Progressive Data Science: Potential and Challenges

Cagatay Turkay, Nicola Pezzotti, Carsten Binnig, Hendrik Strobelt, Barbara Hammer, Daniel A. Keim, Jean-Daniel Fekete, Themis Palpanas, Yunhai Wang, Florin Rusu

Progressive Data Science: Potential and Challenges

Cagatay Turkay
City, University of London, UK
Cagatay.Turkay@city.ac.uk

Nicola Pezzotti*
TU Delft, NL
N.Pezzotti@tudelft.nl

Carsten Binnig
TU Darmstadt, DE
carsten.binnig@cs.tu-darmstadt.de

Hendrik Strobelt
IBM Research AI, USA
hendrik@strobelt.com

Barbara Hammer
Bielefeld University, DE
bhammer@techfak.uni-bielefeld.de

Daniel A. Keim
University of Konstanz, DE
keim@uni-konstanz.de

Jean-Daniel Fekete
INRIA, FR
Jean-Daniel.Fekete@inria.fr

Themis Palpanas
Paris Descartes University, FR
themis@mi.parisdescartes.fr

Yunhai Wang
Shandong University, CN
cloudseawang@gmail.com

Florin Rusu
University of California
Merced, USA
frusu@ucmerced.edu

ABSTRACT

Data science requires time-consuming iterative manual activities. In particular, activities such as data selection, preprocessing, transformation, and mining, highly depend on iterative trial-and-error processes that could be sped up significantly by providing quick feedback on the impact of changes. The idea of progressive data science is to compute the results of changes in a progressive manner, returning a first approximation of results quickly and allow iterative refinements until converging to a final result. Enabling the user to interact with the intermediate results allows an early detection of erroneous or suboptimal choices, the guided definition of modifications to the pipeline and their quick assessment. In this paper, we discuss the progressiveness challenges arising in different steps of the data science pipeline. We describe how changes in each step of the pipeline impact the subsequent steps and outline why progressive data science will help to make the process more effective. Computing progressive approximations of outcomes resulting from changes creates numerous research challenges, especially if the changes are made in the early steps of the pipeline. We discuss these challenges and outline first steps towards progressiveness, which, we argue, will ultimately help to significantly speed-up the overall data science process.

*Nicola Pezzotti is also affiliated with Phillips Research, Eindhoven, NL

1. INTRODUCTION

Data science is an iterative multi-stage knowledge discovery (KDD) process in which analysts start working with raw, often non-cleaned collections of data sources to derive context-relevant knowledge through the observations made and the computational models built. The overall process involves several labor-intensive trial and error steps within the core activities of data selection, preprocessing, transformation, and mining. This iterative, trial-based nature of the process often means that analysts spend significant amount of time on each stage to move through the analysis pipeline—to give an example, the interviews with enterprise analysts by Kandel et al. [34] report that even preparatory data wrangling steps can easily take more than half of the analysts' time, keeping them off from the rather creative and insightful phases of data analysis. In this paper, we argue how progressive methods, where approximate but progressively improving results are provided to analysts in short time, can transform how the KDD process is currently conducted when progressiveness is introduced within each step of the pipeline, and we introduce *Progressive Data Science* as a novel paradigm.

The underpinning idea of progressive methods is to provide analysts with approximate, yet informative, intermediate responses from a computational mechanism in short time. The analysts

are then supported to interactively investigate these early results and empowered to choose to either discard the chosen conditions due to suboptimal early results, or wait for a full-quality result with the chosen conditions following promising first observations. An illustrative example is an unsupervised clustering process where an analyst is trying to find groups within millions of high-dimensional data observations—a computation that would take considerable amount of time even with an efficient algorithm. To further complicate this, analysts would usually like to investigate several different distance metrics that will give them distinguishable, well-defined groups—a task that can easily become intractable if a single clustering run takes a few hours, if not days. In the progressive setting that we envision, an approximate clustering of the observations is provided as quickly as possible, and, if the initial results fail to provide evidence that any useful structure is captured, that distance function could be discarded immediately, saving the analyst precious time – by not waiting for the full result – and making the time available to try the next alternative distance function.

Furthermore, the progressive approaches we envision do not only help to speed up individual steps of the KDD process, but, more importantly, allow data scientists to quickly revisit previous decisions and immediately see their effects on other steps. For example, in a classical setup, data has to be cleaned (by replacing missing values, removing outliers, etc.) before a data mining algorithm or a machine learning model is applied. In a progressive data science pipeline, we envision that a data scientist can start working on the early steps such as cleaning the data (removing obvious problems) and then move forward to the later steps (e.g., apply the clustering algorithm) already on the partially cleaned data. By looking at the result of clustering the data, new data errors might become visible to the data scientist (e.g., certain types of outliers). Based on these observations, the data scientist could revisit and alter the data cleaning step to remove these outliers and immediately see the effects on the clustering algorithm.

