# New Trends in Data Forgetting for Sustainable Data Management

Ramon Rico
Utrecht University
The Netherlands
r.ricocuevas@uu.nl

Arno Siebes
Utrecht University
The Netherlands
a.p.j.m.siebes@uu.nl

Yannis Velegrakis
Univ. of Trento and Utrecht University
The Netherlands
i.velegrakis@uu.nl

## ABSTRACT

Our ability to collect data is rapidly surpassing our ability to store it. As a result, organizations are faced with difficult decisions about what data to retain, and in what form, in order to meet their business goals while complying with storage restrictions. This is typically known as data reduction. This tutorial aims at introducing researchers and practitioners to the topic, and provides a holistic overview of the recent advancement in the field. It covers fundamental principles of data summarization, with a particular emphasis on submodular algorithms, alongside a detailed discussion on the limited existing data forgetting routines. It further underscores the limitations of the data summarization paradigm by introducing the concept of "data rotting" and illustrates the necessity of adopting the new stack data reduction techniques: data forgetting routines. Last, but not least, it discusses the challenges and open research questions in this newly born field.

**PVLDB Reference Format:**
Ramon Rico, Arno Siebes, and Yannis Velegrakis.
New Trends in Data Forgetting for Sustainable Data Management. PVLDB, 18(12): 5472 - 5476, 2025.
doi:10.14778/3750601.3750697

## 1 RELEVANCE, TIMELINESS AND SCOPE

Over the past few years, we have witnessed an extraordinary data-centered revolution in almost every facet of our lives. Massive volumes of data are continuously collected, processed, integrated, and analyzed, resulting in major advancements in fields like logistics, manufacturing, medicine and science. Nonetheless, this revolution encounters significant challenges, mainly due to the vast amount of data generated worldwide, which threatens to surpass our current storage capacity. To date, advances in storage technology have allowed organizations to accumulate data with almost no restriction. However, it is estimated that the size of the global datasphere (i.e., the digital universe) will surpass storage production by an order of magnitude as soon as 2025 [32].

Uncontrolled data storage could compromise the privacy and security of individuals, as recently pointed out by the General Data Protection Regulation (GDPR) [11]. To mitigate this risk, the legislation grants any resident of a protected region the legal authority to require a company to erase their personal data. The consequences of uncontrolled data storage, however, extend beyond user security and privacy concerns. Poor data management practice often results in dirty IT environments where redundant or obsolete data gets accumulated. This not only consumes valuable storage space but also jeopardizes the scalability of data retrieval processes and knowledge discovery algorithms.

To effectively address the aforementioned challenges, data-driven enterprises and research institutes are being forced to face the *data reduction challenge*: retaining the information hidden in the data while respecting regulatory, storage, and processing constraints [25]. Data reduction involves deciding what data to keep and in what form, to comply with legal data regulations and storage restrictions, while minimizing information loss. Storage constraints refer to limitations on the amount of data that any specific institution can store. Additionally, processing constraints pertain to the utilization of data. Meeting processing constraints involves ensuring that data is disposed in a manner that minimally impacts its future expected usage. Furthermore, regulatory constraints simply dictate which data must be retained or deleted within fixed time frames. Such regulations should always be followed regardless of an institution's storage capabilities. Therefore, the algorithmic data reduction task involves finding the portion of data that best meets the processing constraints while fitting into the given storage space.

Due to the vast and ever-growing nature of big data, automating data reduction becomes vital to avoid data-flooding and guaranteeing sustainable data management. Hence, there is an urgent need to develop algorithms based on robust scientific foundations for massive-scale data disposal. Embracing this algorithmic approach to data disposal is crucial for effective knowledge retention and to ensure the sustainability of the data-centered revolution that is reshaping our lives. A common data reduction technique is *data summarization* [16, 17, 21, 23], that aims to reduce big data by replacing it with more compact representations. Another technique is *data forgetting*, that aims to reduce big data by removing data records. The task can be found under different names, but the motivation behind the selection of this characterization will become clear later-on.

