Enhancing the friendliness of data analytics tasks: an automated methodology

Enhancing the friendliness of data analytics tasks: an automated methodology

Paolo Bethaz
Department of Control and Computer Engineering,
Politecnico di Torino, Turin, Italy
paolo.bethaz@polito.it

Tania Cerquitelli
Department of Control and Computer Engineering,
Politecnico di Torino, Turin, Italy
tania.cerquitelli@polito.it

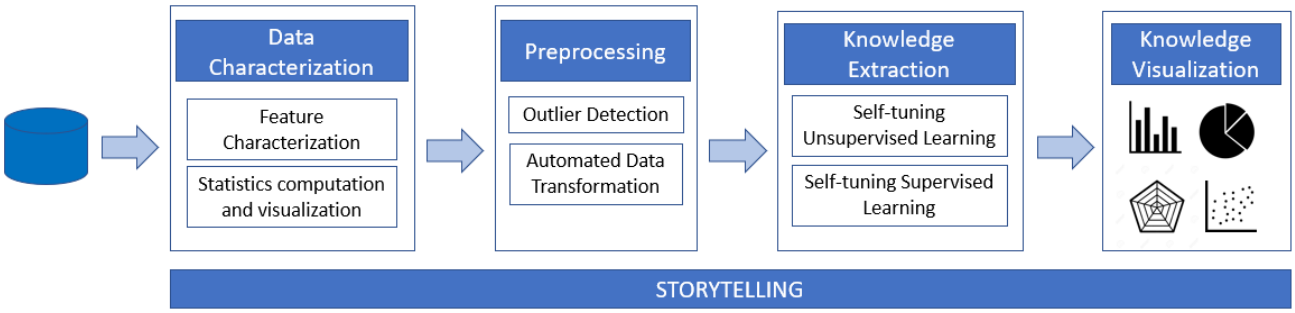

Figure 1: ***Automated Data Exploration Workflow***

ABSTRACT

This paper presents ADESCA (Automated Data Exploration with Storytelling CApability), a new methodology to automatically extract models and structures from the data, with the aim of democratizing data science. Particular importance was placed on the creation of an innovative storyboard for automated data analytics, allowing creation and visualization of data stories. In addition, ADESCA also offers a guided and more traditional step-by-step exploration, where the algorithm parameters are configured automatically and the results are graphically shown through different plots in order to be as understandable and user-friendly as possible. Preliminary experimental results are promising and demonstrated the effectiveness and efficiency of the proposed automated methodology.

1 INTRODUCTION

In today's world there are an infinite number of data that can be analyzed and extracting useful insights from these data has become a fundamental operation in many sectors. However, the exploration of these data requires the intervention of a data scientist with great expertise in the field who often devotes a long time to find a good combination of data-driven algorithms able to discover interesting knowledge items from data. The purpose of our research activity is democratizing the data science by designing and developing a new data-analytics engine providing self-learning capabilities jointly with parameter autoselection to off-load the data scientist from testing and tuning and to easily focus her/his attention only to the most interesting and valuable insights hidden in her/his data.

Some research efforts have already been done in order to improve the data exploration process towards automated approaches. Authors in [14] re-examined the architecture of database management systems, like in DICE system, to combine sampling with speculative execution in the same engine [15]. Different than the above approaches, we did not focus on the architecture, but on the methodological approach, like the authors in [17, 32]. Specifically, in [32] authors proposed Helix, a declarative machine learning system that seeks to optimize end-to-end workflow execution across iterations, while in [17] the READ engine has been discussed to reduce the time required for data exploration.

Regarding the concept of democratizing the data science, works described in [9] and [2] are of considerable importance, as the authors propose to assist "data enthusiastic" users [12] [21] in the exploration of transactional databases in an interactive way. However, these research activities focus on query-oriented techniques, while our purpose is to enhance the friendliness of the knowledge-extraction approach.

There are also a series of tools that try to make the data exploration a process automatic and more graphically intuitive. However, tools like DataWrapper[1], Datapine[2] or ZenVisage [26] only offer a traditional visualization service, where the user can see the distribution of each attribute, after choosing which type of chart to use, among those proposed.