These two examples already showcase the vision for an iterative, high-paced progressive data science process that we argue for. Early promising examples of progressive approaches have been recently introduced in the database [1, 55], machine learning [40, 46, 47, 44], and visualization communities [21, 54, 30]. This paper aims to present a unifying vision through a rethinking of the widely adopted and influential KDD pipeline [36], and introduces Progressive Data Science as a novel knowledge discovery paradigm where progressiveness is inherent in every step of the process.

To present our position, we base the discussion on the individual stages of the KDD pipeline, from data selection, preprocessing, and transformation, to data mining and evaluation, and present how progressiveness can be introduced within each stage and discuss how changes in one stage impact subsequent stages in this progressive setting. In the remainder of this paper, we visit each stage, identify potential opportunities and challenges that lie ahead in integrating progressiveness, and discuss the benefits and implications of this transformation through examples. We then present a number of first promising steps in the database, machine learning, and visualization communities to lead further discussions in this high potential research area.

2. PROGRESSIVENESS CHALLENGES

For each stage in the KDD pipeline [36], we identify opportunities for using progressive methods and present the implications of using progressiveness with respect to their input and output. Informed by and closely following the established KDD pipeline [36], we are rethinking the whole process in a progressive manner as illustrated in Fig. 2. Each section includes concrete examples that provide a clear understanding of the involved challenges.

2.1 Data Selection

Data selection is typically the very first step of a KDD pipeline where users need to explore new data sets and decide whether or not a new data set is relevant for further investigation. In previous work, different approaches have been proposed to increase efficiency for users during the exploration phase [38, 27]. Examples include methods that enable efficient identification of the data subset of interest [39] and fast query execution over raw data sets through online aggregation [25, 10], result reuse [22], or dynamic prefetching [5]. All these techniques aim to quickly provide query results to users to enable efficient selection of interesting data sets. Furthermore, there exist approaches that recommend interesting data sets to the user based on their previous information needs [55, 11]. Another key operation in this stage is the selection of relevant attributes of the data, often referred to as *feature selection* [52]. This phase is of critical importance when the number of attributes in a data set is high and poses challenges for any downstream analysis. During such operations, analysts evaluate the value and importance of features both through

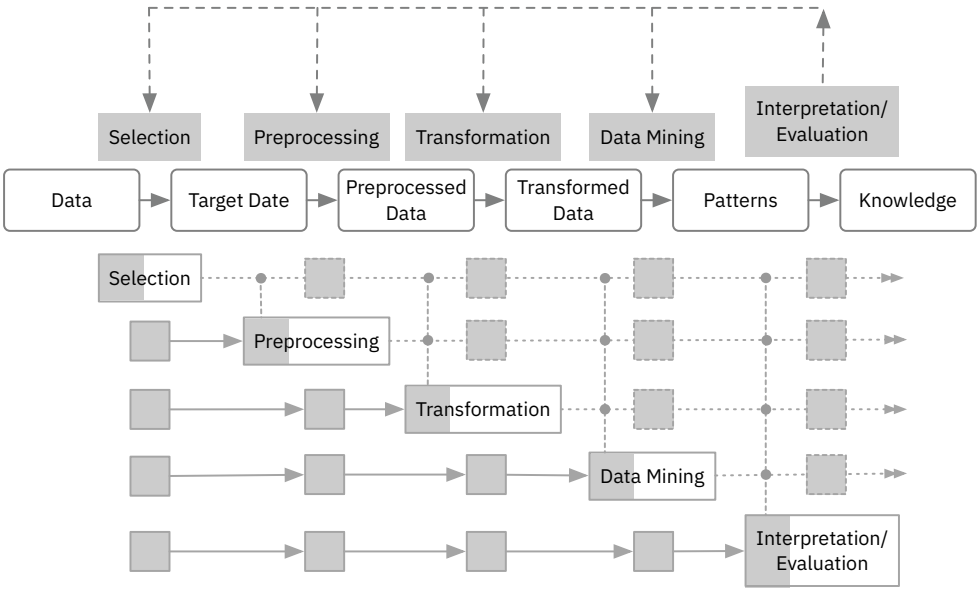

Figure 1: We base our revised Progressive Data Science pipeline on the individual stages of the established KDD pipeline and present how progressiveness can be introduced within each stage. Notice here that each stage operates on and produces data in a progressive manner enabling analysts to effectively move upstream and downstream along the pipeline.

their domain knowledge and through the use of metrics, e.g., variance, entropy, as heuristics.

