

# INforE: Interactive Cross-platform Analytics for Everyone

Nikos Giatrakos  
Athena Research Center  
Technical University of Crete  
[ngiatrakos@athenarc.gr](mailto:ngiatrakos@athenarc.gr)  
[ngiatrakos@softnet.tuc.gr](mailto:ngiatrakos@softnet.tuc.gr)

David Arnu  
RapidMiner GmbH  
[darnu@rapidminer.com](mailto:darnu@rapidminer.com)

Theodoros Bitsakis  
Antonios Deligiannakis  
Minos Garofalakis  
Athena Research Center  
Technical University of Crete  
{[tbitsakis](mailto:tbitsakis@athenarc.gr),[adeli](mailto:adeli@athenarc.gr),[minos](mailto:minos@athenarc.gr)}@athenarc.gr

Ralf Klinkenberg  
RapidMiner GmbH  
[rklinkenberg@rapidminer.com](mailto:rklinkenberg@rapidminer.com)

Aris Konidaris  
Antonios Kontaxakis  
Athena Research Center  
Technical University of Crete  
{[v.b.konidaris](mailto:v.b.konidaris@athenarc.gr),[akontaxakis](mailto:akontaxakis@athenarc.gr)}@athenarc.gr

Yannis Kotidis  
Athens University of Economics &  
Business  
Athena Research Center  
[kotidis@aueb.gr](mailto:kotidis@aueb.gr)

Vasilis Samoladas  
Alkis Simitis  
George Stamatakis  
Athena Research Center  
Technical University of Crete  
{[vsam](mailto:vsam@athenarc.gr),[alkis](mailto:alkis@athenarc.gr),[gstamatakis](mailto:gstamatakis@athenarc.gr)}@athenarc.gr

Fabian Temme  
Mate Torok  
Edwin Yaqub  
RapidMiner GmbH  
{[ftemme](mailto:ftemme@rapidminer.com),[mtorok](mailto:mtorok@rapidminer.com),[eyaqub](mailto:eyaqub@rapidminer.com)}@rapidminer.com

Arnau Montagud  
Miguel Ponce de León  
Barcelona Supercomputing Center  
{[arnau.montagud](mailto:arnau.montagud@bsc.es),[miguel.ponce](mailto:miguel.ponce@bsc.es)}@bsc.es

Holger Arndt  
Stefan Burkard  
Spring Techno GmbH & Co. KG  
{[h.arndt](mailto:h.arndt@springtechno.com),[s.burkard](mailto:s.burkard@springtechno.com)}@springtechno.com

## ABSTRACT

We present INforE, a prototype supporting non-expert programmers in performing optimized, cross-platform, streaming analytics at scale. INforE offers: (a) a new extension to the RapidMiner Studio for graphical design of Big streaming Data workflows, (b) a novel optimizer to instruct the execution of workflows across Big Data platforms and clusters, (c) a synopsis data engine for interactivity at scale via the use of data summaries, (d) a distributed, online data mining and machine learning module. To our knowledge INforE is the first holistic approach in streaming settings. We demonstrate INforE in the fields of life science and financial data analysis.

## CCS CONCEPTS

• **General and reference** → **Cross-computing tools and techniques**; • **Theory of computation** → **Interactive computation**.

This work has received funding from the EU Horizon 2020 research and innovation program INFORE under grant agreement No 825070.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

© 2020 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## KEYWORDS

cross-platform analytics; interactive Big Data analytics; data streams

### ACM Reference Format:

Nikos Giatrakos, David Arnu, Theodoros Bitsakis, Antonios Deligiannakis, Minos Garofalakis, Ralf Klinkenberg, Aris Konidaris, Antonios Kontaxakis, Yannis Kotidis, Vasilis Samoladas, Alkis Simitis, George Stamatakis, Fabian Temme, Mate Torok, Edwin Yaqub, Arnau Montagud, Miguel Ponce de León, Holger Arndt, and Stefan Burkard. 2020. INforE: Interactive Cross-platform Analytics for Everyone. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*, October 19–23, 2020, Virtual Event, Ireland. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

