Vol. 1, No. 1, pp 1 - 21, 2020

PARADIGM Plus

Evaluation of the Bias in the Management of Patient's Appointments in a Pediatric Office

Cinthia Vegega[1,✉], Pablo Pytel[1], and María Florencia Pollo-Cattaneo [1]

[1]Information System Methodologies Research Group,
Universidad Tecnológica Nacional Facultad Regional Buenos Aires. Argentina
cinthiavg@yahoo.com.ar, {ppytel,flo.pollo}@gmail.com

Abstract

The application of Machine Learning algorithms must always take into account the objectives set within the project, the characteristics of the domain where the project will be carried out and the data available to use. Given this, it is essential before collecting data considered as representative of the problem to be solved, because otherwise there may be hidden biases in the data and these may solve a different problem from the one intended. In this context, the aim of this work is to apply a process based on the Gridding method that allows the analysis of the features of the data to be used. This process is applied to the historical data of a pediatric medical office where it is sought to implement an intelligent system that allows to predict the number of normal and over-shift appointments for a particular date and time, since it is desired to hire, when necessary, another pediatric doctor to assist in the care of patients.

Keywords: Intelligent Systems · Machine Learning · Training Data · Repertory Grid · Bias

Received: 23 February 2020 · Accepted: 1 April 2020 · Published: 9 April 2020.

1 Introduction

Machine Learning is considered to be the discipline that studies and models learning processes with their multiple manifestations in order to be able to be transferred to computers [1, 2]. The aim is to reproduce aspects of the behavior of intelligent beings to facilitate the implementation of software systems in complex environments [3] by building models from the observed data [4]. The software systems thus generated are called Intelligent Systems [5, 6] and can be applied in different domains to solve different types of problems. Since the beginning of this century, its use has spread rapidly and it is often a more attractive alternative than building models manually, although "Automatic Learning is not magic, you cannot get something out of nothing" [7], since in order to apply any of the algorithms it is essential to have data. One of the types of problems where Machine Learning has been applied has to do with the construction of models capable of inferring from historical data the dependencies between the past value and the short-term future, which are the so-called Predictive Models. As indicated [8], predicting the future is one of the most important and difficult tasks in the applied sciences. One of the main reasons is that Predictive Models start from the behavior of historical data, so they must assume that the future will be the same (or similar) to the past [9]. In the event that some behavior changes, the generated model will become inefficient. Therefore, it is essential to have a large amount of data representative of the domain of the problem that allows the model to generate better predictions. Likewise, when the data are not representative enough, they are said to have a Bias. This term is used to speak of something biased or twisted [10], so it is used in psychology to refer to the tendency or prejudice of a person to perceive and interpret reality, thus generating a distortion of it. In Statistics, Bias is used to refer to the difference between the value generated by the model and the expected value [11]. In the context of Predictive Models something similar happens. If the data presents a bias, then there is a risk of generating an Intelligent System that is not based on reality and produces erroneous results [7]. In other words, you could be training the algorithms to solve a different problem from the one you want to solve.

For this reason, it is essential to know in advance the biases associated with the data and the intelligent system so that they can be understood by future users to avoid misunderstandings and situations of discrimination [12]. It is not rare that developers with the best intentions can involuntarily produce intelligent systems with prejudicial results, because even they may not understand enough of the problem, its context and the data to prevent unintended results. The worst thing about this scenario is that the bias may be so subtle that it goes undetected during testing. If this system is then put into operation and users come to rely blindly on the long-term results, it could lead to situations of sexism, racism and other forms of discrimination [13].

In this context, an adaptation of the Repertory Grid has been proposed in [14] to evaluate the collected data with the assistance of the available domain experts and determine whether it can be used. Likewise, in [15] the proposed method has been applied to a higher education institution course data to predict the students' performance throughout the course and identify their strengths and weaknesses. In this work, the method is applied to another case study for the management of patient's appointments in a pediatric office in order to evaluate whether they are sufficiently representative to implement a future Intelligent System. For this purpose, section 2 presents the problem identification, section 3 describes the Repertory Grid technique and section 4 describes the data gathered and its context, whose evaluation is dealt with in section 5. Section 6, implements a prototype of the intelligent system based on the data evaluated. Finally, section 7 presents the conclusions and the future line of work.

2 Problem Identification

Obtaining the data for the construction of Predictive Models is not a trivial task. For example, take the case of determining the amount of historical information needed to obtain the best results [16]. According to [17], the usual answer to the question "how much data are needed" is "as much as possible". The more data you have, the better you will be able to identify the structure of the model and the patterns used for prediction, but in practice it is essential to put some limits on it. Although there are publications [18, 19] that indicate minimum requirements for the amount of data to be applied, these are considered over-simplified because they ignore aspects such as the underlying random variability of the data. Therefore, in order to define the amount of data to be used, it is first necessary to identify the available data sources and understand their characteristics. Only then, it will be possible to collect data that fully represent the problem to be solved [20]. The same happens when determining the appropriate attributes to be selected as input and output variables of the model [16]. To this end, in addition to analyzing the sources of historical information, the characteristics of the domain, the objectives set and, often, the technology used for the construction of the Predictive Model must also be considered. It may also happen that the historical data available in computerized repositories are not enough, so other sources should be used, such as the opinion of experts in the field [17]. Likewise, when the data is not sufficiently representative and presents a bias, there is a risk of generating an Intelligent System that is not based on reality and produces erroneous results [7]. In other words, you could be training the algorithms to solve a different problem from the one you want to solve. However, it is not true that this is always a bad thing. According to Tom Mitchell's principle of "the futility of bias-free learning" [21], biases are necessary for algorithms to work. By removing them it may seem that a desirable goal is achieved, but in truth the result becomes virtually useless since an 'unbiased' Intelligent System loses the ability to generalize and process new examples. So, although in ordinary life prejudice is a pejorative word, since preconceived notions are bad, in this case, preconceived notions are absolutely necessary for the algorithm to learn [22].

In any case, as mentioned in the previous section, it is essential to know in advance the biases associated with the data and the Intelligent System so that they can be understood by future users to avoid misunderstandings and discrimination situations [12]. An illustrative example of this situation is Beauty AI [23], which has been publicized as the "first beauty contest judged by robots" and to which approximately 6,000 women from more than 100 countries around the world have registered. According to its creators, this intelligent application seeks to eliminate the prejudices of human jurors by using objective factors of a contestant's image (such as the amount of wrinkles and facial symmetry) to identify the most attractive ones. But, according to [24], at the end of the competition the result was not as expected: of the 44 winners, practically all were white (very few were of Asian origin, and only one had dark skin), although the developers claim that they did not consider skin color as a sign of beauty. This is because during the training the algorithm had a large number of photos of Caucasian women but very few of women of color, that is, there was a significant bias in the data. To solve this problem, the application had to be retrained, including a greater variety of photos and generating a new version.

Another case, more serious, is the software applied in the 'predict policing' initiative [25], which seeks to prevent crime by determining where the police should patrol according to the prediction of the occurrence of future crimes. According to its critics [26], many of the predictions generated by the algorithms tend to fail because the data provided by the police is incomplete, partial and/or erroneous [27]. In addition, the number of failures increase as the system is used. Consider the case in which police officers detect and stop a minor crime in the area they were sent to patrol, which according to [28] will always happen because the police tend to think that there will be a crime and therefore suspect of everyone in the area. The crime data recorded will then be used by the algorithm to increase the risk of the area. In the long run, this could result in more patrol assignments to that area leaving others that may actually be at similar or higher risk with few or no patrols. Something similar has been detected about software used to assess the risk of recidivism in offenders [29]. In [30], it was found that a black defendant was twice as likely to commit future crimes and was therefore assigned a high risk, while white defendants were considered to be low risk. Since this information is used by the judge to determine whether they were granted parole, it has generated a significant disproportion in the prison population.