Other tools, in addition to the graphic display of the attributes, also make it possible to perform some analysis on the data. This is for example the case of AnswerMiner[3] (which allows to build decision trees on data), Dive [13] (which allows for example to perform the ANOVA test) and NCSS[4] (which allows to perform ANOVA test, multivariate analysis and a very simplified cluster analysis).

Taking a cue from the works listed so far, in this paper we propose an exploratory data methodology, named ADESCA (Automated Data Exploration with Storytelling CApability), that is able to automatically extract models and structures from the data, without the intervention of a data scientist. In particular, we now propose a new web-based system that automatically combines state-of-the-art data exploration approaches into a single tool to automatically address all the steps of the traditional knowledge

[1] https://github.com/datawrapper/datawrapper
[2] https://www.datapine.com/
[3] https://www.answerminer.com/
[4] https://www.ncss.com/

data discovery pipeline, adding some new innovative steps compared to the traditional approaches. Innovative steps include: (i) the *automated data transformation* strategy, (ii) the *self-tuning* strategies tailored to each data analytics algorithm, and (iii) the *data storytelling* methodology. The results of each of these techniques are shown to the user in a very intuitive way thanks to user-friendly graphs that effectively support the decision-making process.

The automated data transformation section modifies the structure of the original dataset, reporting the data in a different way, to try to extract information that are too hidden or absent in the original structure. Then, the self-tuning strategies applied to supervised and unsupervised techniques off-load the data scientist to manually select the best algorithm and its input-parameter setting. Finally, the data-driven storytelling is a particular section in the implemented tool in which the most important information and results regarding all the analysis done on the data are shown in a completely automatic way, thanks to an effective display system based on multiple web pages, each containing the visual results of different analyzes.

The paper is organized as follows. Section 2 details the proposed automated methodology, introducing the various innovative techniques integrated in the system. Section 3 presents the preliminary development details, while Section 4 discusses the preliminary experimental results obtained on two real datasets. Finally, Section 5 discusses open issues and presents the future development of this work.

2 METHODOLOGY

The ADESCA methodology has the main purpose of creating automated data-stories, capable of summarizing the most important information of a dataset; but, at the same time, ADESCA also allows a step-by-step exploration of the knowledge data discovery (KDD) pipeline shown in Figure 1. In this last case, the user explores the data, but s/he does not necessarily need to know the algorithms to be applied and s/he does not bother to choose the best value for the algorithm parameters.

The first step in the workflow shown in Figure 1 is a general characterization of the data, where some statistics and graphs concerning the various attributes of the dataset are computed (e.g. correlation matrix, boxplot and Cumulative Distribution Function). Then, to make the data ready for subsequent analysis, ADESCA removes the outliers in the dataset under analysis, identifying them with the DBSCAN clustering [28] (if there is no class label in the original dataset) or thanks to the boxplot analysis (if a label is present). In this last case ADESCA looks for the attributes that give a greater contribution than the others in choosing the final label. To find these attributes, an ANOVA univariate test [20] on each of them is performed, assigning this attribute a p-value. Each attribute associated with a p-value less than 0.05 is considered significant, and for each of these the outliers removal is done analyzing their boxplot representation [8] and eliminating each point less than the minimum (first quartile minus 1.5*Interquartile range) or greater than the maximum (third quartile plus 1.5*Interquartile range). The outliers points found are shown to the user in a very intuitive way, thanks to an interactive 3d scatter chart which represents all the points of the analyzed dataset according to its 3 most significant dimensions (found by applying the PCA [31]), where all the points labeled as 'outliers' are shown in red and the other points are shown in green.

But the most particularly innovative aspects that ADESCA offers include the *Automated Data Exploration* section, the *self-tuning strategy* and the *Data Storytelling* section, described below.

2.1 Automated Data Transformation

Data transformation plays a key role in the KDD process applied to any data type. Furthermore, the more complex the dataset, the greater the benefit that can be obtained by applying a data transformation technique. ADESCA integrates common transformation techniques such as standardization and One Hot Encoding [23], but also more innovative transformations based on pivot tables [16].