## 1 INTRODUCTION

To exploit and extract value out of Big Data, a growing demand arises for data scientists skilled in making sense of it [11, 17]. For instance, data science problems are now at the center of life sciences [3] or, in the financial sector, Big Data analytics are considered among the most necessary financial skills [17]. In fact, the demand for skilled data scientists in various application domains is expected to by far outstrip the supply [11, 17]. Data analysts, such as life scientists or stock market experts, could satisfy the aforementioned demand. However, they are typically non-expert programmers, lacking the skills to code and optimize Big Data workflows. Besides, Big Data technologies are significantly fragmented. Delivering advanced analytics may engage a variety of Big

Data platforms and tools located at a number of, potentially geo-dispersed, clusters or clouds. For instance, in life sciences, there is a number of projects aiming at federating the analysis of relevant data across data centers<sup>1</sup>. Similarly, in the stock market sector, discovering dependencies or correlations among economies or industries involves global analysis of stock streams initially originating from local market data centers, each running its own Big Data platforms.

Moreover, interactive analytics over massive, high speed data streams are necessary in a wide variety of modern applications. In life sciences, studying the effect of drug combinations on simulated tumors can generate cell state data of 100 GB/min [8], which need to be analyzed online to interactively determine successive drug combinations. In the financial domain, NYSE alone generates several terabytes of data a day for thousands of stocks [10]. Stakeholders need to perform data mining and machine learning tasks in an interactive, online fashion for timely investment decisions.

In this work we present INforE, a prototype offering a complete suite, which effectively responds to the aforementioned challenges. More precisely, our contributions can be summarized as follows:

- INforE enables non-programmer data analysts to set up advanced analytics workflows through a new extension to the RapidMiner Studio<sup>2</sup> for graphical design of Big streaming Data workflows.
- To abstract the details of configuring workflows over various Big Data platforms and geo-dispersed clusters, INforE utilizes a novel optimizer. The optimizer (i) instructs the computer cluster at which each operator of the workflow will be evaluated, (ii) chooses a proper Big Data platform, targeting certain optimization objectives and constraints, for operators with equivalent implementations in different such platforms, (iii) instructs provisioned resources.
- To allow interactivity in analytics at scale, INforE includes a novel Synopses Data Engine (SDE) Component destined to provide various types of scalability, combining the virtues of parallelism and data summarization with accuracy guarantees.
- INforE exposes a distributed and Online Machine Learning and Data Mining (OMLDM) Component that incorporates a powerful arsenal of clustering, classification and regression algorithms.
- We demonstrate how INforE enables non-expert programmers to deliver advanced analytics at scale in the demanding fields of life sciences and in the stock market sector.

Seminal related works [4, 5, 9, 14] are designed towards cross-platform execution of global workflows, but focus on batch, instead of stream, processing and do not provision for interactivity. Thus, to our knowledge INforE is the first system for interactive, cross-platform, streaming analytics.

## 2 ARCHITECTURE

Figure 1 presents INforE’s architecture. INforE supports the code-free creation of optimized, cross-platform, streaming workflows over various clusters each running one of Apache Flink, Spark Structured Streaming or Kafka. Architectural components interact via REST APIs. Parts of workflows that the optimizer assigns to different Big Data platforms and clusters, communicate via Kafka. **Graphical Editor Component.** The Graphical Editor Component enables analysts to easily design workflows without coding. This

is achieved by encapsulating streaming analysis functionality into operators. INforE extends the RapidMiner Studio by a so-called “Nest” operator. The Nest operator is a sub-process operator, i.e., families of operators (from the Synopses Data Engine, Online Machine Learning and Data Mining and the Stream Transformations Component – see Figure 1) can be placed inside it. Having placed a Nest operator in a workflow, the user can double click on it and then encapsulate other operators. The workflow is designed via drag and drop actions and operators can be connected by drawing arrows to define the data flow. Such exemplary workflows are illustrated in Figure 2 and Figure 3, explained in Section 3. The operators of the Graphical Editor are “Logical Operators” designed as an abstraction layer, without defining the actual platform where each operator should be executed. Thus, the user can focus on the analytic setup of the process, without handling the technology specific aspects. It is the INforE optimizer’s responsibility to prescribe the “Physical Operators” that instantiate the logical ones to be deployed in clusters hosting supported Big Data Platforms.