One of the main issues in the implementation of Intelligent Systems has to do with the steps associated with collecting the data, integrating, cleaning and pre-processing it. It is therefore essential to identify and understand them first [31]. In [12], the need for Intelligent Systems to avoid undesired behavior and generate evidence that involuntary failures are unlikely is emphasized. To this end, it is essential to have "transparency" not only about the data and algorithms involved, but also in explaining how the results have been generated. Therefore, progress must be "continued" by making AI (in general) and Machine Learning (in particular) a mature field of engineering so that they can create "predictable, reliable, robust and safe systems".

Consequently, this work aims to assist the developers of an Intelligent System, at the beginning of the project, in identifying sources of bias in the data and thus reducing their impact. In the initial tasks of these projects, work is done on the steps associated with collecting the data, integrating it, cleaning it up and pre-processing it. To do this, it is essential to identify and understand the data first [31], which is not a trivial task. The sources of historical information must be analyzed, and the characteristics of the domain, the objectives set and the expectations of users must also be considered. In addition, historical data available in computerized repositories may be insufficient, so other sources, such as expert opinion, should also be used [17, 22]. However, experts are usually unable or unwilling to provide their knowledge, and more effective indirect techniques are needed to educate and represent that knowledge [6]. For this reason, this proposal considers the possibility of applying a semi-automatic procedure to generate a representation of the available data and allow experts to analyze and interpret it more easily.

3 Repertory Grid

The Repertory Grid technique has been defined by psychologist George Kelly in [32] and is based on the Theory of Personal Constructors. A construct is, according to the RAE [33], a "bipolar descriptive category" that people use to organize "data and experiences from their world", which Kelly considers as "a way in which two or more things are similar and therefore different from a third or fourth thing" [34]. So, "personal constructs" are characteristics of objects or elements that allow a person to make judgments [35] and interpret the world around him.

In this sense, the Grid applies these bipolar characteristics to generate an objective representation of the mental image by which a person distinguishes between similar and different elements [36]. To this end, it seeks to "go beyond words" [34] by processing the subjective data provided by the person through simple operations [37].

The basic steps of a Grid are divided into the following five stages [34, 36, 38]:

1° ***Identification of the Elements:*** seeks to identify a homogeneous and representative set of conceptual elements within each category involved in knowledge.

2° ***Identification of Characteristics:*** determines the list of bipolar characteristics that can be attributed to the elements identified above.

3° ***Design of the Grid:*** with the elements and characteristics identified, a two-dimensional matrix (the "grid") is generated, where the elements are located in the columns and the characteristics in the rows. For each element/characteristic intersection, the person must enter a numerical value, and there are three ways to construct a grid:

- *Dichotomous grid:* the values that are assigned are binary, depending on whether the element has (1) or not (0) each characteristic.
- *Classification grid:* it is assigned the logical position or ranking that an element has on each characteristic.
- *Evaluation grid:* the values are defined as the degree of satisfaction with which the element in question covers each characteristic.

4° ***Formalization:*** with the values assigned in the grid, it is proceeded to analyze the relationships between the Elements and the Characteristics independently of each other:

- *Classification of the Elements:* it is a matrix of distances between the elements is generated using the Manhattan distance formula [39] with the values of each pair of columns. With these distances, the elements start to be grouped, always taking the least distant ones (that is, with the smallest value). When all the elements are grouped, each of the groups will be represented with an ordered tree.
- *Classification of the characteristics:* since the characteristics are bipolar, two types of distance must be calculated: distance 1 using the Manhattan distance between the rows of the grid and, distance 2, using the Manhattan distance between one row of the grid and another row of its opposite (in which the complements of the grid values are assigned). Then, these distances are unified by taking the smallest value between each combination of characteristics. Finally, the same steps as for the Element Classification are applied, generating the corresponding ordered tree.

5° ***Analysis of the Results:*** with the ordered trees already generated, it is proceeded to carry out their corresponding analysis and interpretation to determine the number of groups and the similarities between them. Finally, these results are then presented and discussed together with the person who provided the information for the Grid. The application of these steps can be seen in [38].

Although Repertory Grid was originally defined as a means to help mentally ill people become aware of inconsistencies in their own scales of values [34, 38], it has subsequently been successfully applied in other domains [40], such as Computer Science, Marketing, Business Administration, Engineering and Tourism, among others [41]. In this sense, the acquisition of knowledge for the construction of Expert Systems is significant [32, 42]. Since experts in the field feel more comfortable (and more precise) when they are able to use their own terminology, Repertory Grid allows for the acquisition of knowledge in a more natural and flexible way than other techniques [35]. In addition, since a grid can analyze any topic, knowledge can be acquired on a wide range of subjects.

Another application can be found in the "Theorise-Inquire" technique [43], where the Grid is used to evaluate experts' "hunches" about the quality of available data sources. Thus, it is possible to test whether the information registered in the database is in accordance with the expectations of the domain experts (i.e. complete and without errors). While these objectives are similar to those of the procedure proposed above, when looking at an example of this technique in [44], on the analysis of the databases of a large retail chain in the United Kingdom, it can be seen that it works differently. In "Theorise-Inquire", Repertory Grid is used in its traditional sense, i.e. to "make explicit the knowledge of the experts and then use that knowledge to direct the data analysis" [43]. In other words, it is the experts who provide the values of the grid corresponding to the situations they have experienced (elements) and their distinctions from each other (characteristics).

The results of this grid are then associated with the available data so that the experts' theories can be confirmed. In contrast, in the procedure proposed above, the grids are generated directly from the data collected and then the results generated by the formalization of the grid are contrasted with the experts' view.

4 Context Case Study and Collect Data Description

The case study is conducted within a pediatric medical office, which sees children from birth to their teens on their 18th birthday. Office appointments are provided online, with the exception of prenatal or first-time telephone consultations. Despite not having an on-call service, unscheduled consultations from their own patients are often attended to. Appointments for such consultations (called "over-shifts") are granted by telephone on the same day as they are required. Such over-shifts appointments are discharged by the secretary the same day they are requested, since if an over-shift is required, the office must be called to check availability. The over-shifts appointments are registered between the normal appointments. For example, between the 9:00 a.m. and 9:30 a.m. normal appointments there may be an over-shift appointment at 9:15 a.m.

In this context, the aim is to implement an intelligent system that allows for the prediction of the number of normal and over-shift appointments for a particular date and time, given that it is desired to hire, when necessary, another pediatrician to assist in the care of the patients. For this reason, it is considered essential that the system provides results that are consistent with the management of office appointments. In addition to collecting the historical data that will be applied to build the intelligent system, it is necessary to identify the general characteristics of the domain where the prediction is made. Moreover, it is essential to detect situations or events to be considered for which there is no data or the data available is not representative. Otherwise, the system could generate incorrect predictions, leading the physician to make wrong decisions.

