Educational Tool for Generation and Analysis of Multidimensional Modeling on Data Warehouse

Elena Fabiola Ruiz Ledesma[1], Elizabeth Moreno Galván[2]
Enrique Alfonso Carmona García[3], Laura Ivoone Garay Jiménez[4]
Instituto Politécnico Nacional, Escuela Superior de Cómputo, Ciudad de México, México[1]
Instituto Politécnico Nacional, Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas
Ciudad de México, México[2, 3, 4]

***Abstract*—The curricular inclusion of topics, study plans, and teaching programs related to the study of Data Science has been trending mostly in higher-level education for the last years. However, the previous knowledge requirements for students to adequately assimilate these lessons are more specialised than the ones they obtain during secondary education. On the one hand, the interaction with complexes techniques and materials is needed, and on the other, tools to practice on-demand are required in the current learning. So, this is an excellent opportunity for the creation of data analysis tools for educational purpose that could be considered as a starting point of a broad area of application. This paper presents a pedagogical support tool aimed to facilitate the student approach to the basic knowledge of data mining through the practice of the analysis of online analytical processing (OLAP). It is a prototype that allows the visualisation of the multidimensional cubes generated with all possible combinations of the dimensions of the data set, as well as their storage in databases, the recovery operations for views, and the implementation of an algorithm for the selection of the optimal view set for materialising the set of records resulting from a search of the database, and computing the materialisation costs and total records recovered. The prototype also carries out and present recurrent patterns and association rules while considering factors such as support variables and reliability. All of this is steps are done explicitly to aid the students to comprehend the generation process of data cubes in the data mining discipline.**

***Keywords*—Educational data mining; data cube; view materialization; educational software**

I. Introduction

The extensive analysis of Big Data [1] comes from a branch on statistical analysis that companies used to identify spending trends, which is used to predict the consumer's behaviour and to analyse commercial activities. From this initial idea, several data handling techniques have been created, such as Data Mining (DM) or Machine Learning (ML). They have been successfully applied to a range of human effort areas including detection of illness and mobile health [3], [4], [5], [6], environmental and pollution studies [7], [8], [9], [10], being these only examples of the many areas to which these techniques have been recently applied. In the educational field, there are different applications such as educational data mining and learning analytics [11], [12], whose purpose is oriented towards the designing of algorithms, methods, and models, that will allow exploring data from learning environments.

In order to study the data, there is a constant need for a data warehouse. Gathering data from a company or organisation in a single database helps analysts and managers to support their decisions or to find valuable data, but reduce the extraction time and cost are constant requirement [13], [14], [15]. Moreover, the design and construction of a data warehouse require the application of extraction, integration, transformation, and data cleaning processes [16]. So the data warehouse increases their dimension, and a multidimensional model is going to be required. This model is going to be described and defined along with the optimal view set to be materialised. The last process consists of determining the necessary tools for viewing data.

The most traditional tools for data mining and automated learning are becoming insufficient as the tendencies in technology progress, prompting the creation of new and increasingly powerful, complex solutions. In the academic sense, from the students' point of view, the vast availability of these new tools and methods, and its many overlapping uses represents a challenge. An excess in variety makes selecting a study and comprehension strategy even harder than it usually is, especially in their primer approach.

This document is focused on the data preparation from a data warehouse, and it uses a software tool that automatically designs a multidimensional model and helps into the creation and storing of the data warehouse in a database. Once the outline and the hypercube materialisation through previous calculation are ready, the results of the information retrieval are sped up. Also, the optimal view set selection algorithm proposed by Harinarayan is applied in materialising [17]. In the last step, the tool can determine the frequent patterns present in data, and it also calculates the association rules, considering predetermined support and reliability.

Parameters about the performance of this tool, such as the effectiveness and time-saving in calculating, shows an improved performance over the manual method of the same procedure that it is commonly presented to the students in the courses about essential data mining topics.

II. Literature Review

Data analysis is the process of working on data in order to discover useful information for business decision making. Data Analytical Tools are software developed to perform data analysis tasks such as process and manipulate data, analysing relationships and correlations, as well as identifying patterns and trends for interpretation.

In recent years, a large variety of Data Analytical Tools have been created to carry out data science tasks. However, in this section, we will present some of the most used and accessible tools that are currently available for teaching this topic, all of them related to Educational Data Mining [18] or Learning Analytics [11][19] instead of the broader array of tools that could be used for the most modern statistics analysis, since these are complex for the beginner student.

Two characteristics that are commonly present in Data Analytical Tools on the educational field; they are the integration of the functionalities of data mining, and the application of techniques dedicated to data mining for didactic purposes [20]. Nevertheless, innovation in education has become relevant, so several projects appear to apply it. For example, the Hadoop Ecosystem has been designed to help researchers and students in all aspects of typical data analysis and automatic learning processes (Machine Learning) [21].