For progressive data science, it will be interesting to extend data selection in the direction of active techniques that trigger upstream operations such as data cleaning if new relevant data is becoming available. This is similar to the notion of publish-subscribe systems, where users subscribe for certain interesting data items and get notified actively once relevant data is becoming available. For example, in the medical domain, a doctor can register to be informed if entries for patients with a certain disease are being added to the database. Moreover, for progressive data science, upstream operations such as data preprocessing steps (Section 2.2) and model re-training (Section 2.4) could be actively triggered based on such events. In particular, when feature selection is performed progressively, one big potential in downstream analysis is the ability to vary the selection of features, build, for instance, several models and compare performance and utility across these quickly–eventually giving analysts the capability to investigate many analytical hypotheses in short time.

2.2 Data Preprocessing

Data preprocessing in KDD pipeline aims to identify and address quality issues in the data that were selected as interesting. Operations such as the identification of missing values and their imputation, removal of duplicate or problematic records, as well as the identification of outliers are typical for this stage [32]. This is one of those stages where a significant amount of time is spent due to inconsistencies in the way data is gathered or stored.

When conducted in a progressive manner, where, for instance, new data is being made available continuously, some of the key data quality notions might deviate significantly. For instance, with new data being available, new missing value characteristics might emerge, or the scripts that are written to identify and fix data quality issues (e.g., for parsing certain numeric values and for converting them into a unified form) could fail with a dynamically changing representation of such values. The challenges are amplified when models of data are used to fix some of the data quality issues [12]. For instance, where missing values are replaced with the sample average of a feature, or where outliers are flagged based on the distribution characteristics (e.g., those that fall outside the 1.5 times the interquartile range), the varying characteristics of the data over the progressive process pose challenges to address.

Decisions made at this stage often have significant implications for the stages that follow. In

particular, in cases where new "sanitised" data instances are introduced into the data or where problematic records, e.g., outliers, are removed following the process discussed above, any further operation relies on the robustness of these decisions. Erroneous decisions made at this stage could easily bias and skew the models built on the data. One very common example is with missing value imputation and its impact on the data variance [53]. Certain methods can amplify or reduce the co-variation between the data attributes and result in models that pick up on these artificial relations, such as a linear regression model getting stronger if the missing values are imputed following a linear model.

Progressive methods offer effective decision making when applying such critical operations on the data. Where there are several competing strategies to fix data quality issues, analysts can observe the downstream impact of these alternatives and choose those that introduce the least amount of bias. Being able to progressively observe and compare the consequences of a data-level operation on a further modelling stage leads not only to more efficient, but also better-informed decision-making in this stage.

2.3 Data Transformation

In the transformation step of the KDD pipeline, the preprocessed data is modified and reorganized. Ideally, the transformed data becomes better suited as input to the data mining technique that follows, e.g., by removing redundant features or by deriving new ones.

Data transformation techniques heavily depend on the form in which the data is provided as input. If changes are applied in previous steps of the pipeline, the transformation generally holds as long as the number of features and their types are not changed. In a classification setting, for example, if mislabeled data are removed in the preprocessing step, the computations performed here remain unchanged. The scenario is drastically different if features are not removed. Consider, for example, a dataset containing among the features GPS coordinates associated with each data point. As feature transformation, a function that maps coordinates to geographical entities such as regions, states, or nations is defined. If the GPS coordinates are dropped in the data selection phase, the transformation becomes ill-posed. Different strategies can then be adopted to deal with that scenario, such as stopping the progressive computations in the pipeline and informing the user. Another possibility is to ignore the computation of the derived feature and propagate only the ones not affected by the missing input.

Changes in the transformations applied to the data deeply affect the computations performed in later stages of the pipeline, often requiring a change in the data mining algorithms used. Changing the size of the geographical entities in the previous example, e.g, by transitioning from regional to state aggregation, may drastically affect the performance of the data mining or machine learning techniques that follow.

The careful combination of the transformation function and the data mining technique is a cornerstone of the progressive data science pipeline. By directly reflecting the changes applied to the transformation functions, the user can fine-tune model performance by providing better conditioned data.

2.4 Data Mining

The data mining step aims for the inference of a model from a given data set, which can help answering specific questions of the user. Depending on the task, different models can be used, e.g., deep supervised neural networks are suitable for image classification according to given categories, while generative adversarial networks allow the generation of new realistic images. The data mining step typically summarizes three subtasks: selection of a model form and objective function; selection of model and training meta-parameters; and parameter optimization based on the given data (training). A change in the input data can affect all the substeps, and it can do so in unprecedented ways with regard to computational complexity and accuracy of the results: one challenge is raised by an increase in data set size; this usually requires an adaptation of model parameters, which is easily possible for local models such as a kNN (k-nearest-neighbor) classifier, but not so easy for distributed representations such as deep networks. In addition to model parameters, model metaparameters such as the model complexity or number of clusters can also be affected. Notice that classical results from statistical learning theory often guarantee a consistency of the models with increasing data set size, i.e., model adaptation becomes less severe the more data is integrated; yet, these guarantees rely on the often unrealistic assumption of data being independently and identically distributed (i.i.d).