As for the data collected, the appointments are granted from November 2017 to date were obtained. Previously, the management of the appointments was always in charge of the secretary of the office, who from a phone call of the patient (in this case, mother, father or guardian, being a minor), recorded the appointment in a paper agenda. For this reason, as of November 2017, a computer system is being used where the patient can obtain his/her appointment directly through a web page with the additional option of being able to cancel it.

The data was extracted directly from the computer system database in a comma-delimited text file (CSV) containing the following ten attributes:

- *DATE*: Full appointment date, the format of which is dd/mm/yyyy (day/month/year).
- *HOUR*: Appointment time, with format hh:mm (hour:minutes).
- *ID_PATIENT*: Patient's identifier (numerical data type).
- *SURNAME*: Patient's last name.
- *NAME*: Patient's name.
- *ID-SPECIALIST*: Identifier of the pediatrician treating the patient (type of numerical data)
- *SPECIALIST*: Name of attending pediatrician.
- *BOX*: Number of the office where the patient is treated.
- *OBSERVATIONS*: Observations that the patient can add when booking the appointment.
- *STATUS*: Appointment status (type of numerical data) The possible states are the following:
  - 1: "available"
  - 2: "booked"
  - 3: "waiting"
  - 4: "attending"
  - 5: "finished"
  - 6: "suspended"

As for the status of an appointment, its management starts with the creation of the appointment, which is in charge of the office secretary. When an appointment is created, it is created with "available" status. Patients log in to the office website and book their appointments at their convenience. Each appointment that is booked by a patient is set to "booked" status.

Once the patient arrives at the office on the day his or her appointment is booked, the secretary records that the patient has arrived, so the appointment goes to "waiting" status. When the pediatrician attends to the patient, the appointment changes to "attending" status and at the end it changes to "finished" status. At the end of each day, if there are appointments that became available because patients did not show up or were not booked, the secretary records this situation and those appointments are passed to "suspended". The patient may cancel an appointment at any time. This is done through the appointment notification email that is sent to the patient's personal email address. When patients cancel their appointments, the appointments return to "available" status.

5 Results of the Application of the Repertory Grid Method

From the available data obtained for the case study, it is analyzed if these data are representative of the business to build the intelligent system, that is, if the data are sufficient and appropriate to consider all the possible situations of the problem to be solved. This is done by applying the method proposed in [14]. This process is based on the Repertory Grid technique described previously in section 3, but presents some differences with respect to the original technique. The main difference is that the grids are generated directly from the data collected and then the results generated by their formalization are contrasted with the vision of the experts or stakeholders in the business.

In addition, three grids are used: one corresponding to the *Elements* (which allows the evaluation of already known classes) and, two for the *Characteristics* (one direct and one opposite, to evaluate the rest of the attributes). Therefore, it is necessary to first carry out a set of tasks related to the collection and preparation of data to generate the grids, which will then be processed to generate the trees to be interpreted. The results to evaluate the data are presented below.

- **Phase A: Design of the Grids**
  - **Activity A.1- Data Preparation**

    Data preparation tasks including formatting, cleaning and integration of data are carried out. In this context, the following decisions are taken according to each attribute available in the CSV file:

    - *BOX*: is not taken into account as it always takes the same value.
    - *OBSERVATIONS*: is not taken into account because it is empty in all records.
    - *ID_PATIENT*: cannot be used for confidentiality.
    - *SURNAME*: cannot be used for confidentiality.
    - *NAME*: cannot be used for confidentiality.
    - *SPECIALIST*: is not taken into account as it is considered to be redundant with the specialist's identifier, so only the *ID-SPECIALIST* attribute is used.
    - *STATUS*: since appointment records have been found that have not been assigned a final status ("FINISHED" or "SUSPENDED"), it is adjusted as follows:
      - For the 94 records with "booked" status, the patient is considered not to have attended the appointment, so they are moved to the SUSPENDED status.
      - In contrast, the 40 records with "waiting" status and the 94 records with "attending" status are considered to have been attended to by the specialist and are therefore changed to the FINISHED status. Both decisions were consulted with the physician responsible for the office.
    - *DATE*: this attribute is split in two: *DATE-MONTH* (containing the month number) and *DATE-DAY* (that contains the day number). The year of the date is not considered, since the prediction will be made in the future. A new attribute *APPOINTMENT-TYPE* is created to indicate whether or not it is an over-shift, assigning the values of "APPOINTMENT" and "OVERSHIFT-APPOINTMENT". This attribute is calculated considering if there is another appointment for the same specialist with the same day, month, year, hour and a difference of minutes less than 15 (decision consulted with the office secretary).
    - A new *SCHEDULE* attribute is created to indicate whether it is morning, midday or evening appointment.

    This generates a file containing the following seven attributes: *DATE-MONTH*, *DATE-DAY*, *ID-SPECIALIST*, *SCHEDULE*, *STATUS*, *APPOINTMENT-TYPE*, of which the first four are numerical and considered as input attributes and the last two are combined to form the target attribute. As a result of this conjunction. the values: "APPOINTMENT-FINISHED", "OVERSHIFT-APPOINTMENT-FINISHED", "APPOINTMENT-SUSPENDED" and finally "OVERSHIFT-APPOINTMENT-SUSPENDED" are obtained.

  - **Activity A.2- Data Segmentation**

    In this task, the numerical attributes are segmented. The data is imported into the Tanagra tool [45] and the Kohonen SOM neural network [19] are used. With a distribution of 3x3 neurons (i.e. with a maximum requested of 9 clusters), a distribution has been reached that has a cluster with less than 10 tuples ("c_som_2_3"), as can be seen in the sheet "Res Kohonen" (Result of Kohonen)[1]. Also, a new attribute called *Cluster_SOM_1* has been added to the data indicating the cluster identifier assigned to each tuple. This new structure, which is included in the sheet "Datos Segmentados" (Segmented Data), will be used in subsequent tasks. Table 1 shows the number of tuples per cluster.

Table 1: Number of Tuples Assigned per Cluster

Cluster	Cluster ID	Number of Tuples
Nº 1	c_som_1_1	1397
Nº 2	c_som_1_2	1125
Nº 3	c_som_1_3	1354
Nº 4	c_som_2_1	1891
Nº 5	c_som_2_2	1053
Nº 6	c_som_2_3	0
Nº 7	c_som_3_1	995
Nº 8	c_som_3_2	1500
Nº 9	c_som_3_3	21

- **Activity A.3- Design of the Elements Grid**

Using the segmented data obtained in the previous task, the Element Grid is generated. It is a 4x9 matrix since it has four columns corresponding to the values of the class with values "APPOINTMENT-FINISHED", "OVERSHIFT-APPOINTMENT-FINISHED", "OVERSHIFT -APPOINTMENT-SUSPENDED", and "APPOINTMENT-SUSPENDED", as well as, nine rows corresponding to the clusters "c_som_1_1", "c_som_1_2", "c_som_1_3", "c_som_2_1", "c_som_2_2", "c_som_2_3", "c_som_3_1", "c_som_3_2", and "c_som_3_3"). To complete the matrix values, the steps proposed in [14] are applied. To illustrate these steps, the definition of the value V_{12} corresponding to class "OVERSHIFT-APPOINTMENT-FINISHED" and cluster "c_som_1_1" is presented below as an example:

1° The number of tuples is determined in the segmented data for the cluster "c_som_1_1" and class "OVERSHIFT-APPOINTMENT-FINISHED", which is equal to 706.