This tutorial, aims at providing an overview of the state-of-the-art in data reduction. It examines data summarization methods, highlighting their shortcomings, and illustrates the need for data forgetting, the emerging group of data reduction approaches. There have been tutorials on data summarization in the past [16, 17, 21, 23]. However, since summarization is only a part of data reduction, none of these works addresses the new full stack of data reduction, specifically those under the "data forgetting" umbrella. The current work covers this gap by being broader and going far beyond data summarization. It focuses on data reduction (and specifically on data forgetting) an area that is still in its infancy, aiming to introduce researchers and practitioners to this new paradigm. The presentations so far that are highly related and focused to data

Proceedings of the VLDB Endowment, Vol. 18, No. 12 ISSN 2150-8097.
doi:10.14778/3750601.3750697

reduction are the recent keynotes talks by Tova Milo in VLDB'19 [1], DOLAP'24 [2] and DASFAA'24 [3], alongside our own course in the ACM Summer school on Data Science[4]. The former were focusing on the contributions of the specific research group, while the latter was highly introductory, targeting mainly students and included a hackathon-style lab.

# 2 CORE CONCEPTS

## 2.1 Data Summarization

A popular approach to data reduction is *data summarization.* A *summary* of a dataset $D$ is a brief synopsis that has a modest size compared to that of the latter. Summaries come in one of two flavors: as subsets of $D$, or as representative values that replace the original data records. Further, summarization algorithms fall into one of three categories: statistical [1, 4, 36], submodular [2, 3, 26–29], or geometric [5, 6, 13, 31].

*Statistical summarization* routines rely on statistics to create summaries. Popular approaches like aggregation, creating histograms, or sampling, fall under this category. *Aggregation* [36] involves replacing the original dataset with an assembly of summary statistics. That is, numerical constants that convey information about the central tendency, dispersion or shape of the dataset's distribution (e.g. arithmetic mean, standard deviation, skewness or kurtosis). A more intricate version of this very idea lies behind *histograms* [1], which summarizes a dataset by splitting it along any attribute (or attribute group) into a set of buckets. For each, a small set of summary statistics that approximately represent the data in such bucket is computed. Despite its cost-effectiveness, histogramming and aggregation alone could lead to information loss. In fact, after summarizing a dataset using these techniques, any query that aims at retrieving specific datapoints will return an empty answer when ran against the summary. Nonetheless, when combined with *sampling* [4], powerful procedures have been developed for efficiently computing approximate answers to complex queries over the retained synopsis [30]. Sampling involves selecting a representative subset from a given dataset via a stochastic mechanism. Notable sampling techniques include: simple random, systematic, stratified, and clustering/multistage sampling.

Contrary to statistical methods, *submodular summarization* [2, 3, 26–29] routines approach data reduction as a deterministic subset selection exercise. Given a dataset $D$, a set function $f : \mathcal{P}(D) \mapsto \mathbb{R}$ which measures the amount of representativeness that lies within each subset $D' \in \mathcal{P}(D)$, and a budget $B \in \mathbb{N}$, they seek a subset $D^* \subseteq D$ with at most $B$ elements reaching maximal utility. That is,

$$D^* = \underset{D' \subseteq D,\, |D'| \leq B}{\arg\max} \quad f(D').$$

This optimization exercise is intractable for arbitrary functions. However, set functions that capture desirable features in a summary such as overall diversity and coverage over $D$ like the *facility location* mapping [22] are generally *non-negative, monotone*, and *submodular* [24, 34, 35]. Under said properties, the GREEDY algorithm [29] or its popular accelerated versions LAZY-GREEDY [26] and STOCHASTIC-GREEDY [27], yield a solution with $(1 - 1/e)$ approximation guarantee to the optimal one in polynomial time. Despite producing good approximate solutions, the GREEDY algorithm and its accelerated variants lack scalability. That is, their execution often becomes infeasible when running in data intensive environments. The impracticability of polynomial time algorithms in large scale data settings has motivated the scientific community to explore alternatives like the *distributed* (GreeDi [3, 28]) and *streaming* paradigms (SieveStreaming [2]).

Apart from statistical and submodular approaches, summarization techniques originating from computational geometry have also received attention lately. Among *geometric summaries*, coresets and sketches stand out. A *coreset* [13, 31] is a small representation of a dataset used to perform fast approximate inference with strong theoretical guarantees. Coresets are designed so that the result produced when running mining algorithms on such summaries closely resembles the outcome obtained when ran on the complete dataset. Further, sketching allows summarization of streaming data on the fly. A *sketch* [5, 6] is an easily updatable data structure that gets modified as new instances are received. Notable sketching techniques include: Bloom filters, Count-Min, and HyperLogLog sketching.