Another challenge occurs if the data representation changes because features are added to or deleted from the data. Feature-centered models, such as decision trees, allow the integration of additional features easily. Alternatives, e.g., deep networks require retraining, a usually time-consuming

process. In all cases, major changes of the model meta-parameters, model architecture, and learning pipeline can occur, and a novel evaluation and interpretation of the results in the subsequent steps of the KDD pipeline becomes necessary.

Progressive technologies can address several challenges in this context: foremost, model inference is often a time consuming process. Progressive techniques can help make advanced data analysis methods accessible in interactive settings where the mere computational complexity renders its classical form infeasible, for instance support vector machines [14] or random forests [9]. In addition, progressive modeling also carries the potential to interactively shape parts of the data mining pipeline which are not easy to formalize, such as the objective function in multi-criteria settings, e.g., where not only classification accuracy but also model complexity and interpretability play an important role.

2.5 Evaluation Functions

The evaluation step aims to determine a measure of quality or performance of the data mining model. This can incorporate external information, such as class labels, human assessment of the model results, or the use of internal information to the model, such as the objective function it optimizes.

When all previous steps are done in a progressive way, the progressive data science pipeline should support human intervention for two different kinds of time-varying information:

- Progression of data mining results, and
- Progression of model evaluation measures

The former provides direct information of the data mining output, while the latter represents meta information of the process evolution. Regarding the evolution of model evaluation measures, the user can make the decision for an early termination once a monotonic curve is produced, indicating the model with low chances of getting better performance. Moreover, changes in the early steps of the pipeline, e.g., in the data transformation step, may also be evaluated here. For example, fine tuning of the feature space may lead to better optimization of the objective function.

One key progressiveness challenge here is to support analysts in making informed judgments on the quality of the results. In order to effectively evaluate a result where indicators of quality are approximate in a progressive setting, effective heuristics that quantify the uncertainty in the results, and the level of convergence (towards a final result) are needed to be estimated and communicated. Potential ad hoc heuristics could be the rate of change in the overall model over iterations [54], or percentage of data processed [48]. However, further research is needed to develop generalizable, systematically evaluated heuristics for uncertainty and convergence of algorithms to serve as effective evaluation criteria in progressive settings.

2.6 Putting it all together

As discussed in all the stages above, progressiveness brings new opportunities when the widely adopted KDD pipeline is reconsidered through this novel lens. In many of the cases above, progressiveness facilitates a fast-paced, flexible, and adaptable analysis process that empowers analysts in dealing with large, heterogeneous, and dynamic data sources, and in generating and evaluating hypotheses and insights. We argue that a primary mechanism that enables this analysis paradigm is the ability to quickly propagate the results of any stage to downstream steps, observe the resulting impact as early as possible, and make changes to the early stage conditions to iterate further. With this very strength comes also the core challenge of progressive approaches—the inherent uncertainty introduced into the pipeline by progressive methods and how this uncertainty can be recognized and considered. Suitable methods are needed to manage the progressive steps in ways where uncertainty at each stage is clearly decoupled and made transparent. Analysts also need methods where they can control and debug the whole pipeline in a seamless manner where they can iterate between the various stages fluidly both upstream and downstream.

3. PROMISING FIRST STEPS

In the following, we discuss existing approaches that are related to our vision from three different communities: database, machine learning, and visualization. As is demonstrated in this section, there is already a huge body of fundamental work existing in these different communities that can be leveraged to enable our vision of progressive data science. However, what is missing is a more holistic view that discusses new progressive approaches that cut through the individual steps of a KDD process and connect those steps to enable data scientists to revisit decisions in all steps and immediately see the effect of changes to all the other steps. One important long-term challenge is, thus, to bring all the existing individual results together in more open and connectable progressive systems that span over the complete KDD process and help data scientists to solve their problems more efficiently.

3.1 Highlights from DB community

The database community has recently been working on aspects to make the individual steps of a KDD process more interactive. One major line of work is centered around query processing and tackles the question of how to enable database engines to provide interactive response times on large data sets. This line of work not only includes approximate query processing techniques [13] that use sampling to achieve interactivity, but also other query processing techniques that aim to re-use previously computed results in a user session (where database queries are potentially built incrementally) [22, 57, 18]. Another line of research has studied the problem of adaptivity, where the system adapts itself (e.g., the data organization, or the index structures), in order to execute queries in an efficient manner [28, 61]. Furthermore, there also exist more advanced speculative query processing techniques [5, 31], which predict what the user is likely to look at next in order to start the computation eagerly. All the before-mentioned interactive query processing techniques are basic approaches that can help to speed up different KDD steps. For example, sampling-based query processing is not only used in the initial data exploration step to help users identify relevant data faster [17], but also for making data mining and model building approaches more efficient [47, 35, 51]. Moreover, there also exist other lines of research in databases not centered around query processing that can be used to make other KDD steps more interactive. One important line is on interactive data cleaning and wrangling [33, 56, 33, 37, 29] to support more efficient extraction of structured data from semi-structured data sets. Another line of work that is important is on recommendation algorithms for data exploration that suggest potentially interesting insights, enabling an easier understanding of large and new data sets [55, 11]. Furthermore, there exist many directions on related areas such as benchmarking interactive database systems [4, 19], but also on making data exploration more safe and avoid that data scientists "tap" into typical statistical pitfalls [7, 24]. An interesting fact that manifests that interactivity and progressiveness play an important role in the database community is the fact that there are multiple workshops co-located with major conferences (e.g., HILDA @ SIGMOD[1], ExploreDB @ SIGMOD[2], and IDEA @ KDD[3]). All these workshops foster new results on the problems related to the above mentioned areas.