2° The total number of tuples is defined for "OVERSHIFT-APPOINTMENT-FINISHED" class that is equal to 4002.

3° The membership percentage is calculated using equation 1.

$$P_{12} = \frac{T_{12}}{\sum_{i=1}^{9}(T_{i2})} = \frac{706}{4002} = 0.1764 \tag{1}$$

4° The percentage obtained using equation 2 is formatted.

$$V_{12} = \text{Rounding up}\left[P_{12} \cdot 10\right] = \text{Rounding up}\left[1.764\right] = 2 \tag{2}$$

5° Then, 2 is recorded as the V_{12} value in the matrix corresponding to the class and the cluster.

In the same way, the rest of the values are completed, as can be seen in the sheet "Parrilla de Elementos" (Elements Grid), obtaining as a result the grid shown in Table 2.

[1] https://github.com/inteligenciaartificialutn/Paradigm

Table 2: Elements Grid

	APPOINTMENT -FINISHED	**OVERSHIFT-APPOINTMENT -FINISHED**	**OVERSHIFT-APPOINTMENT -SUSPENDED**	**APPOINTMENT -SUSPENDED**
c_som_1_1	1	2	1	1
c_som_1_2	1	1	2	1
c_som_1_3	2	1	1	2
c_som_2_1	2	2	2	2
c_som_2_2	1	1	2	2
c_som_2_3	0	0	0	0
c_som_3_1	1	1	0	1
c_som_3_2	2	2	1	1
c_som_3_3	0	0	0	0

- **Activity A.4- Attribute Weighting**
  In this task, the segmented data of activity A.2 are again used, using the Tanagra tool [45] to perform the weighting of the interdependence between the generated clusters and the data attributes. This is done by generating conditional probabilities through the Näive Bayes algorithm, as can be seen in the sheet "Res NaiveBayes" (Result of Näive Bayes).
- **Activity A.5- Design of the Characteristics Grids**
  From the conditional probability tables obtained from the previous task, the two Characteristics Grids (Direct and Opposite) are generated. These have a 9x4 structure since there are nine cluster values indicated as columns, and 4 attributes as rows. As you can see in the sheet "Parrilla de Caracteristicas" (Characteristics Grid), we proceed in this way with all the combinations obtaining the grids indicated in Tables 3 and 4.

Table 3: Direct Characteristics Grid

	c_som_1_1	**c_som_1_2**	**c_som_1_3**	**c_som_2_1**	**c_som_2_2**	**c_som_2_3**	**c_som_3_1**	**c_som_3_2**	**c_som_3_3**
DATE-MONTH	0	1	10	10	0	0	0	10	10
DATE-DAY	9	0	6	10	10	0	0	0	3
D-SPECIALIST	10	10	10	10	10	0	10	10	0
SCHEDULE	0	4	7	0	4	0	0	0	0

– **Phase B: Formalization and Analysis of the Grids**

- **Activity B.1- Classification of the Elements**
  From the Element Grid obtained in activity A.3, the distances between the columns (i.e., the elements or classes) are calculated using the Manhattan distance [39] over the values in Table 2. In this way, the element distance Matrix shown in Table 5 is obtained. Since the absolute differences between the classes are considered, only the distances above the diagonal are indicated because the distances below it are equal.

Table 4: Opposite Characteristics Grid

	c_som_1_1	c_som_1_2	c_som_1_3	c_som_2_1	c_som_2_2	c_som_2_3	c_som_3_1	c_som_3_2	c_som_3_3
DATE-MONTH	10	9	0	0	10	0	10	0	0
DATE-DAY	1	10	4	0	0	0	10	10	7
D-SPECIALIST	0	0	0	0	0	0	0	0	10
SCHEDULE	10	6	3	10	6	0	10	10	10

Table 5: Matrix of distances between elements

	APPOINTMENT -FINISHED	OVERSHIFT-APPOINTMENT -FINISHED	OVERSHIFT-APPOINTMENT -SUSPENDED	APPOINTMENT -SUSPENDED
APPOINTMENT -FINISHED	-	2	5	2
OVERSHIFT-APPOINTMENT -FINISHED	-	-	5	4
OVERSHIFT-APPOINTMENT -SUSPENDED	-	-	-	3
APPOINTMENT -SUSPENDED	-	-	-	-

From this distance matrix, some groupings are made with the minimum distance criterion. In this case, the minimum distance is 2; then, the classes "APPOINTMENT-SUSPENDED", "APPOINTMENT-FINISHED", and "OVERSHIFT-APPOINTMENT-FINISHED" have been joined. Later, the distance matrix is updated, as shown in Table 6. In this way, only the "OVERSHIFT-APPOINTMENT-SUSPENDED" and the "[APPOINTMENT-FINISHED,

Table 6: Matrix of distances between elements after the first cluster

	OVERSHIFT-APPOINTMENT -SUSPENDED	[APPOINTMENT-FINISHED, APPOINTMENT-SUSPENDED, OVERSHIFT-APPOINTMENT-FINISHED]
OVERSHIFT-APPOINTMENT -SUSPENDED	-	3
[APPOINTMENT-FINISHED, APPOINTMENT-SUSPENDED, OVERSHIFT-APPOINTMENT-FINISHED]	-	-

APPOINTMENT-SUSPENDED, OVERSHIFT-APPOINTMENT-FINISHED]" group needs to be joined to finish (the graphic representation of this is shown in activity B.3).

- **Activity B.2- Classification of the Characteristics**

  In a similar way to the previous task, the distance matrix is generated for the attributes or characteristics of the grids obtained in task A.5. However, since there are two grids, a Direct one associated to the first pole of the attributes and an Opposite one associated to the second pole, in this case for each combination it will be necessary to calculate two distances (d1 and d2). In this way, the Distance Matrix d1 and d2 of the Characteristics is obtained, which can be seen on the sheet "Distancias de Caracteristicas" (Characteristics Distances). As it can be seen, the values of d1 are located above the diagonal and those of d2 below it. Since a single distance is necessary to make the groupings, d1 and d2 must be unified, taking the lowest value for each combination. This means that, for each combination of attributes, the minimum is taken between the value below and above the diagonal.

  Then, the values below the diagonal are discarded, thus obtaining the Unified Distance Matrix (included in the same sheet of the spreadsheet). Finally with the unified matrix, the corresponding groupings are carried out with the minimum distance criterion as it was done with the elements in the previous task (the graphic representation of this is shown in activity B.3).

- **Activity B.3- Interpretation of the Results**

  From the groupings obtained in the tasks B.1 and B.2, both the Ordered Tree of Elements shown in Figure 1 and the Ordered Tree of Characteristics in Figure 2 are generated, which are also available in the sheet "Interpretación Elementos" (Interpretation of Elements) and the sheet 'Interpretación Características' (Interpretation of Characteristics). The interpretation of these trees is included below:

  - Analysis of the Ordered Tree of Elements: Two groups have been generated in this tree. Thus, both suspended normal appointments ("APPOINTMENT-SUSPENDED") and finished normal appointments ("APPOINTMENT-FINISHED"), as well as the finished over-shift appointments ("OVERSHIFT-APPOINTMENT-FINISHED") have a more similar behavior than the suspended over-shift appointments ("OVERSHIFT-APPOINTMENT-SUSPENDED").
  - Analysis of the Ordered Tree of Characteristics: In this tree 3 groups have been generated. As a first grouping, the specialist physician (ID-SPECIALIST) is related to the schedule in which he attends (SCHEDULE). Then, this grouping is related to the month (DATE-MONTH) and finally, to the day of the appointment (DATE-DAY).

