

DATA MINING MODEL PERFORMANCE OF SALES PREDICTIVE ALGORITHMS BASED ON RAPIDMINER WORKFLOWS

Alessandro Massaro, Vincenzo Maritati, Angelo Galiano

Dyrecta Lab, IT research Laboratory, via Vescovo Simplicio,
45, 70014 Conversano (BA), Italy

ABSTRACT

By applying RapidMiner workflows has been processed a dataset originated from different data files, and containing information about the sales over three years of a large chain of retail stores. Subsequently, has been constructed a Deep Learning model performing a predictive algorithm suitable for sales forecasting. This model is based on artificial neural network –ANN- algorithm able to learn the model starting from sales historical data and by pre-processing the data. The best built model uses a multilayer neural network together with an “optimized operator” able to find automatically the best parameter setting of the implemented algorithm. In order to prove the best performing predictive model, other machine learning algorithms have been tested. The performance comparison has been performed between Support Vector Machine –SVM-, k-Nearest Neighbor k-NN-, Gradient Boosted Trees, Decision Trees, and Deep Learning algorithms. The comparison of the degree of correlation between real and predicted values, the average absolute error and the relative average error proved that ANN exhibited the best performance. The Gradient Boosted Trees approach represents an alternative approach having the second best performance. The case of study has been developed within the framework of an industry project oriented on the integration of high performance data mining models able to predict sales using –ERP- and customer relationship management –CRM- tools.

KEYWORDS

RapidMiner, Neural Network, Deep Learning, Gradient Boosted Trees, Data Mining Performance, Sales Prediction.

1. INTRODUCTION

The proposed paper has been developed within the framework of an industrial project. In this section is discussed the state of the art of the project scenario and the main motivations of the proposed paper. In [1] was analysed the applicability of self-learning artificial neural network -ANN- algorithms for the prediction of sales revenues for grocery stores. In particular, the feed-forward multilayer perceptron neural network algorithm (MLP) was applied to predict supermarket sales demonstrating how such networks can predict sales with greater reliability than the case statistical prediction [2]. For this reason ANN models are good candidates for sales forecasting. A better prediction accuracy can be performed by combining other algorithms [3]. This suggested to apply different predictive algorithms in order to select the best performing one. An important aspect to focus about the prediction is the customer behaviour: a study has shown that neural networks could predict the abandonment rate of customers to cellular network services [4]. Customer relationship management -CRM- are important tools concerning customer behaviour [5] thus suggesting to integrate CRM into a platform improved by a data mining engine suitable for sale prediction. ANN can be integrated into a data mining framework according to the

application context [6]. The economic analysis, for example of market segmentation [7], must accompany the model of artificial intelligence in order to orientate the predictive analysis in the specific sector of interest by reducing the prediction error at the same time. A business model in fact is a non-linear model which can be constructed in an appropriate manner through the integration of the different processing phases of the data mining [8], and through an accurate analysis of the market model associated to customers [9]. For this reason a complete system information able to provide a business model should also integrate data of the enterprise resource planning –ERP- platform. In any case, a complete model will have to process also exogenous factors as input [10] which can be controlled or uncontrolled. For example, a predictive analysis of sales can be carried out by analysing customer behaviour correlated with weather conditions [11]-[14]. Concerning business intelligence processes related to large-scale retail (GDO), in literature are analysed data mining algorithms able to optimize sales by analysing special dashboards for monitoring products in stock, for product facing strategies, and for the activation of promotional activities [15]. The business intelligence tools could be implemented by analysing Market Basket Analysis –MBA- indicators and marketing strategies related to the layout of products in the stores [16]-[20], while the sales activity targeted by type of customers could be carried out by profiling and segmenting customers by processing data from a CRM. Data mining algorithms can be implemented through the use of graphical user interfaces –GUIs- easy to use and compatible with several operating systems [15]. This suggest to use user friendly GUIs such as workflow made by objects. A tool suitable for the implementation of a data mining workflow is RapidMiner [21]. Other open source tools which can be adopted are Knime, Orange Canvas, R language, Keras, Theano and TensorFlow [22]-[25]. Sales prediction can be achieved by a data mining process made by the following phases: (i) prior knowledge, (ii) data preparation, (iii) modelling, (iv) application, final knowledge (output results of a decision support system) [21]. All the phases (i), (ii), and (iii) are related to a macro-phase of data pre-processing. The state of the art described above suggested the development of a project represented by the architecture of Fig. 1 where is illustrated the basic architecture layout which integrates different technologies such as CRM, ERP and data mining engine, all communicating with a central database (DB). The central DB is able to collect all customer and sales data considered as input of the data mining engine. In this paper different data mining algorithms have been applied in order to find the best method able to predict weekly sales. The paper is structured as follows:

- Description of the main blocks of the RapidMiner workflow describing the data mining macro-phase of data pre-processing;
- Application of the best workflow predictive models characterized by good error performances;
- Comparison and evaluation of absolute errors, relative errors and correlation between real and predicted values of different predictive algorithms by defining the best performance;
- Conclusions.

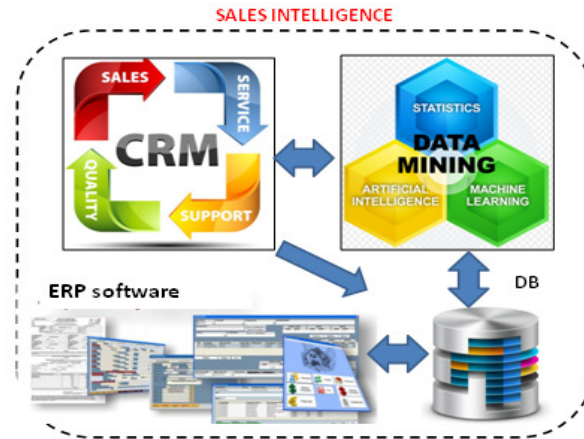


Figure 1. Sales Intelligence platform: system architecture integrating a data mining engine.

2. SALES PREDICTIVE MODELS

In this section are described the workflow RapidMiner layouts able to model sales predictions and used for the information system of Fig.1. In particular the attention is focused on the workflow blocks related the data pre-processing. The data pre-processing is a very important part of the design of a data mining workflow. For this purpose in this section are discussed the functionalities of each block in order to highlight the procedure able to treat data of different stores and of different data sources.

2.1. Experimental Dataset

In order to test predictive workflow it is important to select a database concerning real data. As a reference database, the "Store Sales Forecasting" public dataset made available on the Kaggle platform by Walmart represent a good dataset to process [26]. This dataset describes the sales made by 45 stores in different regions, each with different departments and products for sale. The dataset contains a large number of records (over 140000 sales records in a given week for each store in the training dataset), and it is incomplete (not all fields of all records are completed). For these two reasons it is an excellent dataset to test and to validate forecasting models. The analysed dataset consists of the following four database CSV files to load in the central DB:

- *features.csv*.
This file contains different data of the store, of departments, and of regional activities for given dates. In particular it contains the following attributes:
 - Store (store number);
 - Date (week date);
 - Temperature (average temperature in the region);
 - Fuel_Price (cost of fuel in the region);
 - Markdown1-5 (anonymized data related to promotional markdowns; this dataset is not available for all stores all the time; the missing values are indicated with NA);
 - CPI (consumer price index);
 - Unemployment (unemployment rate);
 - IsHoliday (special week such as holiday week).

- *stores.csv*.
This file contains information about the 45 stores, also indicating the type and the size of store.
- *train.csv*.
This file is the training dataset containing historical data. It covers periods from 2010-02-05 to 2012-11-01. Within this file you will find the following fields:
 - Store (store number);
 - Dept (department number);
 - Date (week data);
 - Weekly_Sales (weekly sales for departments);
 - IsHoliday - whether the week is a special holiday week
- *test.csv*.
This file is identical to train.csv; it contains the testing dataset.

2.2. Data Mining Workflows of the best Predictive Models: data pre-processing and prediction optimisation

The workflow able to calculate sales prediction is made by different linked blocks as sketched in Fig. 2. Below are described the main functionalities of each block concerning a RapidMiner Deep Learning workflow by focusing the attention on data pre-processing.