**Stream Transformations Component.** This component includes stream transformations provided by Big Data platforms such as the DataStream API of Flink or the DataFrames/Datasets of Structured Streaming. The Logical Operators of this component are simply boxes, but the equivalent Physical Operators correspond to the functionality provided by a supported platform. For instance, a join Logical Operator may be interpreted by the optimizer to a Physical join Operator in Spark or Flink.

**Synopsis Data Engine Component (SDE).** INforE manages to provide interactive analytics at scale via the SDE Component [12]. This is due to the fact that, upon included in a designed workflow, the SDE Component can provide 3 types of scalability: (i) enhanced horizontal scalability, i.e., not only scaling out the computation to a number of machines, as Big Data platforms typically do, but also harnessing the load assigned to each by operating on carefully-crafted data summaries, (ii) vertical scalability, i.e., scaling the computation to very high numbers of processed streams by using synopses, such as Fourier Transforms [12], to enable clever data partitioning and (iii) federated scalability i.e., scaling the computation beyond single clusters by controlling the communication required to answer global queries posed over a number of potentially geo-dispersed clusters. Our SDE supports a variety of algorithms which include, but are not limited to [7]: cardinality (HLL Sketches), frequency moment (CountMin, AMS Sketches, Sampling), correlation (Fourier Transforms, Locality Sensitive Hashing), set membership (Bloom Filters) or quantile (GK Quantile) estimation. Past approaches, such as DataSketches<sup>3</sup> or Stream-lib<sup>4</sup> neglect parallelization aspects. Others, like synopsis utilities in Spark, do not account for vertical scalability. Moreover, the SDE Component is implemented following a Synopsis-as-a-Service paradigm (SDEaaS). SDEaaS enables the simultaneous maintenance of thousands of synopses for thousands of streams and allows commonly used synopses to be shared among workflows. Please see [12] for more details on the SDE.

**Online Machine Learning & Data Mining (OMLDM) Component.** OMLDM applies the state-of-the-art in distributed, online machine learning adopting a Parameter Server (PS) paradigm [15]

<sup>1</sup><https://elixir-europe.org/about-us>, <https://commonfund.nih.gov/bd2k/>, <https://cyverse.org/>

<sup>2</sup><https://rapidminer.com/>

<sup>3</sup><https://datasketches.github.io/>

<sup>4</sup><https://github.com/addthis/stream-lib>

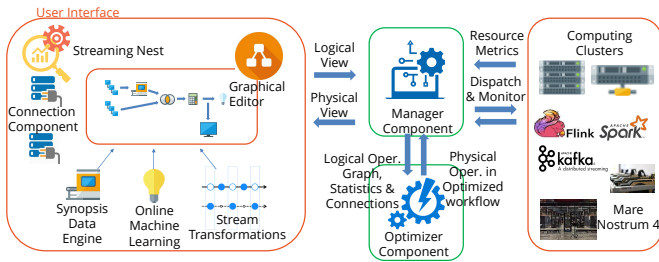


Figure 1: INforE Architectural Components – Big Picture.

for incremental training of models, at the very same time up-to-date models are deployed for inference purposes. Other frameworks, such as Apache SAMOA, lack a PS paradigm or APIs such as MLlib in Spark and FlinkML are mainly focused on batch, instead of stream, processing. We currently support [2, 16] (i) classification: Passive Aggressive, Multiclass Passive Aggressive Classifiers, Online Support Vector Machines and Hoeffding Trees, (ii) clustering: BIRCH, Online k-Means and StreamKM++ and (iii) regression: Passive Aggressive Regressor, Online Ridge Regression, Polynomial Regression algorithms. Preprocessing facilities such as standardization, polynomial feature extraction are also available.

**Optimizer Component.** The Optimizer (middle, bottom of Figure 1) significantly extends our algorithms in [6] to multiple optimization objectives and constraints. It returns a mapping of the logical view (input workflow graph of Logical Operators) provided by the user, to the network of computer clusters and the platforms they host (output workflow graph of Physical Operators and provisioned resources). Our optimization algorithms seek to solve a multi-criteria optimization problem with objectives involving throughput, CPU/GPU/memory usage, communication cost, latency and accuracy (for SDE operators). Due to the conflicting objectives, we resort to Pareto optimal solutions [6]. Good Logical to Physical Operator mappings are indicated by maximizing a weighted combination of the objectives (each objective is assigned a weight and absolute weight values sum up to 1) under resource constraints.