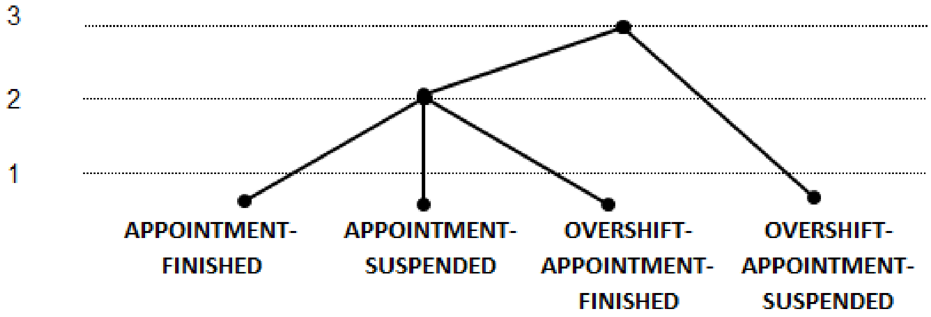
3
2
1
APPOINTMENT-FINISHED
APPOINTMENT-SUSPENDED
OVERSHIFT-APPOINTMENT-FINISHED
OVERSHIFT-APPOINTMENT-SUSPENDED

Figure 1: Ordered Tree of Elements

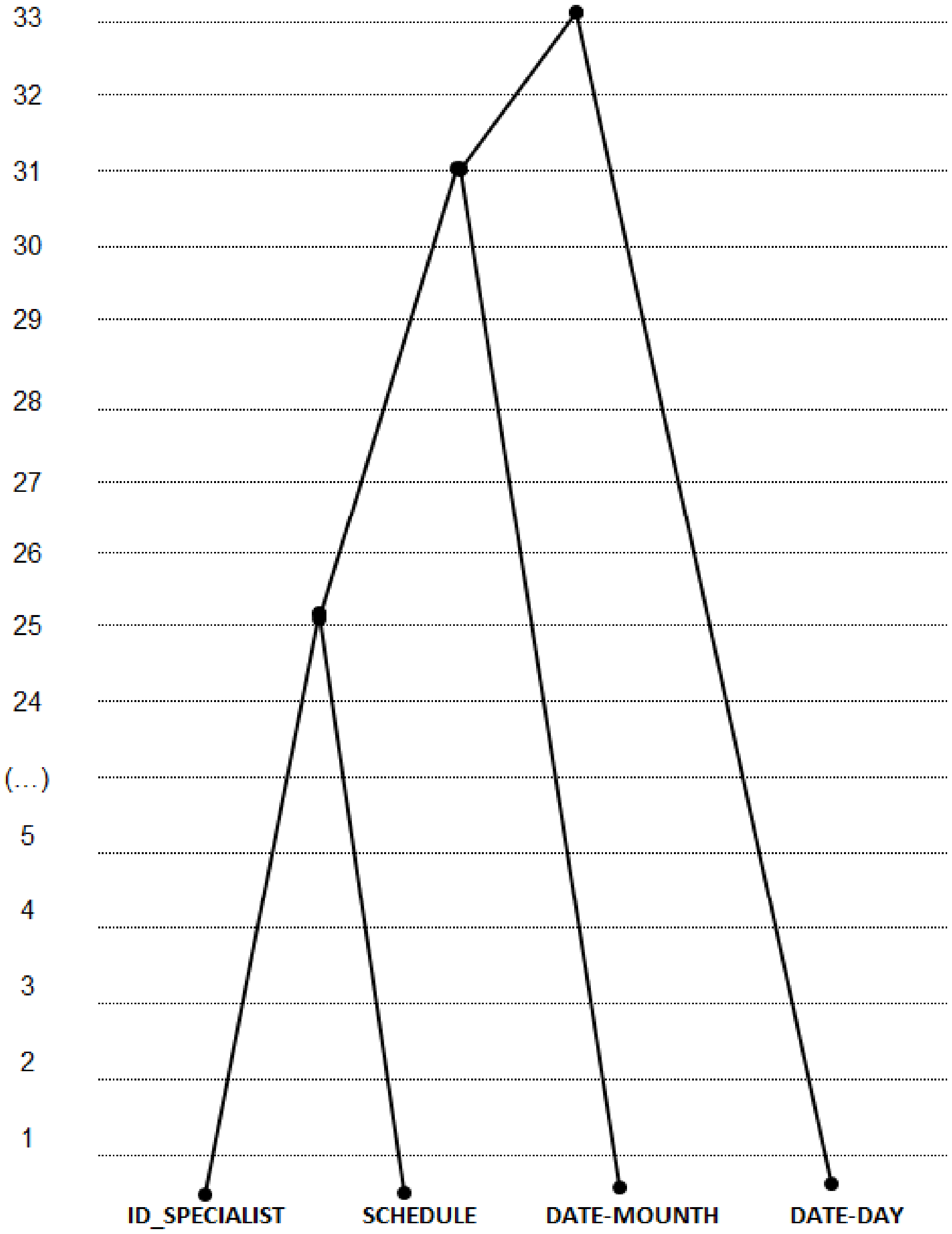

Figure 2: Ordered Tree of Characteristics

- **Activity B.4- Discussion of the Results**

  As a last task, a meeting is held with the pediatrician and the office secretary who take the role of experts in the domain to discuss the results generated in the previous tasks that are interpreted in activity B.3.

  In the first instance, the analysis of the ordered tree of elements is presented to the pediatrician and the office secretary, explaining the meaning of the tree together with its interpretation. In view of the relationship between normal appointments with status completed and suspended together with completed over-shift appointments, they mention that this relationship is correct, given that the behavior of over-shift appointments is similar to the behavior of normal appointments. The only difference is that the over-shift appointments are requested on the same day by the patient and not in advance as the shifts. On the other hand, they agree that it is correct that suspended over-shift appointments are not within the same relationship because it is very rare for over-shift appointments to be canceled by patients. In general, unless the patient requires a very urgent consultation and must attend an appointment, patients who ask for over-shift appointments attend the office at the time indicated by the secretary.

Then, the analysis of the attribute groupings is continued. For this purpose, the ordered tree of characteristics is presented, together with its interpretation. In view of this, they express that they also agree on the relationships generated, since the specialist who attends in the office always has a certain schedule. In this case, within the office, there are only two specialists but the objective is to incorporate even more in determined schedules. Likewise, there are times when the specialists take turns (for example, during vacation periods) so it is considered correct to relate this to the month and finally to the day of the month in which they attend. Likewise, the doctor tells us that he would have liked to know the season of the year, because depending on the season there may be more or less appointments. As the season can be calculated with the existing data, it is not necessary to carry out a new analysis and the interpretation of the tree is considered correct.

It is therefore concluded that the version of the data is representative of the behavior of the office's normal and over-shift appointments. From these conclusions, the version of the data used is representative of the behavior of the appointments of the office, therefore, it is the one that can be used for the implementation of the intelligent system.