[1]http://hilda.io/

[2]https://sites.google.com/a/unitn.it/exploredb18/

[3]http://poloclub.gatech.edu/idea2018/

3.2 Highlights from ML community

Humans' extraordinary mental plasticity enables the seamless life-long learning and efficient incremental adaptation of natural intelligence to novel, non-stationary environments. Yet, one of the major challenges of artificial intelligence remains the question how to efficiently leverage learned strategies to novel environments. Albeit this question is widely unsolved, quite a few promising approaches exist, which carry a high potential as major ingredients of progressive data analytics. Incremental and life-long learning architectures, as an example, address the question how to efficiently adapt data mining models such that they become consistent to novel data, even if the latter might be subject to concept drift [23]. Interestingly, it is possible to set up methods which can efficiently and agnostically deal with a large variety of different types of drift [40]. These machine learning technologies can serve as key ingredients whenever the size of the data set changes in progressive data analysis, with open source tool-boxes for such streaming data analysis being readily available, such as the MOA framework [4] by Bifet et al. [6]. The question on how to deal with changing data representations or tasks is addressed in so-called transfer learning [42]: how can an existing model be transferred to either a different task or a different data representation, thereby preserving relevant common structural principles? Quite a few promising technologies offer interesting ingredients for progressive data analytics pipelines. This includes fast adaptation technologies to transfer a model to a novel probability density function [15] and progressive neural networks for efficiently learning strategies in reinforcement settings [46]. A third example are representation learning technologies which aim for invariant data representations which enable its seamless use for a wide range of different settings [20], whereby universal representations as offered, e.g., by deep networks toolkits, are freely available for important domains such as vision [59].

3.3 Highlights from VIS community

Building progressive visualization and visual analytics systems for data science currently requires complex and expensive developments since existing systems are not designed to be progressive. There has been a few prototypes of progressive visualization and visual analytics systems that proved that the approach was useful and effective for analysts, but they currently remain ad-hoc and monolithic. We will review the most popular ones. Recently,

[4]https://moa.cms.waikato.ac.nz/

there has been some attempts at building infrastructures natively progressive from the ground up. Although the work is still ongoing, more work will be needed to design and implement fully progressive systems at the level of eager ones such as R or Python with their data science stack. While the work is only starting [21, 16], it offers a huge potential for research and opportunities for building scalable interactive data science systems.

The main idea that progressive systems can help human carry long-lasting cognitive tasks has been validated by a study by Zgraggen et al. [60] showing that while human attention is hurt by latencies over 10 seconds, providing progressive results every 1-5 seconds instead of instantaneous results allow analysts to perform exploratory tasks with a similar level of attention. Another experiment by Badam et al. [3] confirmed that analysts can perform complex analyses using a progressive system, understand when to make decisions or when to refrain from making decisions, and interact in a complex way with a progressive system while it is running. However, the experiment was performed using a prototype system where the results of the algorithms were pre-computed to control their latency.

Progressive visualization systems are popular for graph visualization where many graph layout algorithms are iterative. Systems like Tulip [2] and libraries likes D3 [8] implement progressive graph layouts that are popular and allow moving nodes while the algorithm is running to steer the layout. MDSteer [58] provide a similar system for multidimensional scaling, also allowing users to focus on areas of the visualization to steer the computation. However, until recently, progressive visualization was limited to iterative layout algorithms. More recently, several visual analytics applications have been built to deliver progressive results. Stolper et al. [50] coined the term "Progressive Visual Analytic" (PVA), presented the paradigm, explained through an example application "Progressive Insight" meant to mine event sequences to find interesting patterns. The publication has been followed by studies and requirements for PVA [41, 3], by systems performing various kinds of progressive analyses [3, 26], by techniques facilitating progressive computational modelling [54], and by articles describing progressive ML algorithms, such as t-SNE [43, 44], k-nearest neighbors, regression, density estimation [30], and event sequence pattern mining algorithms [50, 49, 45].