In the revision work by S. Slater et al. [22], there is an analysis of several didactic and research tools for educational data mining, classifying them as follows:

- Data Analytical Tools for the handling, cleaning, and formatting of the data, per example: Microsoft Excel and EDM Workbench.
- Data Analytical Tools for model selection and testing, also identification, mapping, exploration and analysis of relationships such as RapidMiner, Weka, KEEL, KNIME, Orange, and SPSS.
- Data Analytical Tools for visualisation of the structure of the tree methodology as Tableau, d3js, and InfoVis.

A. *Data Analytical Tools*

S. Yadav and Urbina provide a list of analytical tools, and their descriptions, as well as the definition [2], [27]. In Table I, the most relevant characteristics are listed, and the tools that could be used to educational porpoises are identified.

TABLE I. TOOLS FOR DATA ANALYSIS

Tool	**Description**
WEKA*	Implements algorithms for data preprocessing, classification, regression, grouping, association rules, and viewing. It is free to use under the Public License GNU, and it contains a wide range of modelling and data processing techniques. [2].
Orange*	This software allows preprocessing, information filters, data modelling, evaluation, and exploration of modelling techniques.
Rapid Miner*	Predictive analysis tool. It is sturdy, easy to use, and has a broad open-source community where the users can integrate their self-made specialised algorithms. It provides the user with learning schemes, models, and algorithms from WEKA and R [2].
Rattle	Free, open-source data mining toolset that is written in the statistic language R. It presents visual and statistical data summaries [2].
Knime*	Integrative software allows data processing, analysis, and exploration as well as advanced prediction algorithms and machine learning.
CLUTO	This tool is for grouping high and low dimension data analysis. It has multiple classes of algorithms for grouping such as partition, agglomeration, and graphic-based partition, similarity/distance functions as Euclidian distance, cosine, correlation coefficient, extended Jaccard, and even self-defined functions [2].
Jaspersoft BI Suite	An open-source suite produces reports based on database columns, reducing the data from the sources to tables and interactive graphics.
Pentaho Business Analytics	Software platform that simplifies the information inclusion from the different sources
Talend Open Studio	Offers a development environment for linking Hadoop data processing works.
Splunk	This tool creates a data index such as a book's structure or a text block.
Apache Storm	A distributed computing system that allows the user to process unlimited data in a reliable real-time way.
Apache Drill	It is an SQL search engine for Big Data exploration. It has been designed from scratch to allow high-performance analysis in semi-structured data.
Cassandra	It is used by large active data sets, coming from Netflix, Twitter.
HBase	A distributed database management system built upon the Hadoop file system and oriented towards columns format,
Neo4j	It is a native graphics database management system which uses the data relations as the first-class entities. It has upgraded performance in comparison to relational databases.
CouchDB	A database management system that is wholly dedicated to web applications, storing data in JSON files.
OrientDB	It combines the flexibility of file databases with graphic databases.
FlockDB	It is an open-source database management system that uses a wide but shallow network graphics. It was designed to store social graphics.
MOA (Massive Online Analysis)	It is a project designed in partnership with WEKA that offers flow analysis online for various WEKA algorithms and with the same user interface [24].
MADlib	It is a collection of SQL-based algorithms; it includes grouping, classification, regression and themed models as well as validation tools [23].
Dato, before GraphLab	It is an independent product that can be connected to Hadoop for graph analysis and machine learning tasks [25].

. * For educational purpose, (Modified from Source: [2],[23], [24],[25]) In the literature review, some of the tools are focused on the learning-teaching area of data science. However, their focus is not entirely didactic for non-experienced users. Because they are for specialised tasks and depending on the data set type, a choice of tool is made. So, these tools have more complexity than required for a basic implementation. In many cases, these tools are developed for more experienced users or area professionals, what it offers an opportunity area for pedagogical tools in this field.

III. Architecture and Design of the Tool

The Data Analytical Tool presented in this paper is developed in Java, and it generates the multidimensional variant for Online Analytical Processing (OLAP) named multidimensional structure (MOLAP). For this purpose, it uses relational database modelling technology for the construction of the data warehouse (DW) in a MySQL system by reading a data source in CSV format separated by commas. In this way, the system creates a multidimensional model generated by a table for each dimension or column in the data source.

The Data Mining based on Lattice is a method to organise data in domains determined by combinations of the dimensions of a dataset [26]. These combinations can be determined by information retrieval with SQL structures named views. The system can calculate the views from the MySQL database and managing a cache memory file by making use of a linked list structure [27], and a modified B-Tree [28], where each node in the tree constitutes a view also named cuboid from the Lattice. This system generates both the logic and the visual representation of a data cube.