As a first attempt at this last type of transformations, we focus on the idea that reporting the same data in a different structure than the original one, new insights that were not very visible before, maybe can now become clearer. We preliminary applied the proposed strategy only on those datasets in which a temporal information exists (e.g. date, month, year, ecc.). Further data transformation techniques will be integrated later in ADESCA to deal with complex and heterogeneous data. The proposed transformation strategy integrated in ADESCA consists in considering only two of the original attributes, reporting the original data in the form of frequency values relative to the chosen attributes, as shown in the Figure 2, where the original clinical dataset on the left is transformed into a new dataset which reports for each row the history of the clinical exams carried out by each patient. The KDD process applied on the new dataset may lead to better insights from data. For example, interesting groups containing patients with a similar examination history (with standard or more specific examinations) might be discovered through a unsupervised learning methodology.

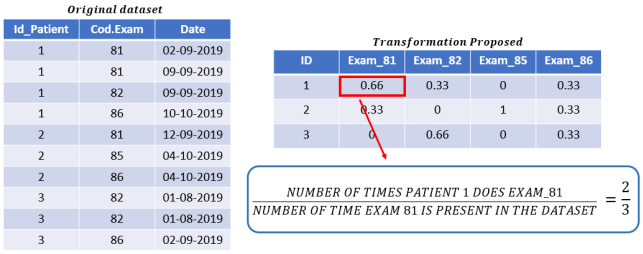

Figure 2: *Example of proposed transformation. The original dataset on the left is transformed according to his two attributes 'Id_Patient' and 'Cod.Exam'*

Moreover, not the results of all the transformations are shown, but ADESCA carries out analyzes and suggests only the transformations for which the results of these analyzes are better, providing us with a more interesting knowledge. The proposed analysis to identify the best transformations consists in: (i) considering each pair of attributes, merging together into a new single column, (ii) calculating the *Average Frequency* of each of these new columns and (iii) choosing only the transformations in which this value is between 1 and a threshold Δ found empirically after trying to apply these transformations on different datasets. Each transformation obtained is shown in a very user-friendly page that contains the scheme of the new transformed dataset, that the user can download entirely in csv format with a click. Then, to provide a graphic idea of the transformation, two interactive 3d scatter charts are shown, that represent the distribution

of the dataset before and after the transformation. The user can click on these graphs, enlarging and rotating them in all angles, allowing an effective comparison between them.

2.2 Data modeling through self-tuning unsupervised and supervised algorithms

A key point in the whole KDD process is that the results of any analysis are computed and graphically shown automatically, without the user having to interact with the system by choosing the most suitable algorithm and its parameter values. To guarantee the automatic nature of the process, we have proposed an ad-hoc methodology for the automatic configuration of the parameters of each algorithm used. These methodologies are particularly relevant for the cluster analysis and for the supervised learning techniques, where algorithms require parameters that greatly affect the quality of the results and so they must be chosen appropriately.

2.2.1 Self-Tuning Unsupervised Learning. As a first attempt, the unsupervised learning in ADESCA includes the cluster analysis performed through three different algorithms: K-means (which requires *k* as parameter), DBSCAN (which requires *MinPoints* and *Eps* like parameters) and Hierarchical clustering (which requires the *number of clusters* as parameter) [28] properly enriched with a self-tuning strategy to automatically identify the best input parameter setting. The results of each clustering algorithm are shown to the user through a clickable 3d scatter chart, in which each cluster is highlighted with a different color, as shown in Figure 3. The same colors are also used in a pie chart, which shows how many elements belong to each cluster and the respective percentages. If the algorithm used implies the concept of centroids, these are also reported as red points in the scatter chart. By clicking on one of these points, the radar chart of the centroid is shown, that jointly represents the value of each component of the centroid on different Cartesian axes having the same origin. Let's now see in detail the proposed self-tuning strategies that allow to automatically choose the input parameters of each algorithm integrated in ADESCA.

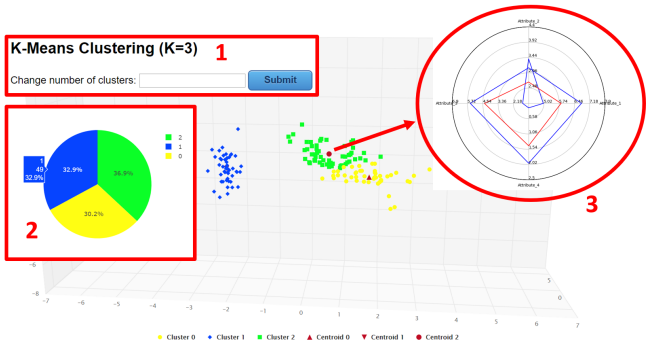

Figure 3: ***The algorithm used and the value of the parameters chosen automatically by ADESCA are shown in the first box. The user can then manually change the value of these parameters. The second box contains the pie chart with the same colors as the scatter chart, while in the circle on the right there is the radar chart obtained by clicking on the centroid related to cluster 2***