6 Implementation of the Intelligent System Prototype

From the data considered as representative and the problem to be solved within the case study, an initial prototype of the intelligent system is built. In this case it is very important to carry out a prediction of the number of normal and over-shift appointments with precision because it is critical to ensure the success of the model, but it is not necessary to know how the results generated were obtained. In other words, it is not necessary for the network to explain or justify the results obtained. Therefore, in this case, it has been determined that the most suitable architecture for the intelligent system corresponds to Multi-Perception Artificial Neural Networks (ANN) with training by backward error correction [16], better known as BackPropagation ANN [46].

This type of network applies supervised training and only has forward connections between its neurons which are organized in several layers [47]:

– the first layer is the input layer for receiving the values from the outside.

– the last layer is the output layer for returning the results generated by the network, and

– he neurons located in the intermediate layers are called hidden or processing layers (in a network there may be no hidden layer, a hidden layer or more than one).

Before starting work on ANN modeling, it is necessary to transform the available data to a format that best suits this architecture. Then, the values are converted into numerical values so that they can be used. Since it is desired to be able to estimate the number of normal and over-shift appointments for a given date and time, taking into account the appointments actually attended (i.e. appointment with a "FINISHED" status) such as those that were requested but then canceled (in a "SUSPENDED" status), the following input and output attributes are defined for ANN:

- Input Attributes:
  - *DATE-MONTH*, corresponding to the month of the date to be estimated.
  - *DATE-DAY*, which corresponds to the day of the month of the date to be estimated.
  - *DATE-SSN*, which indicates the season of the year corresponding to the date to be estimated (where 1 corresponds to "Summer", 2 to "Fall", 3 to "Winter", and 4 to "Spring").
  - *DATE-SCHDL*, which indicates the time of the date to be estimated (where 1 corresponds to "Morning", 2 to "Noon", and 3 to "Evening").
- Output Attributes:
  - *N-APP-FINISH*, which indicates the number of normal appointments with finished status for that date and time.
  - *N-OSA-FINISH*, which indicates the number of over-shift appointments with finished status for that date and time.
  - *N-APP-SUSP*, which includes the number of normal appointments with suspended status for that date and time.
  - *N-OSA-SUSP*, which indicates the number of over-shift appointments with suspended status for that date and time.

An extract of the prepared data can be seen in Table 7.

Table 7: Extract from the prepared data

DATE-MONTH	**DATE-DAY**	**DATE-SSN**	**DATE-SCHDL**	**N-APP-FINISH**	**N-OSA-FINISH**	**N-APP-SUSP**	**N-OSA-SUSP**
4	11	2	3	0	1	2	4
5	17	3	1	11	2	7	8
7	16	3	3	4	1	0	1
9	31	4	2	1	1	1	0
11	1	1	1	9	5	0	4

Once the data is prepared, the initial model is built. For this purpose, the NEAT4J framework[2] is used, which implements in Java the "NeuroEvolution of Augmenting Topologies" or NEAT [48] algorithm for the construction of ANN using evolutionary algorithms. In this case, a fitness function provided by the framework called "MSE NEAT Fitness Function" is used to minimize the error when calculating the square average of the difference between the desired output and the one generated by the network. Due to the requirements of this function, the values of the attributes have had to be reformatted into decimal numbers (when divided by 100), as can be seen in Table 8. This data is then stored in a new CSV file to be accessed from the framework.

Finally, the data from the new CSV file are supplied to NEAT4J, so that the latter may begin to evolve possible ANN topologies to generate the quantities corresponding to the date to be estimated.

[2] http://neat4j.sourceforge.net/

Table 8: Extract from the prepared data and formatted to decimal values

DATE-MONTH	DATE-DAY	DATE-SSN	DATE-SCHDL	N-APP-FINISH	N-OSA-FINISH	N-APP-SUSP	N-OSA-SUSP
0.04	0.11	0.02	0.03	0.00	0.01	0.02	0.04
0.05	0.17	0.03	0.01	0.11	0.18	0.07	0.08
0.07	0.16	0.03	0.03	0.04	0.01	0.00	0.01
0.09	0.31	0.04	0.02	0.02	0.01	0.01	0.00
0.11	0.01	0.01	0.01	0.09	0.05	0.00	0.04

After several runs, the "species" with the best suitability value (i.e., the ANN topology that generates the least error in the prediction) is obtained in "time" (or cycle) 939 of the 3rd run with a suitability value of 0.0569. The topology thus generated is shown in Figure 3.

This topology contains:

- 4 Neurons in the Entrance Layer.
- 4 Neurons in the First Hidden Layer.
- 4 Neurons in the Second Hidden Layer.
- 5 Neurons in the Third Hidden Layer.
- 3 Neurons in the Fourth Hidden Layer.
- 9 Neurons in the Fifth Hidden Layer.
- 4 Neurons in the Exit Layer.

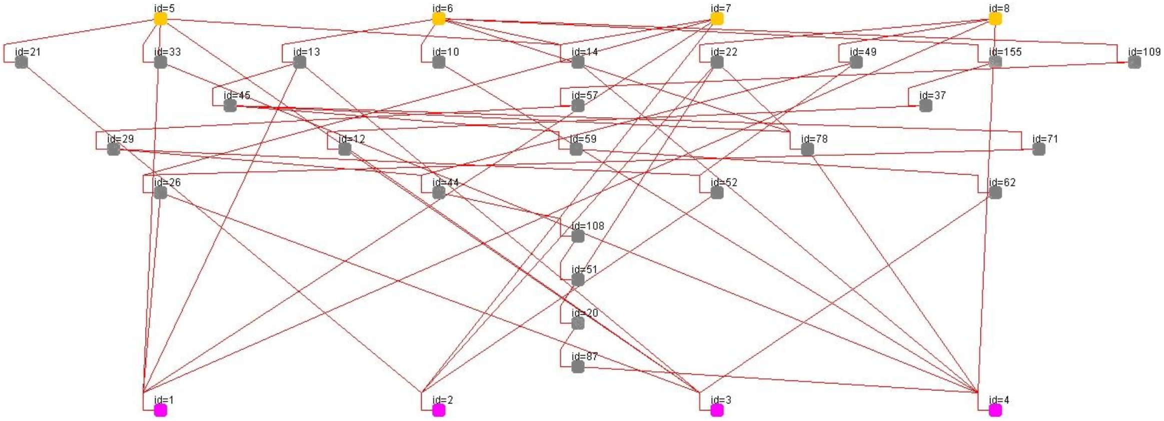

Figure 3: Artificial neural network topology

From this defined topology, the ANN prototype is constructed using the NeurophStudio tool[3]. Then, a new project is generated in which a Multi-Perceptron BackPropagation ANN is defined that applies the topology defined by NEAT4J as can be seen in Figure 4.