For running jobs, the Optimizer Component receives statistics by the Manager Component (Figure 1) about metrics related to the optimization objectives and constraints. Then, we use a Bayesian Optimization approach, inspired by CherryPick [1], to build performance models for operators, workflow parts and entire workflows. These are used as cost functions to guide the decisions of the optimization algorithm in choosing Logical to Physical Operator mappings. For each supported operator, we have a number of micro-benchmarks, besides what is monitored at runtime. In case a workflow is seen for the first time, we have cost models for operators or for parts of the workflow, but not for the whole workflow. Thus, we synthesize the available costs to model its expected performance. To give a simple example, if we only want to maximize the throughput of a workflow with a previously unseen combination of operators, the optimizer chooses Physical Operators so that the minimum throughput of a Physical Operator in the workflow is maximized. The optimizer uses the ELK stack<sup>5</sup> to ingest statistics and update its models while monitoring jobs.

<sup>5</sup><https://www.elastic.co/what-is/elk-stack>

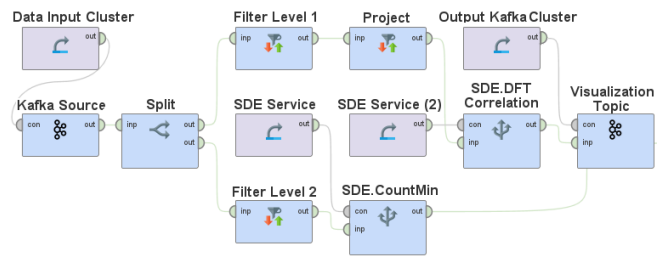



Figure 2: Stock Market Workflow in a Nest Operator.

**Manager Component.** The Manager Component in Figure 1 acts as a middleware between the Graphical Editor, the Optimizer and the computing clusters. When a user presses the submit button in a graphical workflow: (*Step 1*) the workflow and its Logical Operators are interpreted into a JSON file that is fed to the Manager Component, (*Step 2*) the Manager delivers this JSON to the Optimizer which runs its optimization algorithms and returns a new JSON prescribing the Physical Operators of the workflows, (*Step 3*) the Manager Component visualizes the physical view of the workflow to the user via the Graphical Editor, (*Step 4*) a dispatcher in the Manager Component compiles *.jar* files for the optimized workflow to be submitted to the prescribed clusters hosting Big Data platforms. Recall that for parts of the workflow submitted at different clusters, communication of upstream and downstream operators is done via Kafka, (*Step 5*) the Manager Component monitors the execution of the workflow and collects statistics using JMX<sup>6</sup> technology.

**Connection Component.** The Connection Component (on the left of Figure 1) graphically configures access to clusters hosting Big Data platforms (on the right of Figure 1). The Connection Component is also used to graphically specify input and output streams. In Figure 2 and Figure 3, connection objects for input, output streams and services are declared via the  iconed boxes.

### 3 DEMO SPECIFICATIONS

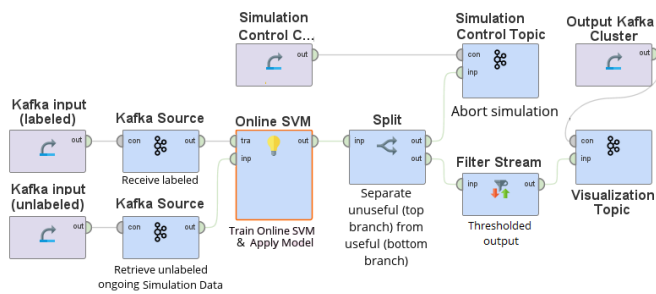
We demonstrate how INforE aids in delivering code-free, advanced analytics in real scenarios and datasets from the life sciences and the stock market domain. Here, we detail exemplary workflows which show the utility of operators from all Stream Transformation, SDE and OMLDM Components in such applications. Users can draw own workflows via the Graphical Editor and press a submit button. INforE will then handle the rest of the execution details, also allowing the user to inspect optimized execution internals.