K-means. The choice of the k parameter is fundamental in the k-means algorithm, because it determines the number of clusters that will be obtained in the final partition. To look for the best solution of the algorithm, ADESCA calculated the k-means for each value of k in the interval [1,φ] and for each of these solutions the SSE (Sum of Squared Error) is calculated. Plotting these results graphically in a decreasing order of SSE, ADESCA looked for the elbow of the proposed curve, that it is the point where the increase in k will cause a very small decrease of the SSE, while the decrease in k will sharply increase the SSE. To calculate the elbow point automatically [27], we have drawn a straight line between the first point and the last point of our curve and we looked for the point of the curve with the greatest distance from the line. To provide the user with a graphical display of how this parameter has been chosen, the elbow graph is plotted next to the scatter chart, by effectively highlighting the elbow point chosen as the value of k.

DBSCAN. This algorithm requires as parameters a radius *Eps* and a number *MinPoints*. To automatically choose the MinPoints value to use, ADESCA calculates the k-dist graph for each value of k in a range between 2 and φ_1. Starting with k=2 and increasing this value by 1 each time, we have seen the differences with the (k+1)-dist graph using the MAPE percentage error [10]. If this percentage value is less than a pre-set threshold τ, we stop the algorithm and we choose the current k as MinPoints.

Using the value of MinPoints (k) just found, to find Eps we now calculate the elbow point of the sorted k-dist graph, with the same technique used in the case of k-means. To make the choice of these parameters more meaningful to the user, next to the 3d scatter chart, the sorted k dist graph is plotted, with the elbow point highlighted.

Hierarchical Clustering. This algorithm requires as a parameter the number of clusters to be obtained in the final result. To find the most suitable value for this parameter, we used the Silhouette index [25], calculating the value of Average Silhouette for each algorithm with a number of clusters between 2 and a pre-set threshold φ_2. The number of clusters chosen as a parameter will be the one associated with the higher Average Silhouette.

The result of the Hierarchical clustering is then visualized graphically in a very user-friendly dendrogram, that is a tree-like diagram, which effectively shows how the various clusters have been grouped together.

2.2.2 Self-Tuning Supervised Learning. The self-tuning approach in ADESCA provides an automatic overview of the predictive performance of a wide set of classification algorithms, whose parameters are automatically optimized by a grid-search, hence automatically identifying the best one and its performance gain with respect to the others. Prediction performance is evaluated by exploiting a 10-fold stratified cross-validation, and the machine learning algorithms used in this section are Support Vector Machine, Decision Tree, K-Nearest Neighbor, and Naive Bayes Classifier [1].

2.3 Storytelling

In the Storytelling section, the most salient information regarding the uploaded dataset are automatically shown to the user through a storyboard (i.e. visual layout mechanism able to summarize specific events and data) that allows addressing visual analytic challenges. We have designed a storyboard, in which all the salient information are shown to the user on different pages. However, different than the traditional concept of storyboard [30], that is always placed on a timeline, we removed the temporal dependence between visualized pages. In fact, the different