Materialised views are structures that improve data access time by precomputing intermediary results; an effective technique for improving query performance is using indexing [29]. In this sense, the algorithm proposed by Harinarayan [17],[30] is used to improve the efficiency of the model for determining the pre-calculated views to materialise, and his algorithm is implemented into the Data Analytical Tool. Given the educational approach of this development, the execution tests were carried out in a synthetically created data cube; that is, with data sources created by an additional computer program, and the measurements were randomly generated.

A. *Architecture of the Data Analytical Tool*

The general tool's architecture is shown in Fig. 1, and the specific elements are described and detailed in the following paragraphs.

The data are obtained from an external data source in CSV format, which is used to fill out a database structured into tables, matching each file's header with a field of the table. The cube is comprised of a vector (D1, D2, …, DN) with N dimensions corresponding to the attributes of the database. The generated cube or Lattice L contains a cartesian product of the N dimensions. Each cuboid represents a possible useful aggregation from structured language queries (SQL) known as views. The materialisation of these views and its efficiency is based on the Harinarayan algorithm implementation. In this work, queries and aggregations are optimised, first, by application of operations like slicing in OLAP that generates data columns corresponding to single values with at least one dimension. Then, it helps the visualisation and recompilation of information about a specific dimension. Finally, dice operation in OLAP is provided, which selects a subset of dimensions considering a specific values range in each dimension.

In the analysis of the patterns, the data source must be a binary matrix, this means that the columns constitute dimensions and the rows are corresponding with the records. So if a record has the dimension, the intersection is filled with a value 1, or 0 on the contrary case.

B. *Lattice Construction*

For the software implementation, making use of the modified B-Tree structure, each view is represented by a node, as shown in Fig. 2. In this case, dimensions, the node attribute is previously determined by the number of attributes of this node; ancestors are the previous nodes, and descendants are the subsequent nodes. Both last sets contain at least one shared dimension with the actual one.

For example, for the three-dimensional data source, N=3 and 2^N possible groupings (nodes) are generated. Their relationship can be appreciated in Fig. 3, and it is essential to note the complexity involved in that excepting the apex and base (Node 7), each node possesses various descendants and several ancestors at the same time.

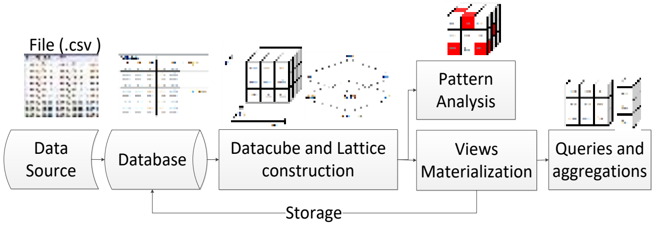


Fig. 1. The Architecture of the Proposed Tool).

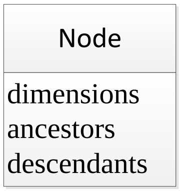


Fig. 2. Node Structure.

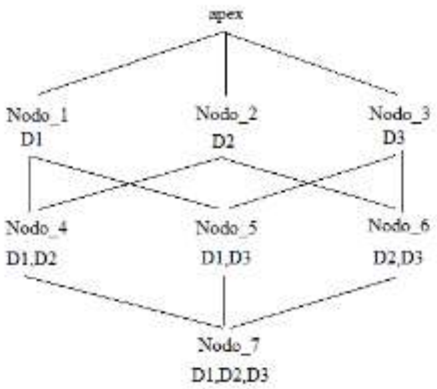


Fig. 3. Three-Dimensional Lattice Structure.

The information of the Lattice's constitution is stored in each node as has been shown in Fig. 2. So apex is 0 dimension, Nodes 1,2, and 3, have one dimension and Node 4,5,6, have two dimensions associated. Node 7 is considered the base cuboid. In this example, Fig. 4 shows the structure of Node 5 as an example; it has two ancestors with dimensions D1 and D2, and its descendants that have its dimensions and D2.

The order of the interconnected node net structure described by the Lattice contains all views that can be used to get any query related to a business question, as well as materialising or to pre-calculating the cuboids. However, it is crucial to know the physical space limitations in the storage unit. It is also recommended to materialise the base cuboid (full detail, apex) as it can be used to respond to any question, and then move on to the less costly views, which results in less time and resources to obtain the desired answers. The generated structure allows the application of other algorithms, so the computational cost that is inverted is justified.

The pseudocode for the creation of the logical structure for the data cube is shown in Fig. 5.