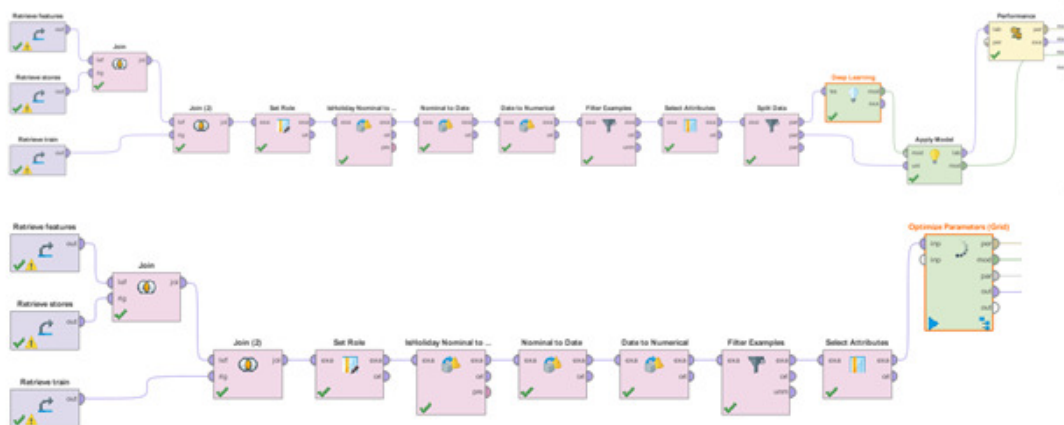


Figure 2. Rapid Miner: (above) complete Deep Learning workflow; (below) workflow with “Optimize Parameter(Grid)” operator.

2.2.1 Data Input: data loading in local repository and data merging by join operator

The data input blocks are modelled by the “Retrieve” operator. The “Retrieve” operator loads a Rapid Miner object into the data flow process. In the specific case it allow to select data stored in the local repository extracted by the csv experimental dataset. In order to design a functional training dataset, a logic union must applied on all three datasets, obtaining a single dataset to process. This operation can be performed in Rapid Miner using the “Join” operator. The “Join” block joins two dataset using one or more attributes of the input dataset as key attributes. In the studied case the first “Join” operator takes data from "Train" and "Stores" datasets and operates a merging process in the "Store" field: at each record of the "Train" dataset it is added the information of the store typology and of the related size expressed in square feet. The second Join operator “Join(2)” creates the dataset resulting from the first join operation and joins it to the "Features" dataset: this allows to associate to each record the information regarding temperature, price index, unemployment index, fuel price and any promotions in progress in a certain area of the store. In conclusion, starting from three datasets of the experimental dataset it is generated a

new dataset containing on each line the data stored in all three initial datasets. In Fig. 3 is illustrated the final dataset created in the local repository by the two “Join” operators.



Figure 3. Fields of the original datasets and of the final one generated by the two “Join” operators.

RapidMiner allows to check the data processing at each steps of the workflow. In Fig. 4 is illustrated an example of scattering data plot indicating the temperatures of all the analysed stores by proving the correct data loading in the local repository. The graphical dashboards could help to interpret results at each stage of the data processing workflow. In this way it is possible to check if the dataset to process is suitable for the data processing by checking good data quality, outliers, anomalous data etc. This allow to adjust the training and the testing datasets. Another example proving the correct importing of the dataset in the local repository are provided by Fig. 5 and Fig. 6.

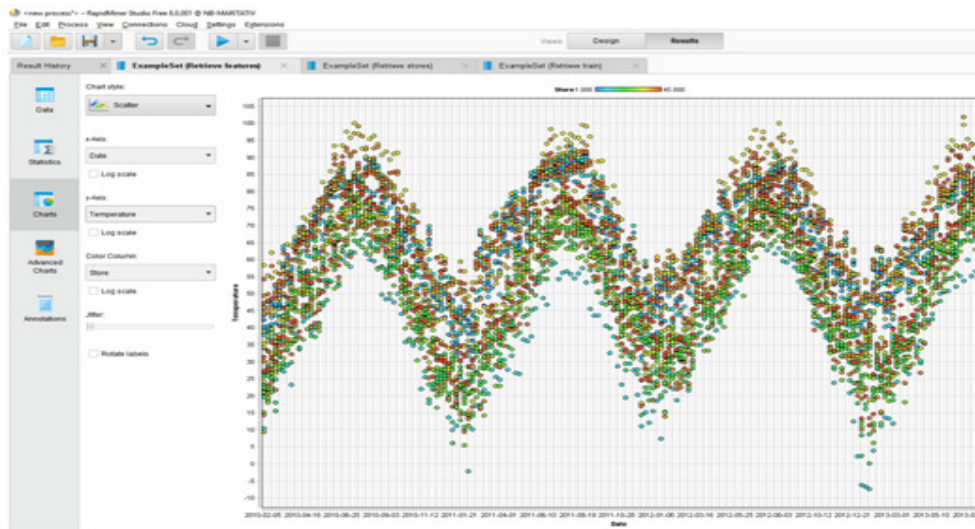


Figure 4. Experimental dataset: store temperatures versus dates.