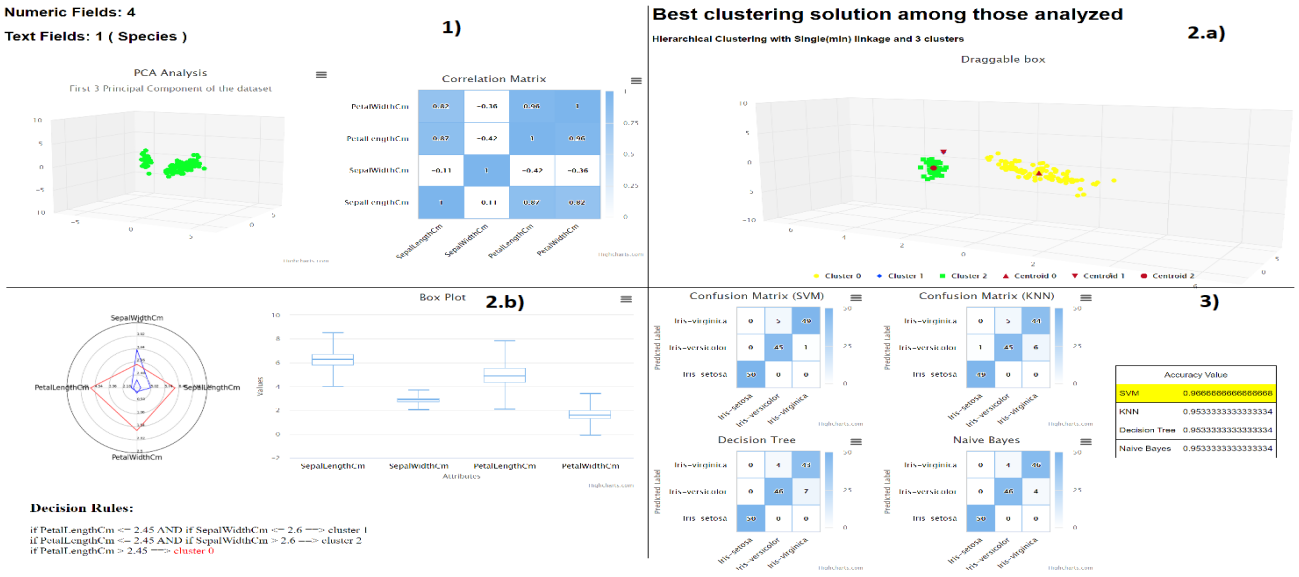

Figure 4: ***D1 Storytelling section. The names of attributes in screen 1 and screen 2.b are: "SepalLength, SepalWidth, PetalLength, PetalWidth". The names of the labels in screen 3 are: "Iris-setosa, Iris-versicolor, Iris-virginica"***

pages that make up our section are independent of each other, as each page contains results relating to a particular analysis, which is unrelated to the others.

Specifically, the purpose of this data-driven storytelling is to tell to the curious-data user the story of the uploaded data automatically, so that the user can have an immediate overview of the insights hidden on the data, without having the technical and methodological expertise on the underlying algorithms. Structures and models hidden in the data are reported in a completely automatically way and the user is not required to interact anymore with the application, s/he can focus her/his attention to the discovered insights. In practice, the information contained here were always present also in previous sections; but while in the other sections all the possible results were shown in cascade, now ADESCA only shows the most interesting solutions, that are those that mostly characterize the dataset. To provide this summary, the Storytelling section tries to offer a visual layout strategy able to summarize the most important concepts. As a first attempt, the storyboard includes four different views, each of which contains results of different analyzes. More details of each view in the storyboard are reported below.

First View - Data Characterization. ADESCA shows a brief overview of the dataset, including general information (number of rows, number of columns, list of attributes, ecc), a 3d scatter chart of the distribution and a correlation matrix containing the Pearson coeffcients between the various numeric fields. Each cell of the matrix is clickable and allows to open a pop up window containing the boxplots and the Cumulative Distribution Functions concerning the attributes related to the clicked cell.

Second View - Unsupervised Learning. When the user exploits ADESCA step by step, the clustering section provides all the results obtained by the various clustering algorithms performed with the parameters chosen automatically by the proposed self-tuning strategy. And later the user can also see new results by manually changing these parameters. But now, in the second view of the data-driven storyboard, between all the solutions automatically computed by ADESCA and those manually selected by the user (if any), only the best one is displayed (chosen using the Silhouette score). This solution is shown as a 3d scatter chart, depicting each cluster with a different color. By clicking on a point belonging to a particular cluster, a new page is opened containing specific information characterizing that cluster content (e.g., radar chart, boxplots, paths extracted from the decision tree built on that solution).

Third View - Supervised Learning. This page is present only if the original data set contains a label. In this case, the results of a series of classification algorithms are shown here. For each integrated algorithm a confusion matrix is shown to compare the predicted labels with the original ones, by highlighting the algorithm with the highest performance.

Fourth View - Data Transformation. This page shows the three best transformations between those obtained in the appropriate section, reporting for each of these the scheme of the new dataset and its distribution in a 3d scatter chart. To choose which are the best transformations among those obtained, we apply to each of them the same clustering algorithm by automatically set the input parameter. The transformation that yielded the highest silhouette score is selected. This page is present only if there is a temporal attribute in the original dataset.