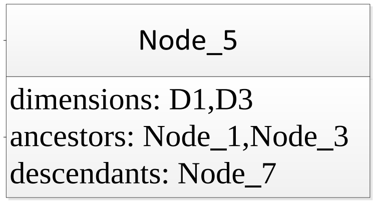


Fig. 4. Representation of the Values of an Example Node.

```
// Creating a dimension list
for of each dimension
 ListDimensions <- name dimension

// Creation of apex node
create a new node Nodeactual
Nodeactual.dimensions <- apex
Nodeactual.ancestors <- null
Nodeactual.descendants <-null

// Creation of Nodes for storing one dimension cuboids
for each dimension in ListDimensions
 create a new node Nodeactual
 Nodeactual.dimensions <- name dimension
 Nodeactual.ancestors <-apex
 Nodeactual.descendants <-null

//Creation of nodes for storing cuboids of 2 or more dimensions
Until all levels are covered
 for each dimension in ListDimension
 create new node Nodeactual
 Nodeactual.descendants <-null
 for all groups of n dimensions in ListDimensions
 for all nodes in previous level
 if dimension in Nodenivel -1
            Add Nodenivel -1 to the list of ancestors from the actual
            node
            Add Nodeactual to the list of ancestors from Nodenivel -
            1
```

Fig. 5. Multidimensional Cube Generation Pseudocode.

C. View Materialisation

The formulation of business questions can be carried out through structure query language expressions (SQL), based on the database previously stored. Furthermore, it is possible to use operators on numeric-type dimensions of the views resulting in answering expressions, for example:

SELECT field, Op field FROM table GROUP BY field

Where, *field*: is a subset of database attributes or dimensions D1, D2, …, DN and *Op (field)* is an operation in a numeric-type dimension, as COUNT, SUM, MAX, MIN.

The cost of generation of a view, represented by C(v) is associated to the computational cost of using a view considering that it decreases with the number of dependent relationships, thus, the calculation cost is divided between the total number of dependent relationships.

Therefore, the construction of the net is modified, as shown in Fig. 6.

An advantage of this new structure is to get a straightforward application of Harinarayan's Greedy algorithm proposal [17]. An efficient view generation is done, as shown in the pseudocode in Fig. 7. After selecting a view set named S, the benefit of the view v, denoted by B (v, S) is calculated. B is the difference in cost of storing a descendant view and the cost of its ancestor view and then multiply the difference by the number of relationships dependent to view v. The only views that benefit from the materialisation of v are the ones that can be calculated from v, including the v itself. The list of these views is named as w.

Therefore, the total benefit is the sum of all the benefits from the w set. In Fig. 7, the pseudocode can be appreciated.

The for testing the of use of the developed Data Analytical Tool and its effectiveness in the classroom, it was used in production for the analysis of data from several experimental datasets, speeding up the obtaining of results for the respective case studies was reported. Then a follow session was done, and the results are presented in the next section.

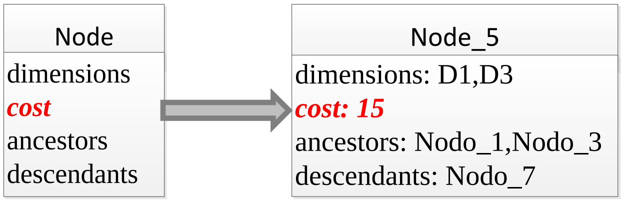
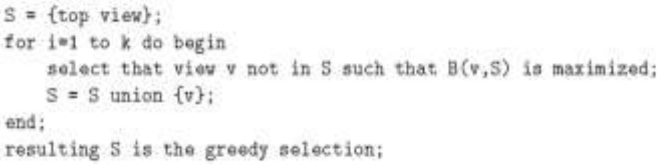


Fig. 6. Addition of the Cost Attribute to the Nodes.



```
S = {top view};
for i=1 to k do begin
    select that view v not in S such that B(v,S) is maximized;
    S = S union {v};
end;
resulting S is the greedy selection;
```

Fig. 7. Greedy Algorithm (Source: Harinayaran[17]).

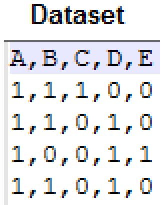
IV. Results

In this section are presented the efficiency of the tool to show the results about the calculus of the cuboid, definition of the association rules, and MOLAP visual representation using synthetic data as input.

The test data consists of a synthetic input composed of five dimensions named A, B, C, D and E respectively fulfilled with four records with random 1 and 0 numeric data values. The small size of the synthetic dataset was selected for display purposes as shown in Fig. 8, but the data capacity is limited by MySQL restrictions whose consists on the storage engine such as InnoDB that supports a maximum of 65,535 bytes per row limited by the data type that it hosts, that is approximate of 1.073.741.824 rows.

The corresponding lattice representation must look as shown in Fig. 9, where a labelled node represents each combination of dimension.