**Life Science Scenario.** We use INforE to provide a virtual laboratory for simulating tumor behavior under various drug combinations. We have simulated tumor data<sup>7</sup> produced by instances of the PhysiBoss framework [13] ran at the MareNostrum 4<sup>8</sup> super-computer. The scenario involves 3 clusters each hosting a Big Data platform. INforE ingests from PhysiBoss data related to the state of each cell agent, the concentration of various densities such as oxygen and time series data on the number of necrotic, apoptotic

<sup>6</sup><https://docs.oracle.com/javase/tutorial/jmx/overview/>

<sup>7</sup><http://doi.org/10.5281/zenodo.3922263>, <http://doi.org/10.5281/zenodo.3921049>, <http://doi.org/10.5281/zenodo.3923070>

<sup>8</sup><https://www.bsc.es/marenostrum/marenostrum>



**Figure 3: Life Science Analytics Workflow in a Nest Operator.**

or proliferating cells. In this model, effective drugs’ activity forces tumor cells into necrosis and fewer to apoptosis, as detailed in [13].

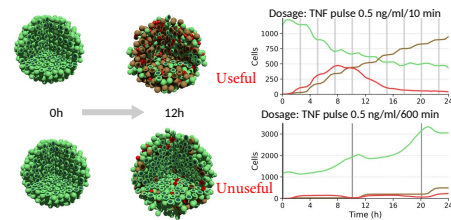
In Figure 3, our target is to distinguish which simulations and, thus, respective drug combinations are effective in terms of increasing the number of necrotic and apoptotic cells, simultaneously reducing the proliferating ones. For that purpose, data from the various PhysiBoss instances are loaded via two Kafka topics for unlabeled and labeled simulations. Then, they are fed to an online Support Vector Machine (Online SVM) operator from the OMLDM Component to train models for distinguishing “useful” drug combinations from “unuseful” ones and simultaneously classify running simulations by applying the trained model. Training and unlabeled data enter different ports of Online SVM. The Split operator separates unuseful simulations from useful ones. Those with unuseful outcomes correspond to PhysiBoss instances that should be killed (Figure 3, top split branch). The most promising, in terms of alive vs killed tumor cells, of the rest are chosen for visualization and further study (Figure 3, bottom split branch). INforE integrates <http://www.povray.org/> in Figure 4 to visualize the tumor shape and evolution as well as time series for necrotic (brown), apoptotic (red) and proliferating (green) cells upon applying TNF inhibitor drug in different intervals–pulses.

**Stock Market Scenario.** We use real stock data provided by Spring Techno GmbH & Co. KG, from 9 markets<sup>9</sup>. The workflow of Figure 2 utilizes Level 1<sup>10</sup> and Level 2<sup>11</sup> stock data to discover cross-correlations of stocks and the leader stock, i.e., the one that accumulates the highest number of bids in the correlated pair, for each discovered correlation. In Figure 2, data arrive at a Kafka source. The Split operator separates Level 1 from Level 2 data. It directs Level 2 data to the bottom branch of the workflow. There, the Level 2 bids are Filtered (i.e., for monitoring only a subset of stocks or keep only bids above a price/volume threshold). The bids per stock are counted using a CountMin sketch [12] (SDE.CountMin) provided by the SDE Component. When a trade for a stock is realized, a new Level 1 tuple is directed by Split to the upper part of the workflow. A Project operator keeps only the fields of the incoming tuples on which correlations will be calculated. Pairwise similarities of stocks’ time series are computed using the approximations of the Discrete Fourier Transform (SDE.DFT Correlation) operator [12] of the SDE. This operator accepts as parameters a threshold for outputting

<sup>9</sup><http://doi.org/10.5281/zenodo.3886895>

<sup>10</sup><https://www.investopedia.com/terms/l/level1.asp>

<sup>11</sup><https://www.investopedia.com/terms/l/level2.asp>



**Figure 4: Life Science Scenario – Visualization of the volume and number of necrotic (brown), apoptotic (red) and proliferating (green) cells due to TNF application on a spheroidal tumor.**

only the highly correlated pairs of stocks, a time window for taking into account only the most recent trades upon judging correlations and the number of DFT coefficients that will be used on reduced size vectors compared to those in the original window [12]. Cross-correlations and leader stocks are visualized via <https://d3js.org/> tool integration. For similar purposes, a StreamKM++ operator can be used to extract clusters of stocks, while identifying leaders.