## 2.2 Data Forgetting

Despite their different nature, all summarization techniques share a similar philosophy when addressing data reduction: every data point in the input dataset $D$ is assumed to convey a certain degree of valuable information. Since the ultimate goal of summarization is to construct a synopsis that is as representative as possible of the *complete* dataset, every record in $D$ is presumed to be relevant.

*Data rotting* [18, 19] was first to challenge this premise by proposing that data, like everything else in nature, "rots away" losing its value over time. *Data forgetting*, a concept that we introduce, algorithmically formalizes data reduction under the rotting assumption. The inspiration behind our choice of the name "data forgetting" stems from the original vision paper [20]. Kersten and Sidirourgos [20] propose that data management systems should not store records indefinitely but rather have the capability to selectively "forget" entries that have become valueless. The forgetting framework acknowledges that, at any point in time, value is not equally distributed among all regions of a given dataset. That is, upon fixing a notion of *data value / data importance* orthogonal to representativeness, there are some portions in $D$ that contain higher value than others. Data forgetting routines aim to identify the valuable regions within a dataset and "forget" (i.e., delete) everything that lies outside of them. To date, only two existing techniques fall under this paradigm: amnesia algorithms and submodular-based routines.

*Amnesia algorithms* created a paradigm shift in the conception of the role of databases [20]. Amnesia algorithms are online probabilistic rules that provide databases the power to controllably delete data entries over time. Hence, unlike traditional ones, amnesia-scheme equipped databases are not conceived as static objects whose sole purpose is storing data indefinitely, but as dynamic entities able to dispose of data points that will not be useful in the future. The amnesia model is very simple: Data arrives at successive time steps in form of equally-sized batches, and only $B$ data points are held

[1] https://vldb.org/2019/?program-schedule-keynote-speakers
[2] https://dolapworkshop.github.io
[3] https://www.dasfaa2024.org/keynotes/
[4] https://europe.acm.org/seasonal-schools/data-science/2025

inside the database at any point in time. That is, at each time step $t$, $m$ new records arrive and $m$ existing entries in the database are forgotten to meet the storage budget $B$. The way such removal is executed depends on the specific amnesia strategy. In general, the $m$ least *useful* data points are deleted at each step $t$. Nonetheless, usefulness is an extremely domain-dependent concept. Depending on the type of data and case application, it can vary unimaginably. Three notions of data usefulness are proposed in [20], each resulting in a different amnesia scheme: temporal, spatial, and query based. Nevertheless, a significant limitation of these routines lies in the absence of theoretical guarantees

A deterministic alternative to probability-reliant amnesia algorithms are *submodular-based routines*. Submodular-based routines exploit modifications of vanilla submodular algorithms like `GREEDY` (preference cover problem [14]) or its popular accelerated version `LAZY GREEDY` (photo archival problem [7, 8]) to sieve subsets that maximize topology-aware versions of query satisfiability like adaptations of the facility location function. In particular, [7] proposes an effective pre-processing step to speed objective evaluation during the execution of `LAZY GREEDY`. Nevertheless, a noteworthy limitation of such methods resides in the necessity for objective evaluation. Submodular methods become unfeasible when objective evaluation is very expensive. This is certainly the case when maximizing the expected query fulfillment under the facility location mapping.

- (1) **Introduction and Motivation**
    - (a) The need for Data Reduction
    - (b) Data Forgetting: the new Data Reduction paradigm
    - (c) Application of Data Forgetting
- (2) **Redundancy Removal**
    - (a) Data Deduplication [12][10]
    - (b) Data Cleaning [33]
    - (c) Reliability [9]
- (3) **Data Summarization**
    - (a) Statistical Summarization
        - (i) Aggregation [36]
        - (ii) Histogramming [1]
        - (iii) Sampling [4]
    - (b) Geometric Summarization
        - (i) Coresets [13, 31]
        - (ii) Sketches [5, 6]
    - (c) Submodular Summarization
        - (i) Set functions and Submodularity [22]
        - (ii) Summarization objectives [24, 34, 35]
- (4) **Data Forgetting**
    - (a) Data Rotting [18, 19]
    - (b) Amnesia algorithms [20]
- (5) **Submodular Data Forgetting**
    - (a) Forgetting Objectives [7, 8, 14]
    - (b) The `GREEDY` algorithm [29]
    - (c) Accelerated versions of `GREEDY`
        - (i) The `LAZY-GREEDY` [26]
        - (ii) The `STOCHASTIC-GREEDY` [27]
    - (d) Scaled up versions of `GREEDY`
        - (i) The GreeDi [3, 28]
        - (ii) The SieveStreaming [2]
- (6) **Challenges and Remarks**