4. CONCLUSION

With this paper we introduce *Progressive Data Science* as a new paradigm where analysts are provided with approximate yet informative, intermediate responses from computational mechanisms in short time anywhere within the analysis pipeline. In this approach, letting the analysts interact with the intermediate results allow an early detection of wrong or suboptimal choices, and offer significant improvements within the iterative, traditionally trial-and-error based stages of data science process. In this paper, we presented a unifying vision through a rethinking of the widely adopted and influential KDD pipeline and discussed the various challenges arising from progressiveness followed with a discussion on the promising first steps from different communities where progressive methods are of interest.

We propose *Progressive Data Science* as a novel knowledge discovery paradigm where progressiveness is inherent in every step of the data science process, and ensuring success in such a novel paradigm requires the concerted effort from various research communities. There are several already promising first steps from different communities that demonstrate the potential of the approach, and many interesting scientific challenges lie ahead which require multidisciplinary thinking. We are confident that teams of researchers from complementary domains will address these challenges to further establish this paradigm.

5. ACKNOWLEDGMENTS

The authors would like to thank Schloss Dagstuhl Leibniz Center for Informatics for their support in the organisation of the "Dagstuhl Seminar 18411 - Progressive Data Analysis and Visualization".

6. REFERENCES

[1] Agarwal, S., Mozafari, B., Panda, A., Milner, H., Madden, S., and Stoica, I. BlinkDB: queries with bounded errors and bounded response times on very large data. In *Proceedings of the 8th ACM European Conference on Computer Systems* (New York, NY, USA, 2013), EuroSys '13, ACM, pp. 29–42.

[2] Auber, D., Archambault, D., Bourqui, R., Delest, M., Dubois, J., Lambert, A., Mary, P., Mathiaut, M., Mélançon, G., Pinaud, B., Renoust, B., and Vallet, J. TULIP 5. In *Encyclopedia of Social Network Analysis and Mining*, R. Alhajj and J. Rokne, Eds. Springer, New York, NY, Aug. 2017, pp. 1–28.

[3] Badam, S. K., Elmqvist, N., and Fekete, J.-D. Steering the craft: Ui elements and visualizations for supporting progressive visual analytics. *Computer Graphics Forum 36*, 3 (2017), 491–502.

[4] Battle, L., Angelini, M., Binnig, C., Catarci, T., Eichmann, P., Fekete, J.-D., Santucci, G., Sedlmair, M., and Willett, W. Evaluating visual data analysis systems: A discussion report. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics* (New York, NY, USA, 2018), HILDA'18, ACM, pp. 4:1–4:6.

[5] Battle, L., Chang, R., and Stonebraker, M. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data* (New York, NY, USA, 2016), SIGMOD '16, ACM, pp. 1363–1375.

[6] Bifet, A., Holmes, G., Kirkby, R., and Pfahringer, B. MOA: massive online analysis. *Journal of Machine Learning Research 11* (2010), 1601–1604.

[7] Binnig, C., Stefani, L. D., Kraska, T., Upfal, E., Zgraggen, E., and Zhao, Z. Toward sustainable insights, or why polygamy is bad for you. In *CIDR 2017, 8th Biennial Conference on Innovative Data Systems Research, Chaminade, CA, USA, January 8-11, 2017, Online Proceedings* (Chaminade, CA, USA, 2017), www.cidrdb.org, pp. 56–63.

[8] Bostock, M., Ogievetsky, V., and Heer, J. D^3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (Dec. 2011), 2301–2309.

[9] Breiman, L. Random forests. *Machine learning 45*, 1 (2001), 5–32.

[10] Cheng, Y., Zhao, W., and Rusu, F. Bi-level online aggregation on raw data. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management* (New York, NY, USA, 2017), SSDBM '17, ACM, pp. 10:1–10:12.

[11] Chirigati, F., Doraiswamy, H., Damoulas, T., and Freire, J. Data polygamy: The many-many relationships among urban spatio-temporal data sets. In *Proceedings of the 2016 International Conference on Management of Data* (New York, NY, USA, 2016), SIGMOD '16, ACM, pp. 1011–1025.

[12] Collins, L. M., Schafer, J. L., and Kam, C.-M. A comparison of inclusive and restrictive strategies in modern missing data procedures. *Psychological methods 6*, 4 (2001), 330.

[13] Cormode, G., Garofalakis, M. N., Haas, P. J., and Jermaine, C. Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends in Databases 4*, 1-3 (2012), 1–294.

[14] Cortes, C., and Vapnik, V. Support-vector networks. *Machine learning 20*, 3 (1995), 273–297.

[15] Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence 39*, 9 (2017), 1853–1865.

[16] Crotty, A., Galakatos, A., Zgraggen, E., Binnig, C., and Kraska, T. Vizdom: interactive analytics through pen and touch. *Proceedings of the VLDB Endowment 8*, 12 (2015), 2024–2027.

