Letter from the Special Issue Editor

Approximate query processing (AQP) has a long history in databases—for at least 30 years, people have been interested in trading accuracy for better performance, with "better performance" equating to faster response time or lower memory utilization. Over the years, various approximation methodologies have been considered, including sampling, histograms, wavelets, and sketches.

Curiously, those decades of high-quality academic and industrial research on approximation have had surprisingly little commercial impact, at least until recently. While databases are often sampled, direct support for AQP in data management systems is not widespread. There are plenty of explanations for this: one can argue that few people had sufficient data that performance was enough of a concern to accept approximation; end users were suspicious of approximations and statistical methods; database approximation lacked a "killer app" where approximation was absolutely necessary in order to obtain an answer.

There is reason to believe, however, that this lack of adoption of AQP techniques may soon end. The advent of the "Big Data" era and the importance of analytics in general means that very large data sets are ubiquitous. Further, cloud computing means that data processing is now often pay-as-you-go, so less accuracy may mean less cost. Statistics and data science are everywhere. Data mining and machine learning are inherently statistical endeavors—approximation is often vital to scaling such algorithms. If applications such as crowdsourcing become important, approximation is mandatory.

In this issue, we have a slate of very interesting articles on modern applications of AQP as well as modern AQP methodologies. In the issue's introductory paper, "A Handbook for Building an Approximate Query Engine," Mozafari and Niu describe many of the fundamentals that must be understood to build a modern approximate query processing engine. In the second paper, "Scalable Analytics Model Calibration with Online Aggregation," Florin Rusu and his co-authors describe how a now-classical approach to AQP called "online aggregation" can be used to help solve an important problem in large-scale statistical learning: controlling the speed of convergence of gradient descent algorithms. In "On the Complexity of Evaluating Order Queries with the Crowd," Groz, Milo, and Roy consider a case where AQP is mandatory: when crowdsourcing must be used to discover the answer to questions that cannot be answered by simply scanning a database.

In the paper entitled "SampleClean: Fast and Reliable Analytics on Dirty Data," Krishnan and his co-authors describe how AQP can be used to perform in-database analytics when some of the data stored are inaccurate: rather than cleaning all of the data, a small minority can be cleaned and the result of cleaning the rest approximated. In "Independent Range Sampling on a RAM," Hu, Qiao, and Tao investigate the problem of how to actually draw the samples that are needed to power the approximation provided by a sampling-based query: how can we efficiently draw a uniform random sample from a database range query?

In "Hidden Database Research and Analytics (HYDRA) System," Lu and co-authors describe their research in using sampling to answer queries over "hidden databases," or those that are accessible only through a limited user interface (typically on the web), and cannot be examined directly. Finally, in "Approximate Geometric Query Tracking over Distributed Streams," Minos Garofalakis describes how complex queries can be tracked over distributed data streams using the Geometric Method. In such a situation, AQP may be mandatory as it is impossible to move all data to a single location for exact processing.

I hope that you enjoy the issue as much as I enjoyed putting it together!

Chris Jermaine
Rice University