In the interface, this Lattice is represented by a button for each node. It replaces the names of the dimensions and respectively views by numbers, preventing dimension names from being longer, so the example lattice is shown in Fig. 10.



```
Dataset
A,B,C,D,E
1,1,1,0,0
1,1,0,1,0
1,0,0,1,1
1,1,0,1,0
```

Fig. 8. Test Synthetic Dataset.

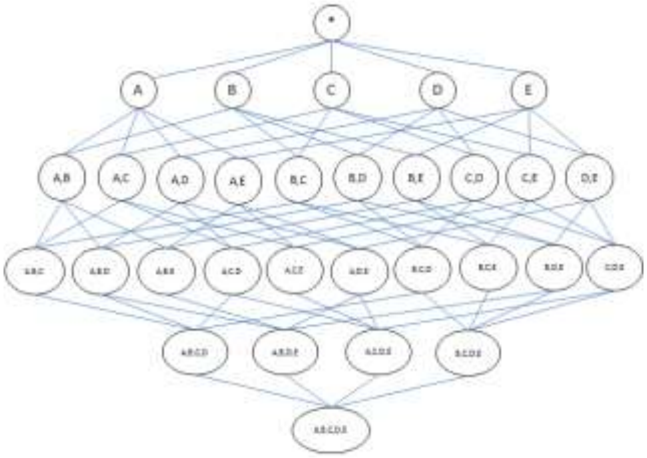


Fig. 9. Lattice Representation.

Fig. 10. Lattice Representation of the Synthetic Dataset Created by the Tool.

After the lattice generation, all the simple views whose consists of SQL sentences such as *select a,b from table*, are calculated automatically. Then, the visualisation of any view is obtained clicking on the button that represents the node of interest. A new window will appear showing the records that comply with the query. In Fig. 11, can be appreciated the result of clicking the button with numbers 0,1,2 that correspond to the node that relates the dimensions A, B and C.

Once, all the possible views to generate are available, the analysis of which ones are adequate for being materialised is carried out using the Harinarayan algorithm, the calculation provided by the tool is shown in Fig. 12 with N=4.

The result is the materialised view set as S = {V2, V4, V7, V5, V10, V6, V12, V3}, according to the employed method. In this output table, the student could analyse the procedure of optimisation whose manual calculations would have been complicated and time-consuming.

In data mining, the technique used to find item sets, subsequences, or substructures that appear in a data set frequently (patterns), requires the following definitions and operations.

Considering I = {I_1, I_2,..., I_m} as a set of items or dimensions, D the task-relevant data, T a set of items such that T ⊆ I. Let A be a set of items, a transaction T is said to contain A if and only if A⊆T. An association rule is an implication of the form A⇒B, where A ⊂ I, B ⊂ I, and A ∩ B = φ. The rule A ⇒ B has confidence c in the transaction set D, where c is the percentage of transactions in D containing A that also contains B. It is taken to be the conditional probability, P(B|A). Then

$$\text{support}(A \Rightarrow B) = P(A \cup B) \tag{1}$$

$$\text{confidence}(A \Rightarrow B) = P(B|A) \tag{2}$$

Rules are called strong when (1) and (2) are satisfied with a, a minimum support threshold (min sup) and a minimum confidence threshold (min conf). If the relative support of an itemset I satisfy a pre-specified minimum support threshold, then I is a frequent itemset [31].

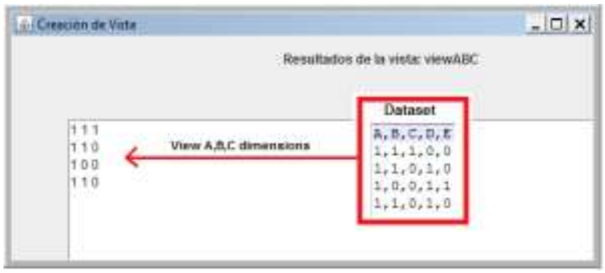
To show this calculation by the program, the frequent patterns that comply with the minimum support requirements (i.e. min sup: 2) are highlighted with black buttons in the interface, as shown in Fig. 13, so the students could visualise the combination of frequent dimensions in the dataset.

From Equation (2), we have

$$\text{confidence}(A \Rightarrow B) = P(B|A) = \frac{\text{support}(A \cup B)}{\text{support}(A)} = \frac{\text{support count}(A \cup B)}{\text{support count}(A)} \tag{3}$$

Equation (3) shows that the confidence of rule A⇒B can be derived from the support counts of A and A∪B, and it is straightforward to derive the corresponding association rules A⇒B and B⇒A [31]. Then, the association rule for a relation e.g. {A,D}, is calculated as follows: conf ({B} → {A,D} in the interface: conf ({1}→ {0,3} = supp({0,3})/ supp({1}) = 3 / 3 = 1.0. At last, the association rules are determined for those who complied the minimum confidence value (i.e. 60% or 0.60), and the interface displays the result, as shown in Fig. 14.