**Figure 1: Data Reduction Tutorial Outline**

## 3 STRUCTURE AND OUTLINE

The tutorial is structured in six main parts. The first introduces gradually the challenges posed by the ever-growing digital universe within a world of limited storage, motivating the need for data reduction. It motivates the audience by starting from a well-known clip from Martin Kersten [15], where he sets out why we should rethink our approach to data retention, and suggests that users should be more concerned about the duration of data retention or consider distilling it into summarized information.

The second part provides a very brief overview on reducing data through redundancy removal. It shows how data cleaning [33] and data de-duplication [12][10] at the instance and the schema level can be used to improve data quality, increasing in this way the data reliability [9] with less space.

The third part discusses the state-of-the-art on data summarization but under the prism of data reduction. It provides an overview of summarization objectives [24, 34, 35] alongside an introduction to submodular set functions [22]. It presents the most relevant data summarization techniques [1, 4–6, 13, 31, 36], with a particular emphasis on the submodular class [2, 3, 26, 28, 29], and discusses their ability and effectiveness in reducing datasets. It continues with the limitations of the data summarization paradigm and justify the need for transitioning to the novel data reduction paradigm: data forgetting.

The fourth part focuses on the latest developments in data reduction. First, it introduces the concept of data rotting [18, 19]. Following that, it explores a series of techniques that we have grouped together and refer to as *data forgetting* routines. It will highlight and provide in-depth explanations of the main differences between these techniques and traditional summarization methods, explaining why they are better suited to address the needs of modern dataset reduction. It will start with the amnesia algorithms [20].

The fifth part will dive deeper into a special case of data forgetting techniques, the submodular-based [7, 8, 14]. In contrast to amnesia-based routines, submodular-based data forgetting routines enable the forgetting of datasets with strong theoretical guarantees. Nonetheless, a noteworthy limitation of these routines resides in the necessity for extensive function evaluation, which means that in data intensive settings where function evaluation is costly, these algorithms become infeasible.

The last part is dedicated to challenges and open research directions. It discusses questions for further research, like: Can we *rapidly* forget massive data without compromising solution quality? Can we forget massive data *on the fly* with a reasonably good approximation ratio? Can we forget datasets where points have *different* storage costs? Can machine learning help in finding out the best subset of data/metadata to store? Can we easily integrate these techniques into existing data management systems?

## 4 TARGET AUDIENCE

The tutorial is intended for researchers and practitioners interested in topics of big data management, data storage, data reduction, data curation, data cleaning, and data quality. No prior knowledge is required to follow the tutorial, but some familiarity with fundamental database and basic data mining concepts facilitates it. It aims at fostering collaborations between disciplines like data management, data mining, and knowledge discovery. Researchers and students will find interesting ideas and challenges to start research in data reduction, particularly in data forgetting methods. Moreover, they will get an overview of the existing state of the art approaches. Practitioners, on the other hand, will find the tutorial appealing since it will present a new generation data reduction techniques, which can be easily applied on a variety of existing data management systems of structured and non-structured data.

## 5 PRESENTERS

**Ramon Rico** is a PhD candidate of Computer Science, a member of the Data Intensive Systems group and of the AI Lab for Sustainable Finance at Utrecht University. He is also an external AI researcher at the ING group in Amsterdam. He has received his BSc Hons. degree in Mathematics from the Autonomous University of Madrid, and his MSc in Computer Science from Utrecht University. He conducted his master thesis on the forefront of data reduction methods, advancing the current state of the art in data forgetting methods. His scientific interests include submodular optimization methods for big data reduction and self-explainable graph neural networks for financial applications.