[17] Crotty, A., Galakatos, A., Zgraggen, E., Binnig, C., and Kraska, T. The case for interactive data exploration accelerators (ideas). In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics* (New York, NY, USA, 2016), HILDA '16, ACM, pp. 11:1–11:6.

[18] Dursun, K., Binnig, C., Cetintemel, U., and Kraska, T. Revisiting reuse in main memory database systems. In *Proceedings of the 2017 ACM International Conference on Management of Data* (New York, NY, USA, 2017), SIGMOD '17, ACM, pp. 1275–1289.

[19] Eichmann, P., Zgraggen, E., Zhao, Z., Binnig, C., and Kraska, T. Towards a benchmark for interactive data exploration. *IEEE Data Eng. Bull. 39*, 4 (2016), 50–61.

[20] Evangelopoulos, G., Voinea, S., Zhang, C., Rosasco, L., and Poggio, T. Learning an invariant speech representation, 2014.

[21] Fekete, J.-D., and Primet, R. Progressive analytics: A computation paradigm for exploratory data analysis, 2016.

[22] Galakatos, A., Crotty, A., Zgraggen, E., Binnig, C., and Kraska, T. Revisiting reuse for approximate query processing. *PVLDB 10*, 10 (2017), 1142–1153.

[23] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. A survey on concept drift adaptation. *ACM computing surveys (CSUR) 46*, 4 (2014), 44.

[24] Guo, Y., Binnig, C., and Kraska, T. What you see is not what you get!: Detecting

simpson's paradoxes during data exploration. In *Proceedings of the 2Nd Workshop on Human-In-the-Loop Data Analytics* (New York, NY, USA, 2017), HILDA'17, ACM, pp. 2:1–2:5.

[25] Hellerstein, J. M., Haas, P. J., and Wang, H. J. Online aggregation. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 1997), SIGMOD '97, ACM, pp. 171–182.

[26] Höllt, T., Pezzotti, N., van Unen, V., Koning, F., Eisemann, E., Lelieveldt, B., and Vilanova, A. Cytosplore: Interactive immune cell phenotyping for large single-cell datasets. *Computer Graphics Forum 35*, 3 (2016), 171–180.

[27] Idreos, S. *Big Data Exploration*. Taylor and Francis, London, UK, 2013, ch. 3, pp. 274–293.

[28] Idreos, S., Manegold, S., and Graefe, G. Adaptive indexing in modern database kernels. In *Proceedings of the 15th International Conference on Extending Database Technology* (New York, NY, USA, 2012), EDBT '12, ACM, pp. 566–569.

[29] Jin, Z., Anderson, M. R., Cafarella, M., and Jagadish, H. V. Foofah: Transforming data by example. In *Proceedings of the 2017 ACM International Conference on Management of Data* (New York, NY, USA, 2017), SIGMOD '17, ACM, pp. 683–698.

[30] Jo, J., Seo, J., and Fekete, J.-D. Panene: A progressive algorithm for indexing and querying approximate k-nearest neighbors. *IEEE Transactions on Visualization and Computer Graphics* (2018), 1–1.

[31] Kamat, N., Jayachandran, P., Tunga, K., and Nandi, A. Distributed and interactive cube exploration. In *2014 IEEE 30th International Conference on Data Engineering* (Chicago, IL, USA, March 2014), IEEE Computer Society, pp. 472–483.

[32] Kandel, S., Heer, J., Plaisant, C., Kennedy, J., van Ham, F., Riche, N. H., Weaver, C., Lee, B., Brodbeck, D., and Buono, P. Research directions in data wrangling: Visualizations and transformations for usable and credible data. *Information Visualization 10*, 4 (2011), 271–288.

[33] Kandel, S., Paepcke, A., Hellerstein, J., and Heer, J. Wrangler: Interactive visual specification of data transformation scripts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2011), CHI '11, ACM, pp. 3363–3372.

[34] Kandel, S., Paepcke, A., Hellerstein, J. M., and Heer, J. Enterprise data analysis and visualization: An interview study. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (Dec. 2012), 2917–2926.

[35] Kraska, T. Northstar: An interactive data science system. *PVLDB 11*, 12 (2018), 2150–2164.

[36] Kriegel, H.-P., and Schubert, M. Kdd pipeline. In *Encyclopedia of Database Systems*, L. Liu and M. T. Özsu, Eds. Springer New York, New York, NY, 2013, pp. 1–2.

[37] Le, V., and Gulwani, S. Flashextract: A framework for data extraction by examples. *SIGPLAN Not. 49*, 6 (June 2014), 542–553.

[38] Lissandrini, M., Mottin, D., Palpanas, T., and Velegrakis, Y. *Data Exploration Using Example-Based Methods*. Morgan & Claypool Publishers, CA, USA, 2018.