The students reported that the intuitive interface of the tool focused on concrete operations supported them not to spend extra time in software configurations or learning a complicated interface for the same purpose.

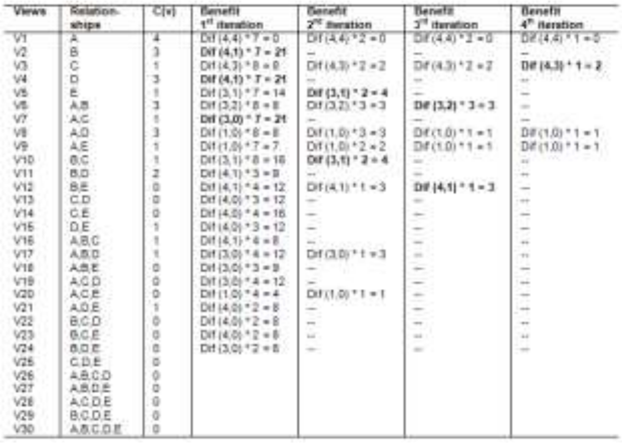


Resultados de la vista: viewABC

```
111
110
100
110
```

Dataset
A, B, C, D, E
1, 1, 1, 0, 0
1, 1, 0, 1, 0
1, 0, 0, 1, 1
1, 1, 0, 1, 0

Fig. 11. Test of a Tree-Dimension view Execution.



Views	Relation-ships	C(v)	Benefit 1st iteration	Benefit 2nd iteration	Benefit 3rd iteration	Benefit 4th iteration
V1	A	4	Dif (4,4) * 7 = 0	Dif (4,4) * 2 = 0	Dif (4,4) * 2 = 0	Dif (4,4) * 1 = 0
V2	B	3	**Dif (4,1) * 7 = 21**	–	–	–
V3	C	1	Dif (4,3) * 8 = 8	Dif (4,3) * 2 = 2	Dif (4,3) * 2 = 2	**Dif (4,3) * 1 = 2**
V4	D	3	**Dif (4,1) * 7 = 21**	–	–	–
V5	E	1	Dif (3,1) * 7 = 14	**Dif (3,1) * 2 = 4**	–	–
V6	A,B	3	Dif (3,2) * 8 = 8	Dif (3,2) * 3 = 3	**Dif (3,2) * 3 = 3**	–
V7	A,C	1	**Dif (3,0) * 7 = 21**	–	–	–
V8	A,D	3	Dif (1,0) * 8 = 8	Dif (1,0) * 3 = 3	Dif (1,0) * 1 = 1	Dif (1,0) * 1 = 1
V9	A,E	1	Dif (1,0) * 7 = 7	Dif (1,0) * 2 = 2	Dif (1,0) * 1 = 1	Dif (1,0) * 1 = 1
V10	B,C	1	Dif (3,1) * 8 = 16	**Dif (3,1) * 2 = 4**	–	–
V11	B,D	2	Dif (4,1) * 3 = 9	–	–	–
V12	B,E	0	Dif (4,1) * 4 = 12	Dif (4,1) * 1 = 3	**Dif (4,1) * 1 = 3**	–
V13	C,D	0	Dif (4,0) * 3 = 12	–	–	–
V14	C,E	0	Dif (4,0) * 4 = 16	–	–	–
V15	D,E	1	Dif (4,0) * 3 = 12	–	–	–
V16	A,B,C	1	Dif (4,1) * 4 = 8	–	–	–
V17	A,B,D	1	Dif (3,0) * 4 = 12	Dif (3,0) * 1 = 3	–	–
V18	A,B,E	0	Dif (3,0) * 3 = 9	–	–	–
V19	A,C,D	0	Dif (3,0) * 4 = 12	–	–	–
V20	A,C,E	0	Dif (1,0) * 4 = 4	Dif (1,0) * 1 = 1	–	–
V21	A,D,E	1	Dif (4,0) * 2 = 8	–	–	–
V22	B,C,D	0	Dif (4,0) * 2 = 8	–	–	–
V23	B,C,E	0	Dif (4,0) * 2 = 8	–	–	–
V24	B,D,E	0	Dif (3,0) * 2 = 6	–	–	–
V25	C,D,E	0				
V26	A,B,C,D	0				
V27	A,B,D,E	0				
V28	A,C,D,E	0				
V29	B,C,D,E	0				
V30	A,B,C,D,E	0				

Fig. 12. Output of Benefit Calculation by the Harinarayan Algorithm Implementation.