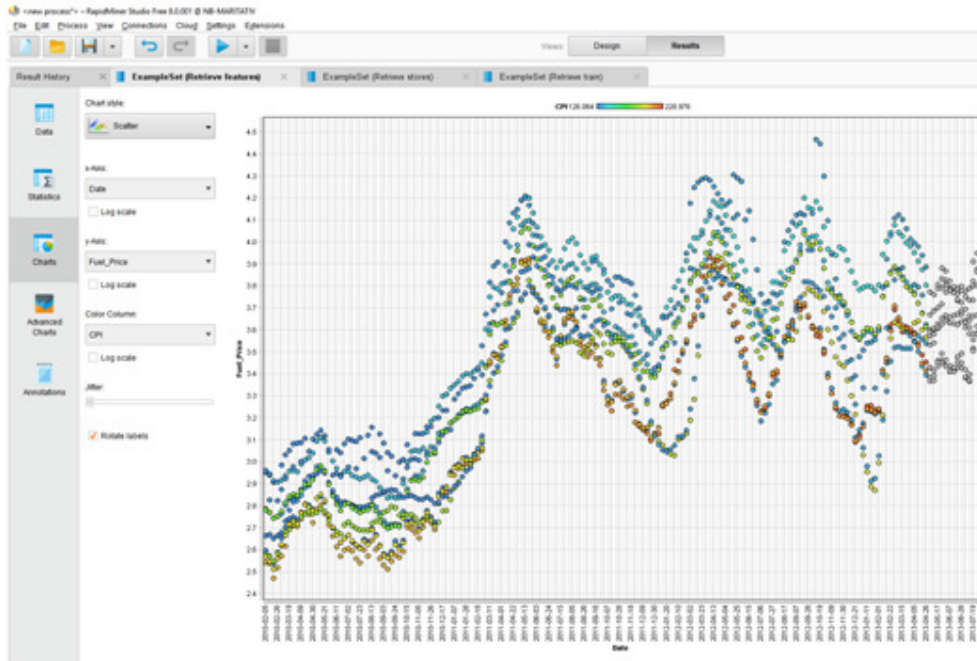


Figure 5. Experimental dataset: fuel price versus dates.

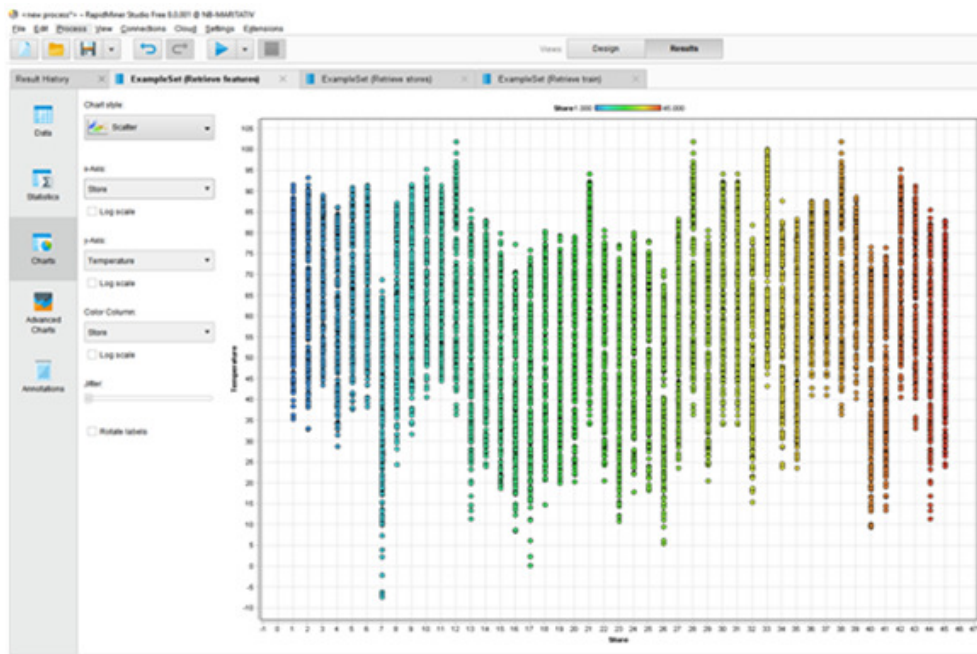


Figure 6. Experimental dataset: store temperatures versus dates.