This ANN is then trained and validated with the data analyzed in the previous section. Although in a real project it would not make sense to validate an intelligent system with the same data with

[3] http://neuroph.sourceforge.net/

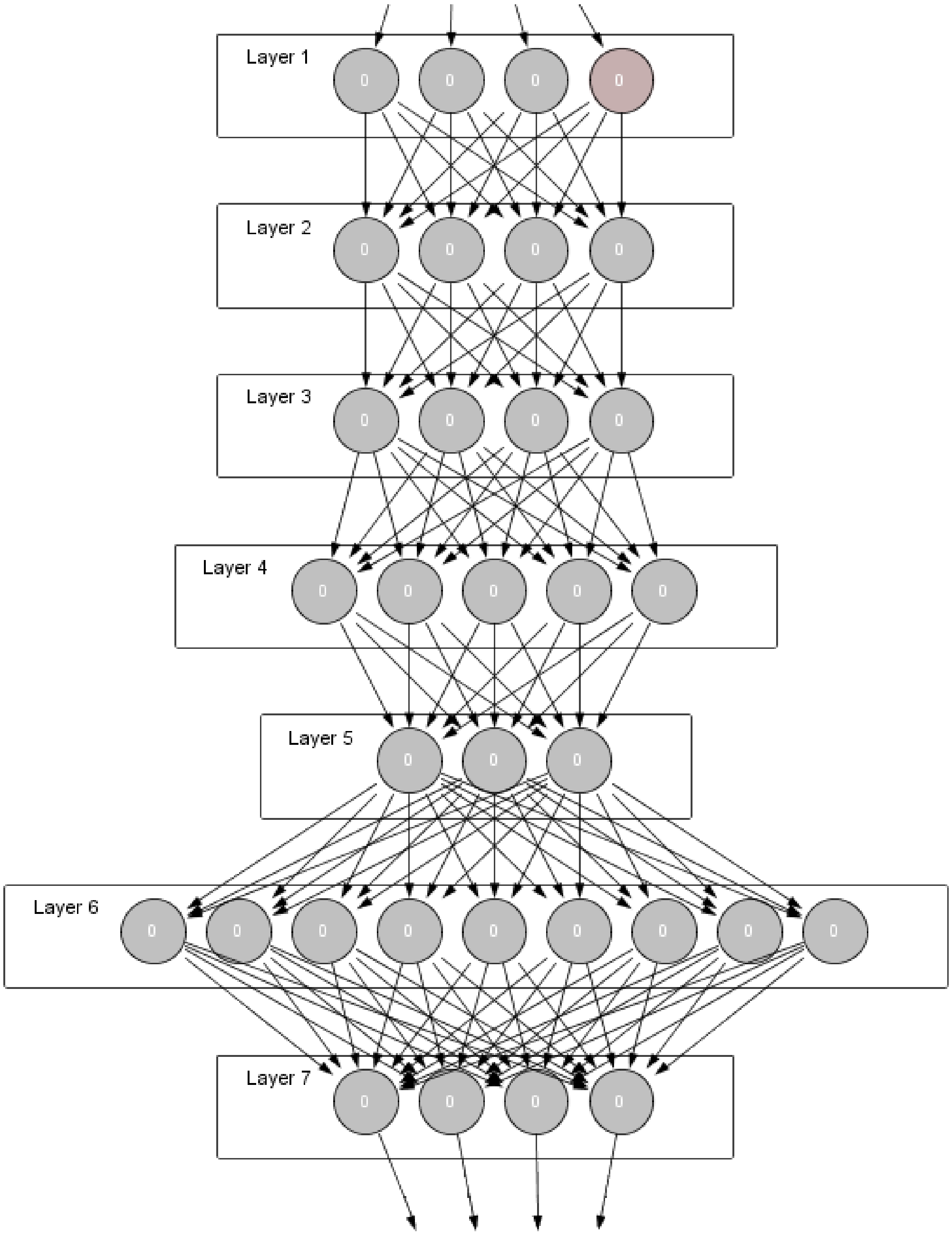

Figure 4: ANN architecture implemented in NeurophStudio

which it was trained (given that the precision thus obtained is not reliable), here we only seek to confirm that the probabilities provided by the network can be considered as representative of the data used. In this case, an accuracy of 98.8% is obtained for completed normal appointments, 98.9% for suspended normal appointments, 95.3% for completed over-shift appointments and 97.8% for suspended over-shift appointments. Based on these results, it is possible to ensure that this ANN is useful to be used as a basis for the Intelligent System that predicts the number of normal and over-shift appointments in the doctor's office.

7 Conclusions

The Repertory Grid technique is used to generate an objective representation of the mental image that a person has of the elements in a given area, seeking to go beyond the words that the person could convey. In the present work, an adaptation of this technique is implemented to evaluate the data obtained about the medical appointments of a pediatric office, in order to determine if they can be used for the construction of an intelligent system that predicts the number of normal and over-shift appointments in a certain date. From the analysis made, it has been detected that the version of the data is considered sufficiently representative and therefore useful for the implementation of the Intelligent System. In turn, this has been confirmed with a prototype with an Artificial Neural Network modeled and trained from these data.

As a future line of work, the prototype will be applied in the medical office to evaluate its capacity for generalization in new situations.

Authors' Information

- **Cinthia Vegega** is an Information Systems Engineer from the Universidad Tecnológica Nacional Facultad Regional Buenos Aires and currently is finishing her Master in Information Systems Engineering at the same university. She currently works as Interim Adjunct Professor at Universidad Tecnológica Nacional Facultad Regional Buenos Aires and Universidad Tecnológica Nacional Facultad Regional La Plata.

- **Pablo Pytel** is an Information Systems Engineer from the Universidad Tecnológica Nacional Facultad Regional Buenos Aires, has a Master in Software Engineering from the Universidad Politécnica de Madrid, and a PhD in Computer Sciences from the Universidad Nacional de La Plata. He currently works as a Professor in the at the Universidad Tecnológica Nacional Facultad Regional Buenos Aires and Universidad Nacional de Lanús.

- **María Florencia Pollo-Cattaneo** is an Information Systems Engineer from the Universidad Tecnológica Nacional Facultad Regional Buenos Aires, has a Master in Software Engineering from the Universidad Politécnica de Madrid, and a PhD in Computer Sciences from the Universidad Nacional de La Plata. She currently works as a Head Professor at the Universidad Tecnológica Nacional Facultad Regional Buenos Aires and Universidad Tecnológica Nacional Facultad Regional La Plata. She is also GEMIS Group Chair.

Authors' Contributions

- Cinthia Vegega focused on the specifications of the stages of the proposed process and wrote the manuscript.

- Pablo Pytel supervised and guided the proposed process, managed and supervised the validation of the proposed process, and wrote the manuscript.

- María Florencia Pollo-Cattaneo managed and supervised the validation of the proposed process, and wrote the manuscript.

Competing Interests

The authors declare that they have no competing interests.

Funding

The research reported in this paper has been totally funded by the research project "Prácticas Ingenieriles aplicadas para la implementación de Sistemas Inteligentes basados en Machine Learning" (UTI5103TC) within Universidad Tecnológica Nacional Facultad Regional Buenos Aires.

Availability of Data and Material

GEMIS-TD-2019-09-TR-2019-10-ResultadosTurnosMedicos. Results of clustering procedure on data for medical appointments in a pediatric office.
Available at https://github.com/inteligenciaartificialutn/Paradigm

Editor

- **Hector Florez**, Ph.D. *Universidad Distrital Francisco Jose de Caldas, Colombia*

Reviewers

- **Mohamed Salim El Bazzi**, Ph.D. *Université Ibn Zohr, Morocco*
- **Tong Li**, Ph.D. *Beijing University of Technology, China*
- **Ivan Mura**, Ph.D. *Duke Kunshan University, China*

References

[1] E. Alpaydin, *Introduction to machine learning*. MIT press, 2014.

[2] T. M. Mitchell, *Machine learning*. McGraw-Hill, New York, 1997.

[3] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, *Machine Learning: An Artificial Intelligence Approach*. Springer, 1983.