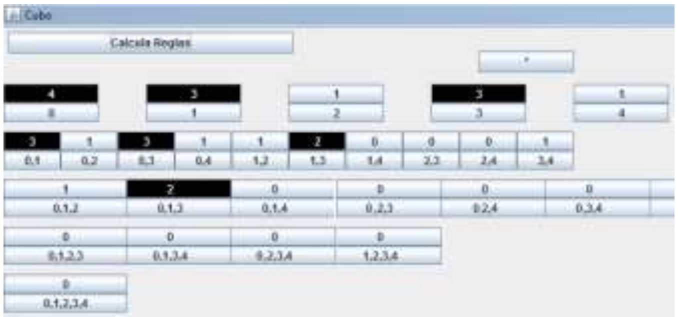
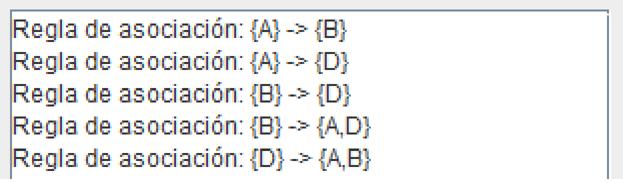


Fig. 13. Frequent Patterns Visualisation by Analytical Tool.



```
Regla de asociación: {A} -> {B}
Regla de asociación: {A} -> {D}
Regla de asociación: {B} -> {D}
Regla de asociación: {B} -> {A,D}
Regla de asociación: {D} -> {A,B}
```

Fig. 14. Output Program of Association Rules Calculation.

V. Conclusions

Among the tendencies that have been found in a revision of the techniques reported, they are not oriented toward the pedagogic aspect of teaching data science topics; on the contrary, they use data science as an automatic learning tool for other areas.

In the learning of data mining, the analysis of data cubes is a technique that has prevailed as an efficient form of data analysis. However, its manual design is a very challenge task to be carried out within large volumes of data, so that is why the automation of the processes through computational tools constitutes an excellent aid for the data analysts. In this sense, the presented tool helps automatise the data cube tasks plus the storage model in vectors/matrixes. Usage of the tree structure gives a natural indexation and provides an efficient extraction of the data thanks to the pre-structuring of the added data. All the advantages of automated calculation can be explained in work out session with easy examples to explore the method in various cases as well as testing the solutions for exercises.

The visual presentation and the interaction with the consequence of the changes could improve the understanding of data mining because it constitutes a reinforcement to the constructivism approach in education, that is why the tool is developed with a visual interface focused on a data analysis task. Even though the implemented algorithms are not the only ones that can be used to perform these tasks, they are considered as the basis for a well understanding of more complex proposals. In this preliminary results with students, they could explore the system capabilities to analyse their dataset being able to obtain results in less time and effort than manually, as well as obtain a new data warehouse in MySQL for future tests.

VI. Future Work

Even though this tools us completely functional, some improvements could be made. Firstly, testing the tool in control and observation student groups to have feedback of the student and learn about the effect of this digital resources in a virtual class. Besides, adding more functions and embedding the description of the processes in the interface could made this tool a self-learning tool.

In the technical approach, if the memory is limited then, the structure's baseload could be accelerated via data chunks, and it could improve the time consumption for large datasets. Moreover, a further study of alternative algorithms for the data cube creation algorithm using the tree structure could be implemented and could help the students to compare the performance in their practice in this learning tool.

References

[1] Mohanty S., Jagadeesh M., Srivatsa H, "Big Data" in the Enterprise. In: Big Data Imperatives. Apress, Berkeley, CA, 2013.

[2] A. B. Urbina, & De la Calleja, J., "Brief review of educational applications using data mining and machine learning", Revista Electrónica de Investigación Educativa, 19(4), 84-96. https://doi.org/10.24320/redie.2017.19.4.1305, 2017.

[3] P. Vijayakumar, S. M. Ganesh, L. J. Deborah, and B. S. Rawal.: A new SmartSMS protocol for secure SMS communication in m-health environment, Comput. Electr. Eng., vol. 65, pp. 265–281, 2018.

[4] Y. Kazemi and S. A. Mirroshandel.: A novel method for predicting kidney stone type using ensemble learning, Artif. Intell. Med., vol. 84, pp. 117–126, 2018.

[5] M. Echeverría, A. Jimenez-Molina, and S. A. Ríos.: A semantic framework for continuous u-health services provisioning, Procedia Comput. Sci., vol. 60, no. 1, pp. 603–612, 2015.

[6] U. R. Acharya et al.: Data mining framework for breast lesion classification in shear wave ultrasound: A hybrid feature paradigm, Biomed. Signal Process. Control, vol. 33, pp. 400–410, J. Wang, R. Boesch, and Q. X. Li.: A case study of air quality - Pesticides and odorous phytochemicals on Kauai, Hawaii, USA, Chemosphere, vol. 189, pp. 143–152, 2017.