**Arno Siebes** is a professor of Computer Science at Utrecht University, where he holds the chair of Algorithmic Data Analysis. He is the head of the Algorithmic Data Analysis group. He has a PhD degree in Computer Science from Twente University (1990). He worked at the Dutch National Research Center for Mathematics and Computer Science (CWI) from 1985 until 2000. He was one of the co-founders of the data mining company “Data Distilleries”, which by way of SPSS became part of IBM. In 1999 he became a part time full professor at the Technical University Eindhoven, since 2000 he is full time chair of Algorithmic Data Analysis and full professor at the department of Information and Computing Sciences of Utrecht University. Since 2017 he serves as the scientific director of the national research school SIKS. His recent research is on mining (sets) of patterns. Results of that line of research have been published in top-tier conferences in the field such as ECML PKDD, ICDM, KDD, and SDM, as well as in top journals such as “Data Mining and Knowledge Discovery”. He also has served as a Tutorials Chair for ICDM08 and co-organized a Dagstuhl Seminar on "Detecting Local Patterns" in 2004.

**Yannis Velegrakis** is a professor of Computer Science at Utrecht University, holding a chair on Very Large Data Management. He is the head the Data Intensive Systems group and the leader of the Master’s in Data Science. His area of expertise includes Big Data Understanding, Knowledge Management, Highly Heterogeneous Information Integration & Data Exchange, Data Curation, and Data Quality. He holds a PhD degree in Computer Science from the University of Toronto. He has been a professor at the University of Trento, a PI with the Archimedes Research Unit of the ATHENA Research Center, a researcher at the AT&T Research Labs, and has also spent time as a research visitor at IBM Almaden Research Center, the Huawei European Research Center in Munich, the University of California, Santa-Cruz, and the University of Paris-Saclay. He has been the general chair for VLDB13 and ICDE24, and the PC Chair for EDBT21. He is currently serving as president of the executive board of EDBT, and as a member of the VLDB Endowment Executive Committee. He has served several times as Associate Editor or Area Chair in SIGMOD, VLDB, EDBT, and ICDE, and has given tutorials in ESCW20, SIGIR19, SIGMOD19, VLDB17, ICDE15, ICDE12, ESWC12, and EDBT11.

## REFERENCES

[1] Mohiuddin Ahmed. 2019. Data summarization: a survey. *Knowledge and Information Systems* 58 (2019), 249–273. https://doi.org/10.1007/s10115-018-1183-0

[2] Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. 2014. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 671–680.

[3] Rafael Barbosa, Alina Ene, Huy Nguyen, and Justin Ward. 2015. The power of randomization: Distributed submodular maximization on massive datasets. In *International Conference on Machine Learning*. PMLR, 1236–1244.

[4] William G. Cochran. 1977. *Sampling Techniques, 3rd Edition.* John Wiley.

[5] Graham Cormode. 2017. Data Sketching. *Commun. ACM* 60, 9 (aug 2017), 48–55. https://doi.org/10.1145/3080008

[6] Graham Cormode, Minos Garofalakis, Peter J Haas, Chris Jermaine, et al. 2011. Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends® in Databases* 4, 1–3 (2011), 1–294.

[7] Susan B. Davidson, Shay Gershtein, Tova Milo, Slava Novgorodov, and May Shoshan. 2022. PHOcus: Efficiently Archiving Photos. *Proc. VLDB Endow.* 15, 12 (sep 2022), 3630–3633. https://doi.org/10.14778/3554821.3554861

[8] Susan B Davidson, Shay Gershtein, Tova Milo, Slava Novgorodov, and May Shoshan. 2023. Efficiently Archiving Photos under Storage Constraints. (2023).

[9] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. 2016. Knowledge-Based Trust: Estimating the Trustworthiness of Web Sources. *IEEE Data Eng. Bull.* 39, 2 (2016), 106–117. http://sites.computer.org/debull/A16june/p106.pdf

[10] Muhammad Ebraheem, Saravanan Thirumuruganathan, Shafiq R. Joty, Mourad Ouzzani, and Nan Tang. 2018. Distributed Representations of Tuples for Entity Resolution. *Proc. VLDB Endow.* 11, 11 (2018), 1454–1467. https://doi.org/10.14778/3236187.3236198

[11] EU. 2016. General Data Protection Regulation (GDPR).

[12] Wenfei Fan. 2012. Data Quality: Theory and Practice. In *Web-Age Information Management - 13th International Conference, WAIM 2012, Harbin, China, August 18-20, 2012. Proceedings (Lecture Notes in Computer Science)*, Hong Gao, Lipyeow Lim, Wei Wang, Chuan Li, and Lei Chen (Eds.), Vol. 7418. Springer, 1–16. https://doi.org/10.1007/978-3-642-32281-5_1

[13] Dan Feldman. 2020. Introduction to core-sets: an updated survey. *arXiv preprint arXiv:2011.09384* (2020).