2.2.2 Data pre-processing operators

After the union data process has been performed the data transformation process by the following blocks:

- “Set Role”. The role of an attribute describes how other operators handle this attribute. The default role is regular, other roles are classified as special. A dataset to process can have many special attributes, but each special role can only appear once. This block is able to set this special role.
- “IsHoliday_Nominal to Binomial” (Nominal to Binomial). The Nominal to Binomial operator is applied to change the type of nominal attributes to a binomial type. This operator also maps binomial values such as ‘true’ and ‘false’.
- “Nominal to Date”. The “Nominal to Date” operator converts the selected nominal attribute of the input dataset into date and/or time type. This conversion is performed by considering specific date format string. The old nominal attribute will be changed with a new date and/or time attribute.
- “Date to Numerical”. The “Date to Numerical” operator allows to select millisecond, second, minute, hour, day, week, month, quarter, half year, and year by converting information into a number.
- “Filter Examples”. The operator data matching with a defined logic condition.
- “Select Attributes”. The operator provides different filter types to facilitate attribute selection by executing a regular expression.

In order to use efficiently attributes such as the dates, by considering if the week taken into consideration has holidays, it is necessary to change the attributes type. These attributes are initially recognized by RapidMiner as text fields. The attribute "Data" must be converted into a data format that can be processed by Rapid Miner. The attribute "IsHoliday", which has only two possible values, must be converted in the number 0 for the lines with the textual value "False", and in the number 1 for lines characterised by the text value "True". In this way these attributes will be transformed as input data which can be processed by the artificial neural network represented by the “Deep Learning” operator. Using the "Nominal to Date" operator, it is possible to change the format of the text attribute that contains dates into a data type attribute compatible with RapidMiner processing logic. In order to use efficiently the information provided by this last attribute it is important to consider that the sales trend follows the period of the year in which they occur, it is important to change the "Date" field again using the "Date to Numeric" operator thus allowing to convert the date into a numerical format. In the analysed case the date is converted into a number that corresponds to the week number counted on an annual basis: in this way, dates which are "close" on different years will be treated similarly by the system which will be enabled to recognize particular periods of the year such as the Christmas holidays or the Summer holiday period. The “Filter Example” operator is applied to select only the examples related to a particular shop and to a particular department of the selected shop. The “Select Attributes” operator are applied to select attributes having major information such as *Date*, *Dept*, *Store*, and *Weekly_Sales*. To perform the training and validation of the artificial neural network model it was necessary to train it on the train dataset and test it on another dataset with the same structure as the first one but with different data. In RapidMiner "Split Data" operator is the special operator adopted to split dataset into the training and the testing datasets. In our case we make a split 85:15 in particular starting from the original dataset are created the training dataset and the testing dataset containing respectively 85% and 15% of the data, respectively. The testing dataset is used to test the accuracy of the created model.

2.2.3 Data processing: Deep Learning and Gradient Boosted Trees operators

The “Deep Learning” block is based on a multi-layer feed-forward ANN that is trained with stochastic gradient descent using back-propagation method. The network can contain a large number of hidden layers consisting of neurons with tanh, rectifier and maxout activation functions. Advanced features such as adaptive learning rate, rate annealing, momentum training, dropout and other regularizations enable high predictive accuracy by decreasing the probabilistic prediction error. The “Deep Learning” operator is able to instantiate a type of artificial neural network arranged to work on multiple layers (hence the term "deep", deep) and allows the selection of different types of activation functions for the artificial neural network. All the main training parameters are also available, in order to be able to instantiate almost all types of artificial neural network today state of the art. The “Optimize Parameters (Grid)” operator (see Fig. 2 below) is an operator which executes the sub-process for all combinations of selected values of the parameters by delivering the optimal parameter values and the optimal parameter set. This last operator has been applied for Deep Learning and Gradient Boosted Trees applications.