3 PRELIMINARY DEVELOPMENT

A preliminary implementation of ADESCA has been developed in Python [24], including the scikitlearn library [22] (for the analytic tasks) and the pandas library [19] (for manipulating data in a tabular or sequential format). To build the architecture of the system we used the micro-framework Flask [11], while the graphic part of the web pages is managed thanks to CSS and Javascript [7], including the HighCharts library [18] for creating interactive graphs.

ADESCA has been experimentally evaluated by analyzing different datasets, using τ=3, φ=10, φ_1=50 and φ_2=11 as default values in the clustering analysis, and Δ=2.5 as a default value of the parameter in the automated data transformation.

As a first try, ADESCA was promoted at the 'Festival della Tecnologia' [5], held at the Politecnico di Torino on November 8th 2019. During this event, about 200 people composed of university students and students of the last year of high school were able to test our tool. What made this event significant was the fact that most of the students who used ADESCA had very limited skills in areas such as data exploration or data mining. However, they managed to understand the tool without problems, also testing it in its various analyzes, and easily understanding automated insights extracted from data. This experience showed us the practicality and the user-friendliness of the work done.

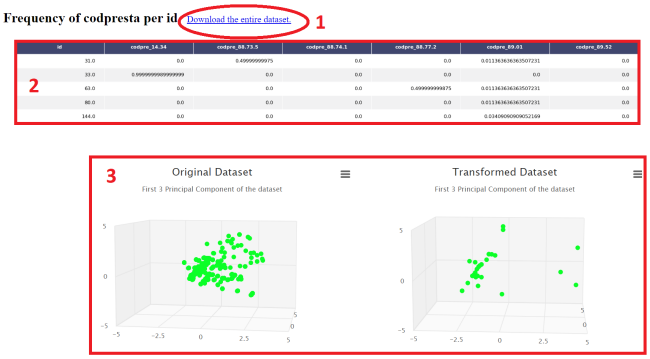

Figure 5: ***Layout that illustrates how the page showing each transformation obtained is divided. In rectangle 2 the first five lines of the new dataset are shown, while clicking in the red circle at the top the user can download the entire dataset in csv format. Finally, in rectangle 3 there is the comparison between the 3d distribution of the dataset before and after the transformation***

4 PRELIMINARY EXPERIMENTAL RESULTS

The objective of the preliminary experiments is to demonstrate the capability of ADESCA in discovering useful knowledge items in an automated fashion and presenting only the relevant outcomes to the end user. To this aim, dataset size does impact neither negative nor positive in the analysis. Furthermore, for reasons of space, we will not report all the analyzes performed by the tool, but we will report only the results of the storytelling section, which is the most significant and innovative section. The first dataset analyzed, D1, contains a label attribute, while the second dataset, D2, contains a temporal attribute. D1 is called 'Iris', a free dataset available at the UCI Machine Learning Repository [6]. It contains 150 records, each of which represents a different flower. Each flower is characterized by a series of numerical attributes and it belongs to a 'Species' label. Each species contains 50 flowers. D2 is instead a dataset containing medical information. In particular, it has 500 lines and each line contains information about the exam made by a patient, on a given date (temporal attribute). Table 1 contains information about the execution time (in seconds) of each section of ADESCA, separately per dataset. The computational cost of the storytelling section is obtained by summing the time of each page that composes it.

The results of the Storytelling section for D1 are shown in Figure 4 in four screens. The first screen shows the results of the data characterization. Screen 2.a shows the best clustering solution to apply to the dataset, while screen 2.b is a window containing information related to a specific cluster and is obtained

[5]https://www.festivaltecnologia.it/

Table 1: ADESCA Computational Time

	D1	D2
Data Characterization	1.58	2.01
Clustering	9.34	14.37
Storytelling	2.09	9.56
Data Transformation	-	1.50

by clicking on a point of it. Finally, screen 3 shows the results of the classification techniques and it is present because the original dataset contains a label.
Regarding instead the storytelling section of D2, the results related to data characterization and clustering analysis are in the same form as those reported for D1. But, since D2 does not have a label, the page containing the classification algorithms is no longer present. In its place there is a page indicating the three best possible transformations for D2 (because now a temporal attribute is present). Each of these transformations is shown to the user on a page with the same layout reported in the Figure 5.