[4] L. Ljung, "Perspectives on system identification," *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 7172–7184, 2008.

[5] P. R. Cohen and E. A. Feigenbaum, *The handbook of artificial intelligence*, vol. 3. Butterworth-Heinemann, 2014.

[6] R. García Martínez, D. Pasquini, and M. Servente, *Sistemas inteligentes*. Nueva Librería,, 2003.

[7] P. Domingos, "A few useful things to know about Machine Learning," *Communications of the ACM*, vol. 55, no. 10, pp. 78–87, 2012.

[8] G. Bontempi, S. B. Taieb, and Y.-A. Le Borgne, "Machine learning strategies for time series forecasting," in *European business intelligence summer school*, pp. 62–77, Springer, 2013.

[9] E. A. R. Santoyo and J. A. L. González, "Comparación de predicción basada en redes neuronales contra métodos estadísticos en pronósticos de ventas," *Ingeniería Industrial. Actualidad y Nuevas Tendencias*, vol. 4, no. 12, pp. 91–105, 2014.

[10] J. Pérez Porto and A. Gardey, "Bias definition." Available at https://definicion.de/sesgo/, 2010. Accessed: 2020-01-01.

[11] M. Vivanco, *Muestreo estadístico. Diseño y aplicaciones*. Editorial Universitaria, 2005.

[12] N. Collins, "Artificial intelligence will be as biased and prejudiced as its human creators," *Pacific Standard*, vol. 1, 2016.

[13] K. Crawford, "Artificial intelligence's white guy problem," *The New York Times*, vol. 25, 2016.

[14] C. Vegega, P. Pytel, and M. F. Pollo, "Método basado en el emparrillado para evaluar los datos aplicables para entrenar algoritmos de aprendizaje automático," *Desarrollo e innovación en ingeniería*, pp. 106–137, 2017.

[15] C. Vegega, P. Pytel, L. Straccia, and M. F. Pollo-Cattaneo, "Evaluation of the bias of student performance data with assistance of expert teacher," in *International Conference on Applied Informatics*, pp. 16–31, Springer, 2018.

[16] S. Walczak, "An empirical analysis of data requirements for financial forecasting with neural networks," *Journal of management information systems*, vol. 17, no. 4, pp. 203–222, 2001.

[17] R. J. Hyndman, A. V. Kostenko, *et al.*, "Minimum sample size requirements for seasonal forecasting models," *foresight*, vol. 6, no. Spring, pp. 12–15, 2007.

[18] S. J. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 252–264, 1991.

[19] D. R. Stockwell and A. T. Peterson, "Effects of sample size on accuracy of species distribution models," *Ecological modelling*, vol. 148, no. 1, pp. 1–13, 2002.

[20] E. Alpaydin, *Machine learning: the new AI*. MIT press, 2016.

[21] T. M. Mitchell, *The need for biases in learning generalizations*. Department of Computer Science, Laboratory for Computer Science Research, 1980.

[22] P. Domingos, *The master algorithm: How the quest for the ultimate learning machine will remake our world*. Basic Books, 2015.

[23] W. I. Limited, "Beauty AI. First Beauty Contest Judged by Robots." Available at http://beauty.ai/, 2015. Accessed: 2020-01-01.

[24] S. Levin, "Beauty Contest was Judged by AI and the Robots didn't like Dark Skin," 2016. The Guardian.

[25] W. L. Perry, *Predictive policing: The role of crime forecasting in law enforcement operations*. Rand Corporation, 2013.

[26] D. Robinson and L. Koepke, "Stuck in a pattern: Early evidence on 'predictive policing'and civil rights," *Upturn report*, pp. 1–29, 2016.

[27] E. Edwards, "Predictive policing software is more accurate at predicting policing than predicting crime," *HUFFPOST POST*, 2016.

[28] R. Lloyd, "Critics say a predictive policing system could amplify racial bias in oakland," 2016.

[29] Northpointe, "Practitioner's guide to compas core," 2015.

[30] S. M. L. K. Angwin, J. Larson, "There's software used across the country to predict future criminals. and it's biased against blacks," 2016.

[31] J. M. . A. S. J. Trujillano, "Aproximación metodológica al uso de redes neuronales artificiales para la predicción de resultados en medicina.," *Med Clin, Barc*, 2004.

[32] G. Kelly, *The Psychology of Personal Construct*. New York, Norton, 1955.

[33] RAE, "Real académica española constructo."

[34] F. Fransella, R. Bell, and D. Bannister, *A manual for repertory grid technique*. John Wiley & Sons, 2004.

[35] R. Wolf and H. S. Delugach, "Knowledge acquisition via tracked repertory grids," *Computer Science Dept., Univ. Alabama in Huntsville*, 1996.

[36] D. Carrizo Moreno, "Comparación de efectividad de las técnicas de educción de requisitos software: visión novel y experta," *Ingeniare. Revista chilena de ingeniería*, vol. 20, no. 3, pp. 386–397, 2012.

[37] M. Easterby-Smith, "The design, analysis and interpretation of repertory grids," *International Journal of Man-Machine Studies*, vol. 13, no. 1, pp. 3–24, 1980.

[38] P. Britos, B. Rossi, and R. García Martínez, "Notas sobre didáctica de las etapas de formalización y análisis de resultados de la técnica de emparrillado. un ejemplo," in *Proceedings del V Congreso Internacional de Ingeniería Informática*, pp. 200–209, 1999.

[39] P. E. Black, "Manhattan distance Dictionary of algorithms and data structures." Available at https://xlinux.nist.gov/dads//HTML/manhattanDistance.html, 2006. Accessed: 2020-01-01.

[40] J. M. Bradshaw, K. M. Ford, J. R. Adams-Webber, and J. H. Boose, "Beyond the repertory grid: new approaches to constructivist knowledge acquisition tool development," *International Journal of Intelligent Systems*, vol. 8, no. 2, pp. 287–333, 1993.

[41] L. A. Saúl, M. A. López-González, A. Moreno-Pulido, S. Corbella, V. Compan, and G. Feixas, "Bibliometric review of the repertory grid technique: 1998–2007," *Journal of Constructivist Psychology*, vol. 25, no. 2, pp. 112–131, 2012.

[42] P. McGeorge and G. Rugg, "The uses of contrived knowledge elicitation techniques," *Expert Systems*, vol. 9, no. 3, pp. 149–154, 1992.

[43] S. Stumpf and J. McDonnell, "Using repertory grids to test data quality and expertś hunches," in *14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings.*, pp. 806–810, IEEE, 2003.

[44] S. Stumpf and J. McDonnell, "Data, information and knowledge quality in retail security decision making," in *Proceedings of I-KNOW*, vol. 3, pp. 2–4, Citeseer, 2003.

[45] R. Rakotomalala, "Tanagra: data mining software for academic and research purposes," *Actes de EGC*, pp. 697–702, 2005.

[46] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Parallel Distributed Processing*, vol. 1, 1986.

[47] H.-N. Robert *et al.*, "Theory of the backpropagation neural network," *Proc. 1989 IEEE IJCNN*, vol. 1, pp. 593–605, 1989.

[48] H. Heidenreich, "NEAT: An Awesome Approach to NeuroEvolution." `https://towardsdatascience.com/neat-an-awesome-approach-to-neuroevolution-3eca5cc7930f?gi=fb4832b83f34`, 2019. Accessed: 2020-01-01.