[39] Lissandrini, M., Mottin, D., Palpanas, T., and Velegrakis, Y. Multi-example search in rich information graphs. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018* (Paris, France, 2018), IEEE Computer Society, pp. 809–820.

[40] Losing, V., Hammer, B., and Wersing, H. Knn classifier with self adjusting memory for heterogeneous concept drift. In *2016 IEEE 16th International Conference on Data Mining (ICDM)* (Barcelona, Spain, Dec 2016), IEEE, pp. 291–300.

[41] Mühlbacher, T., Piringer, H., Gratzl, S., Sedlmair, M., and Streit, M. Opening the black box: Strategies for increased user involvement in existing algorithm implementations. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1643–1652.

[42] Pan, S. J., Yang, Q., et al. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering 22*, 10 (2010), 1345–1359.

[43] Pezzotti, N., Hllt, T., Lelieveldt, B., Eisemann, E., and Vilanova, A. Hierarchical stochastic neighbor embedding. *Computer Graphics Forum 35*, 3 (2016), 21–30.

[44] Pezzotti, N., Lelieveldt, B. P., van der Maaten, L., Höllt, T., Eisemann, E., and Vilanova, A. Approximated and user

steerable tsne for progressive visual analytics. *IEEE Transactions on Visualization and Computer Graphics 23*, 7 (2017), 1739–1752.

[45] Raveneau, V., Blanchard, J., and Prié, Y. Progressive sequential pattern mining: steerable visual exploration of patterns with PPMT. In *Visualization in Data Science (VDS at IEEE VIS 2018)* (Berlin, Germany, Oct. 2018), IEEE.

[46] Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks, 2016.

[47] Rusu, F., Qin, C., and Torres, M. Scalable analytics model calibration with online aggregation. *IEEE Data Eng. Bull. 38*, 3 (2015), 30–43.

[48] Schulz, H.-J., Angelini, M., Santucci, G., and Schumann, H. An enhanced visualization process model for incremental visualization. *IEEE transactions on visualization and computer graphics 22*, 7 (2016), 1830–1842.

[49] Servan-Schreiber, S., Riondato, M., and Zgraggen, E. Prosecco: Progressive sequence mining with convergence guarantees. In *Proceedings of the IEEE International Conference on Data Mining* (Singapore, 2018), IEEE Computer Society, IEEE.

[50] Stolper, C. D., Perer, A., and Gotz, D. Progressive visual analytics: User-driven visual exploration of in-progress analytics. *IEEE Transactions on Visualization and Computer Graphics 20*, 12 (2014), 1653–1662.

[51] Sun, C., Rampalli, N., Yang, F., and Doan, A. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. *PVLDB 7*, 13 (2014), 1529–1540.

[52] Tang, J., Alelyani, S., and Liu, H. *Feature selection for classification: A review*, 1st ed. Chapman & Hall/CRC, London, UK, 2014, pp. 37–70.

[53] Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., and Altman, R. B. Missing value estimation methods for dna microarrays. *Bioinformatics (Oxford, England) 17*, 6 (Jun 2001), 520–5.

[54] Turkay, C., Kaya, E., Balcisoy, S., and Hauser, H. Designing progressive and interactive analytics processes for high-dimensional data analysis. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 131–140.

[55] Vartak, M., Rahman, S., Madden, S., Parameswaran, A., and Polyzotis, N. See db: efficient data-driven visualization recommendations to support visual analytics. *Proceedings of the VLDB Endowment 8*, 13 (2015), 2182–2193.

[56] Wang, J., Krishnan, S., Franklin, M. J., Goldberg, K., Kraska, T., and Milo, T. A sample-and-clean framework for fast and accurate query processing on dirty data. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data* (New York, NY, USA, 2014), SIGMOD '14, ACM, pp. 469–480.

[57] Wasay, A., Wei, X., Dayan, N., and Idreos, S. Data canopy: Accelerating exploratory statistical analysis. In *Proceedings of the 2017 ACM International Conference on Management of Data* (New York, NY, USA, 2017), SIGMOD '17, ACM, pp. 557–572.

[58] Williams, M., and Munzner, T. Steerable, progressive multidimensional scaling. In *Proceedings of the IEEE Symposium on Information Visualization* (Austin, TX, USA, 2004), IEEE Computer Society, IEEE, pp. 57–64.

[59] Zacharias, J., Barz, M., and Sonntag, D. A survey on deep learning toolkits and libraries for intelligent user interfaces, 2018.

[60] Zgraggen, E., Galakatos, A., Crotty, A., Fekete, J.-D., and Kraska, T. How progressive visualizations affect exploratory analysis. *IEEE Transactions on Visualization and Computer Graphics 23*, 8 (Aug 2017), 1977–1987.

[61] Zoumpatianos, K., Idreos, S., and Palpanas, T. Ads: the adaptive data series index. *The VLDB Journal 25*, 6 (2016), 843–866.