In particular, the first proposed transformation contains a record for each patient and represents the history of clinical examinations performed by each patient. Then, applying unsupervised analysis on the dataset before and after the transformation, we can note that on the original dataset we are not able to find subgroups (because it is not relevant to group individual prescriptions together), while in the second case we are grouping the patients and from the automated cluster analysis we can clearly see two distinct groups containing patients with similar examination histories. From this result, we can deduce that the transformation in this case was useful, because we are now able to capture a model from the data that was not previously visible.

5 OPEN ISSUES AND FUTURE DIRECTIONS

A first attempt towards automated data exploration has been achieved through ADESCA. Preliminary experimental results on small datasets demonstrate the friendliness of the knowledge extraction approach towards the democratization of data science. The storytelling capability, based on storyboard, allows curious-data users easily analyzing their data through visual layout mechanism able to summarize specific events and data without technical and methodological expertise. Specifically, thanks to ADESCA, all users, even the less experienced, can take advantage of their data, by viewing only their most salient aspects in the appropriate storyboard. Moreover, since the algorithms and the parameters are chosen automatically by ADESCA, the proposed methodology is suitable for all those non-technical users who can draw interesting insights from their data. Obviously, what has been done so far must be considered as a preliminary step. Data exploration is a very complex topic, and making it automatically makes everything even difficult.
The idea is to extend ADESCA, covering topics that are currently still open, like: (i) the management of heterogeneous data types (e.g. combination of time series data with unstructured data, images [29] or audio signals), (ii) the capability to perform feature engineering to capture specific data properties, (iii) the integration of more complex machine learning algorithms able to deal with unstructured and high dimensional data [5], (iv) the ability to deal with huge datasets, (v) the capability to enrich the data under analysis with additional and related open-data, (vi) the

capability of properly managing geo-referenced [4] and time-series data [3], and (vii) the improvement of the proposed data transformation strategies to obtain more generalizable outcomes.

REFERENCES

[1] Charu C Aggarwal. 2014. *Data classification: algorithms and applications.* CRC press.

[2] Marcello Buoncristiano, Giansalvatore Mecca, Elisa Quintarelli, Manuel Roveri, Donatello Santoro, and Letizia Tanca. 2015. Database Challenges for Exploratory Computing. *SIGMOD Rec, Association for Computing Machinery* 44, 2 (Aug. 2015), 17–22. https://doi.org/10.1145/2814710.2814714

[3] L. Cagliero, T. Cerquitelli, S. Chiusano, P. Garza, and X. Xiao. 2017. Predicting critical conditions in bicycle sharing systems. *Computing* 99, 1 (2017), 39–57. https://doi.org/10.1007/s00607-016-0505-x cited By 14.

[4] Tania Cerquitelli, Evelina Di Corso, Stefano Proto, Paolo Bethaz, Daniele Mazzarelli, Alfonso Capozzoli, Elena Baralis, Marco Mellia, Silvia Casagrande, and Martina Tamburini. 2020. A Data-Driven Energy Platform: From Energy Performance Certificates to Human-Readable Knowledge through Dynamic High-Resolution Geospatial Maps. *Electronics* 9, 12 (2020). https://doi.org/10.3390/electronics9122132

[5] Evelina Di Corso, Tania Cerquitelli, and Francesco Ventura. 2017. Self-tuning techniques for large scale cluster analysis on textual data collections. In *Proceedings of the Symposium on Applied Computing, SAC 2017, Marrakech, Morocco, April 3-7, 2017*, Ahmed Seffah, Birgit Penzenstadler, Carina Alves, and Xin Peng (Eds.). ACM, 771–776.

[6] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[7] David Flanagan. 2006. *JavaScript: the definitive guide.* " O'Reilly Media, Inc.".

[8] Michael Galarnyk. 2018. *Understanding Boxplots.* https://towardsdatascience.com/understanding-boxplots-5e2df7bcbd51

[9] Antonio Giuzio, Giansalvatore Mecca, Elisa Quintarelli, Manuel Roveri, Donatello Santoro, and Letizia Tanca. 2019. INDIANA: An interactive system for assisting database exploration. *Information Systems* 83 (2019), 40–56.