In both scenarios, Split, Project and Filter come from the Stream Transformations Component.

## REFERENCES

- [1] O. Alipourfard, H. Liu, J. Chen, S. Venkataraman, M. Yu, and M. Zhang. 2017. CherryPick: Adaptively Unearthing the Best Cloud Configurations for Big Data Analytics. In *NSDI*.
- [2] A. Benczur, L. Kocsis, and R. Pálovics. 2018. Online machine learning in big data streams. *arXiv:1802.05872* (2018).
- [3] J. Cumbers. 2019. How The Cloud Can Solve Life Science’s Big Data Problem. <https://www.forbes.com/sites/johncumbers/2019/12/19/how-the-cloud-can-solve-life-sciences-big-data-problem/>. [Online; accessed 25-Aug-2020].
- [4] K. Doka, N. Papailiou, D. Tsoumakos, C. Mantas, and N. Koziris. 2015. IREs: Intelligent, Multi-Engine Resource Scheduler for Big Data Analytics Workflows. In *SIGMOD*.
- [5] A. J. Elmore, J. Duggan, M. Stonebraker, and et al. 2015. A Demonstration of the BigDAWG Polystore System. *Proc. VLDB Endow.* 8, 12 (2015).
- [6] I. Flouris, N. Giatrakos, A. Deligiannakis, and M. Garofalakis. 2020. Network-wide complex event processing over geographically distributed data sources. *Inf. Syst.* 88 (2020).
- [7] M. Garofalakis, J. Gehrke, and R. Rastogi. 2016. Data Stream Management: A Brave New World. In *Data Stream Management - Processing High-Speed Data Streams*. Springer.
- [8] N. Giatrakos, N. Katzouris, A. Deligiannakis, and et al. 2019. Interactive Extreme-Scale Analytics Towards Battling Cancer. *IEEE Technol. Soc. Mag.* 38, 2 (2019).
- [9] I. Gog, M. Schwarzkopf, N. Crooks, M. P. Grosvenor, A. Clement, and S. Hand. 2015. Musketeer: all for one, one for all in data processing systems. In *EuroSys*.
- [10] T. Groenfeldt. 2013. At NYSE, The Data Deluge Overwhelms Traditional Databases. <https://www.forbes.com/sites/tomgroenfeldt/2013/02/14/at-nyse-the-data-deluge-overwhelms-traditional-databases/>. [Online; accessed 25-Aug-2020].
- [11] J. Heer and S. Kandel. 2012. Interactive Analysis of Big Data. *ACM Crossroads* 19, 1 (2012).
- [12] A. Kontaxakis, N. Giatrakos, and A. Deligiannakis. 2020. A Synopses Data Engine for Interactive Extreme-Scale Analytics. In *CIKM*.
- [13] G. Letort, A. Montagud, G. Stoll, R. Heiland, E. Barillot, P. Macklin, A. Zinovyev, and L. Calzone. 2019. PhysiBoSS: a multi-scale agent-based modelling framework integrating physical dimension and cell signalling. *Bioinform.* 35, 7 (2019).
- [14] J. Lucas, Y. Idris, B. C. Rojas, J. A. Q. Ruiz, and S. Chawla. 2018. RheemStudio: Cross-Platform Data Analytics Made Easy. In *ICDE*.
- [15] M.Li, D. Andersen, J. W. Park, and et al. 2014. Scaling Distributed Machine Learning with the Parameter Server. In *OSDI*.
- [16] J. Silva, E. Faria, R. Barros, E. Hruschka, A. Carvalho, and J. Gama. 2013. Data Stream Clustering: A Survey. *ACM Comput. Surv.* 46, 1 (2013).
- [17] V. Zhang. 2019. The Rise of the Financial Data Scientist\*. <https://www.nasdaq.com/articles/the-rise-of-the-financial-data-scientist-2019-09-27>. [Online; accessed 25-Aug-2020].