[7] Q. Wang, J. Wang, M. Z. He, P. L. Kinney, and T. Li.: A county-level estimate of PM2.5related chronic mortality risk in China based on multi-model exposure data, Environ. Int., vol. 110, no. February 2017, pp. 105–112, 2018.

[8] D. Uni and I. Katra.: Airborne dust absorption by semi-arid forests reduces PM pollution in nearby urban environments, Sci. Total Environ., vol. 598, pp. 984–992, 2017.

[9] M. A. Bari and W. B. Kindzierski.: Ambient fine particulate matter (PM2.5) in Canadian oil sands communities: Levels, sources and potential human health risk, Sci. Total Environ., vol. 595, pp. 828–838, 2017.

[10] K. R. Malik, Y. Sam, M. Hussain, and A. Abuarqoub.: A methodology for real-time data sustainability in smart city: Towards inferencing and analytics for big-data, Sustain. Cities Soc., vol. 39, no. April, pp. 548–556, 2018.

[11] Sushil S. Chaurasia, Devendra Kodwani, Hitendra Lachhwani, Manisha Avadhut Ketkar, “Big data academic and learning analytics: Connecting the dots for academic excellence in higher education”, International Journal of Educational Management, ISSN: 0951-354X, 2018.

[12] Siemens George, "Learning Analytics: The Emergence of a Discipline", Volume: 57 issue: 10, page(s): 1380-1400, 2013.

[13] L. Zhang and J. Wen.: A systematic feature selection procedure for short-term data-driven building energy forecasting model development, Energy Build., vol. 183, pp. 428–442, 2019.

[14] F. Wang and J. Liang.: An efficient feature selection algorithm for hybrid data, Neurocomputing, vol. 193, pp. 33–41, 2016.

[15] Y. Lin, H. Wang, S. Zhang, J. Li, and H. Gao.: Efficient quality-driven source selection from massive data sources, J. Syst. Softw., vol. 118, pp. 221–233, 2016.

[16] Z. Manbari, F. AkhlaghianTab, and C. Salavati.: Hybrid fast unsupervised feature selection for high-dimensional data, Expert Syst. Appl., vol. 124, pp. 97–118, 2019.

[17] V. Harinarayan, A. Rajaraman, and J. D. Ullman.: Implementing data cubes efficiently, SIGMOD, 1996.

[18] A. Peña-Ayala.: Educational data mining: A survey and a data mining-based analysis of recent works, Expert Syst. Appl., vol. 41, no. 4 PART 1, pp. 1432–1462, 2014.

[19] Stefan Slater, Srećko Joksimović, Vitomir Kovanovic, Tools for Educational Data Mining: A Review, Journal of Educational and Behavioral Statistics 42(1):85-106 · January 2017.

[20] Muyesser Eraslan Yalcin, Birgul Kutlu , “Examination of students’ acceptance of and intention to use learning management systems using extended TAM”, Volume 50, Issue 5, pp. 2414-2432, 2019.

[21] Sara Landset, Taghi M. Khoshgoftaar, Aaron N. Richter and Tawfiq Hasanin, “A survey of open source tools for machine learning with big data in the Hadoop ecosystem”, Landset et al. Journal of Big Data, 2015.

[22] S. Slater, S. Joksimovic, V. Kovanovic, R. Baker, and D. Gasevic.: Tools for educational data mining : a review,” January, 2017.

[23] Sonam Yadav, “Open Source Big Data Databases Tools”. PCQuest; Gurgaon, 2016.

[24] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H., “The WEKA data mining software: an update”, SIGKDD Explorations, 11(1), 2009.

[25] Demsar, J., Curk, T., Erjavec, A., Gorup, C., Hocevar, T., Milutinovic, M., et al., “Orange: data mining toolbox in python”, Journal of Machine Learning Research 14, 2349-2353, 2013.

[26] Yichang, “Data Mining method based on Lattice”, 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), China, 2012.

[27] Karuna, Gupta G. “Dynamic Implementation Using Linked List”, International Journal Of Engineering Research & Management Technology, Volume 1, Issue-5, pp. 44-48, 2014.

[28] Petra G., Miroslav B., “Analysis of B-tree data structure and its usage in computer forensics”, Conference: Central European Conference on Information and Intelligent Systems, 2010.

[29] Aouiche K., Darmont J., “Data Mining based materialised view and index selection in data warehouses”, Journal of Intelligent Information Systems Vol. 33, DOI: 10.1007/s10844-009-0080-0, 2007.

[30] J. Han, J. Pei, G. Dong, and K. Wang.: Efficient computation of Iceberg cubes with complex measures, ACM SIGMOD Rec., vol. 30, no. 2, pp. 1–12, 2005.

[31] Jiawei H., Micheline K., “Data Mining Concepts and Techniques”, 2nd Edition, Ed. Morgan Kaufmann Publishers, pp. 230, 2006.