[10] Paul Goodwin and Richard Lawton. 1999. On the asymmetry of the symmetric MAPE. *International Journal of Forecasting* 15, 4 (1999), 405–408. https://EconPapers.repec.org/RePEc:eee:intfor:v:15:y:1999:i:4:p:405-408

[11] Miguel Grinberg. 2018. *Flask web development: developing web applications with python.* " O'Reilly Media, Inc.".

[12] Pat Hanrahan. 2012. Analytic database technologies for a new kind of user: the data enthusiast. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data.* ACM, 577–578.

[13] Kevin Hu, Diana Orghian, and César Hidalgo. 2018. Dive: A mixed-initiative system supporting integrated data exploration workflows. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics.* ACM, 5.

[14] Stratos Idreos, Olga Papaemmanouil, and Surajit Chaudhuri. 2015. Overview of data exploration techniques. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data.* ACM, 277–281.

[15] Prasanth Jayachandran, Karthik Tunga, Niranjan Kamat, and Arnab Nandi. 2014. Combining User Interaction, Speculative Query Execution and Sampling in the DICE System. *Proc. VLDB Endow.* 7, 13 (Aug. 2014), 1697–1700. https://doi.org/10.14778/2733004.2733064

[16] Bill Jelen and Michael Alexander. 2010. *Pivot Table Data Crunching: Microsoft Excel 2010.* Pearson Education.

[17] Udayan Khurana, Srinivasan Parthasarathy, and Deepak S. Turaga. 2014. READ: Rapid data Exploration, Analysis and Discovery. In *EDBT*. 612–615.

[18] Joe Kuan. 2015. *Learning highcharts 4.* Packt Publishing Ltd.

[19] Wes McKinney et al. 2011. pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing, Seattle* 14, 9 (2011), 1–9.

[20] Rupert G Miller Jr. 1997. *Beyond ANOVA: basics of applied statistics.* CRC press.

[21] Kristi Morton, Magdalena Balazinska, Dan Grossman, and Jock Mackinlay. 2014. Support the data enthusiast: Challenges for next-generation data-analysis systems. *Proceedings of the VLDB Endowment* 7, 6 (2014), 453–456.

[22] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.

[23] Kedar Potdar, Taher Pardawala, and Chinmay Pai. 2017. A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. *International Journal of Computer Applications* 175 (10 2017), 7–9. https://doi.org/10.5120/ijca2017915495

[24] Guido Rossum. 1995. *Python Reference Manual.* Technical Report. Amsterdam, The Netherlands, The Netherlands.

[25] Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics* 20 (1987), 53–65.

[26] Tarique Siddiqui, Albert Kim, John Lee, Karrie Karahalios, and Aditya Parameswaran. 2016. Effortless Data Exploration with Zenvisage: An Expressive and Interactive Visual Analytics System. *Proc. VLDB Endow.* 10, 4 (Nov. 2016), 457–468. https://doi.org/10.14778/3025111.3025126

[27] Manoj Singh. 2017. *Finding the elbow or knee of a curve.* https://dataplatform.cloud.ibm.com/analytics/notebooks/54d79c2a-f155-40ec-93ec-ed05b58afa39/view?access_token=6d8ec910cf2a1b3901c721fcb94638563cd646fe14400fecbb76cea6aaae2fb1

[28] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2013. Data mining cluster analysis: basic concepts and algorithms. *Introduction to data mining, Pearson Education India* (2013), 487–533.

[29] Bartolomeo Vacchetti, Tania Cerquitelli, and Riccardo Antonino. 2020. Cinematographic Shot Classification through Deep Learning. In *44th IEEE Annual Computers, Software, and Applications Conference, COMPSAC 2020, Madrid, Spain, July 13-17, 2020.* IEEE, 345–350.

[30] Rick Walker, Llyr Ap Cenydd, Serban Pop, Helen C Miles, Chris J Hughes, William J Teahan, and Jonathan C Roberts. 2015. Storyboarding for visual analytics. *Information Visualization* 14, 1 (2015), 27–50.

[31] Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems, Elsevier* 2, 1-3 (1987), 37–52.

[32] Doris Xin, Litian Ma, Jialin Liu, Stephen Macke, Shuchen Song, and Aditya G. Parameswaran. 2018. Helix: Accelerating Human-in-the-loop Machine Learning. *CoRR* abs/1808.01095 (2018). arXiv:1808.01095 http://arxiv.org/abs/1808.01095