[14] Shay Gershtein, Tova Milo, and Slava Novgorodov. 2020. Inventory Reduction via Maximal Coverage in E-Commerce.. In *EDBT*. 522–533.

[15] HiPEAC-TV. 2017. ’Data will rot away like everything in nature’ - Martin Kersten. https://www.youtube.com/watch?v=6Jbbwq1FDEY&t=64s. YouTube video, last accessed on July 13, 2025.

[16] Rishabh Iyer, Abir De, Ganesh Ramakrishnan, and Jeff Bilmes. 2022. Subset selection in machine learning: Theory, applications, and hands on. In *Thirty-Sixth Conference on Artificial Intelligence, AAAI-2022 Tutorial Forum.*

[17] Anubhav Jangra, Sourajit Mukherjee, Adam Jatowt, Sriparna Saha, and Mohammad Hasanuzzaman. 2023. A survey on multi-modal summarization. *Comput. Surveys* 55, 13s (2023), 1–36.

[18] Martin Kersten. 2015. Big Data Space Fungus. In *Proceedings of the 7th CIDR.*

[19] Martin Kersten. 2016. Keynote: DataFungi, from Rotting Data to Purified Information.. In *Proceedings of the 32nd ICDE.*

[20] Martin Kersten and Lefteris Sidirourgos. 2017. A Database System with Amnesia. In *CIDR.*

[21] Haridimos Kondylakis, Dimitris Kotzinos, and Ioana Manolescu. 2019. RDF graph summarization: principles, techniques and applications (tutorial). In *EDBT/ICDT 2019-22nd International Conference on Extending Database Technology-Joint Conference.*

[22] Andreas Krause and Daniel Golovin. 2014. Submodular function maximization. *Tractability* 3 (2014), 71–104.

[23] Vivian Lai, Alison Smith-Renner, Ke Zhang, Ruijia Cheng, Wenjuan Zhang, Joel Tetreault, and Alejandro Jaimes. 2022. An exploration of post-editing effectiveness in text summarization. *arXiv preprint arXiv:2206.06383* (2022).

[24] Hui Lin and Jeff A Bilmes. 2012. Learning mixtures of submodular shells with application to document summarization. *arXiv preprint arXiv:1210.4871* (2012).

[25] Tova Milo. 2019. Getting Rid of Data. *J. Data and Information Quality* 12, 1, Article 1 (nov 2019), 7 pages. https://doi.org/10.1145/3326920

[26] Michel Minoux. 1978. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques: Proceedings of the 8th IFIP Conference on Optimization Techniques Würzburg, September 5–9, 1978*. Springer, 234–243.

[27] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. 2015. Lazier than lazy greedy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29.

[28] Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. 2013. Distributed submodular maximization: Identifying representative elements in massive data. *Advances in Neural Information Processing Systems* 26 (2013).

[29] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming* 14 (1978), 265–294.

[30] Laurel Orr, Magdalena Balazinska, and Dan Suciu. 2020. Sample debiasing in the themis open world database system. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 257–268.

[31] Jeff M Phillips. 2017. Coresets and sketches. In *Handbook of discrete and computational geometry*. Chapman and Hall/CRC, 1269–1288.

[32] David Reinsel, John Gantz, and John Rydning. 2018. Data Age 2025 - The digitization of the world from edge to core. *https://www.seagate.com* (2018).

[33] Theodoros Rekatsinas, Xu Chu, Ihab F. Ilyas, and Christopher Ré. 2017. HoloClean: Holistic Data Repairs with Probabilistic Inference. *Proc. VLDB Endow.* 10, 11 (2017), 1190–1201. https://doi.org/10.14778/3137628.3137631

[34] Ian Simon, Noah Snavely, and Steven M Seitz. 2007. Scene summarization for online image collections. In *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 1–8.

[35] Sebastian Tschiatschek, Rishabh K Iyer, Haochen Wei, and Jeff A Bilmes. 2014. Learning Mixtures of Submodular Functions for Image Collection Summarization. In *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (Eds.), Vol. 27. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/file/a8e864d04c95572d1aece099af852d0a-Paper.pdf

[36] Graham J. G. Upton and Patrick Cook. 2002. A Dictionary of Statistics.