decreasing? Why did we observe a sudden burst of error messages in online monitoring? Database researchers have noted the need for *why*-questions, and the literature contains several approaches, mostly tailored to specific applications. Despite the interest and the work in this area, there is currently no consensus of what an explanation to a query answer should be, and how one should compute it.

There are two challenges in addressing *why*-questions. The first is to decide the type of the explanation, in other words what to return in response to *why?* The system may return a particular tuple, or a predicate defining a subset of the data, or just an attribute name. This challenge is best addressed with techniques from psychology and HCI, since the choice of the explanation type is ultimately evaluated by end users. The second challenge is to define a quantitative degree of explanation: given the list of all potential explanations, compute a quantitative score, and present them to the user ranked by this score. This challenge requires both a formal definition of the score, and the design of an algorithm to compute that score.

The paper *Query Games in Databases* by Livshits et al. proposes an elegant numerical definition of an explanation, based on a well known concept in economics, called the Shapley value of cooperative games. While originally proposed in economics, the Shapley value has been applied to a variety of domains, such as game theory, political science, and risk analysis. I would also encourage the reader to check out Shapley's original paper from 1952, which is short and remarkably accessible, and available in the survey [33]. Next, Livshits' paper adopts the Shapley value for explaining query answers: the players are the tuples in the database, and the "game" is the query answer. This definition associates a numerical score to every tuple in the database, which represents the tuple's contribution to the observed query output. The paper illustrates with several examples, then focuses on algorithmic aspects of computing this score. As with other definitions of explanation, computing the Shapley value is harder than computing the query answer itself. In fact, the authors prove that it is hard for #P in general, and characterize the queries for which this score can be computed in polynomial time or is #P-hard. They also describe an approximation algorithm, which is of practical interest.

One major attraction of the Shapley value is that it is uniquely defined by three simple axioms. In other words, the Shapley value is the unique scoring function that satisfies three, very natural properties, which Shapley called *symmetry*, *efficiency*, and *aggregation*. Thus, a definition of explanation based on the Shapley value feels principled, in contrast to other definitions of explanation that were mostly inspired by influence analysis or by the responsibility score in causal analysis, and feel more ad-hoc.

Stepping back to look at the bigger picture, the problem studied in this paper is related to *explainable AI*, whose goal is to make automated decisions based on machine learning models transparent to the end users. Researchers have adapted the Shapley value to that setting too: the units of explanation are the features of the ML model, and the goal of the explanation is to associate a score to each feature, quantifying its contribution to the output of the classifier and, like in Livshits' approach, computing the explanation score turns out to be significantly harder than computing the classifier's output. When viewed from this perspective, it becomes clear that Livshits' paper makes a foundational contribution to the general quest of explaining automated decisions.