Using “Optimize Parameters (Grid)” operator for the Deep Learning application, the following settings have been made:

- Activation function: Exponential Rectifier;
- Hidden layers: 5 layers hidden by 50 artificial neurons each;
- Training periods: 2000 for the tests and validation of the parameters; 20000 for the final training of the artificial neural network;
- Cost function: quantile;
- Distribution function: automatic (the Laplace distribution function is selected);
- Max w2: 0 (maximum value of the sum of the square roots of the inputs of a neuron, set to zero allows any value without limits, by default it is set to the value 10).

The best parameters setted by the optimizer are the following:

Deep Learning.activation = ExpRectifier
Deep Learning.learning_rate = 0.22311111111111112
Deep Learning.loss_function = Quantile
Deep Learning.distribution_function = Gaussian

The workflow concerning the Gradient Boosted Tree processing (see Fig. 7) is similar to the previous one illustrated in Fig. 2.

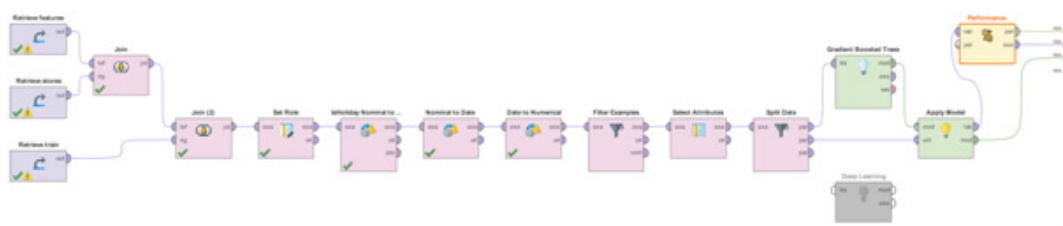


Figure 7. Rapid Miner: complete Gradient Boosted Trees workflow.

Using the “Optimize Parameters (Grid)” operator, the parameters to be optimized in the Gradient Boosted Tree are the following ones:

- Number of trees to be created;
- Maximum depth of each tree;
- Minimum number of lines to be taken into account;
- Learning rate;

The best parameters setted by the optimizer are the following:

Gradient Boosted Trees.number_of_trees = 10
Gradient Boosted Trees.maximal_depth = 41
Gradient Boosted Trees.min_rows = 18.64
Gradient Boosted Trees.learning_rate = 0.6400360000000002

3. RESULTS AND COMPARISON

3.1. Best results: Artificial Neural Network performed by “Deep Learning” operator and output interpretation

In this section are discussed some results concerning Deep Learning results which will perform the best algorithm performance. Setting the best parameters provided by the “Optimize Parameters (Grid)” was executed the Deep Learning workflow. In order to estimate performances the "Performance (Regression)" operator (see last block of Fig. 2) has been adopted. This block provides as output the following parameters useful for the comparison:

- Average absolute error: it represents the average absolute deviation of the forecast from the actual value (it is expressed in dollars units);
- Average relative error: it is calculated as the average of the fraction that sees in the numerator the error in absolute value between the predicted values and the respective real values and the denominator the real value (it is expressed in percentage);
- Correlation: it provides a percentage correlation value between predicted and actual values in a range between 0 and 100 where 100 represents the perfect prediction of data by the model (it is expressed in percentage).

Below are reported the performance values found on some datasets regarding the departments of the shop "1", using 2000 training periods for the model:

Table 1. Results comparison results of the Deep Learning algorithm.

Departments	Absolute Error (\$)	Relative Error (%)	Correlation (%)
3	2615.523 +/- 2547.691	15.25% +/- 11.70%	97,4%
7	2298.615 +/- 1505.329	13.63% +/- 9.81%	73%
9	5113.234 +/- 3994.312	15.60% +/- 9.76%	85,1%
11	2061.526 +/- 973.173	14.52% +/- 10.25%	81,5%
27	658.316 +/- 305.240	12.61% +/- 9.20%	80,5%
34	1138.421 +/- 698.660	10.08% +/- 5.18%	82,5%

From Fig. 8 to Fig. 13 are illustrated some output results concerning the comparison with real sales data and predictive ones versus date, according with the cases of Table 1. The results must be read as follow: if the prediction is similar to the real data concerning similar past periods the sales will follows the same trend, otherwise will occur a trend variation. In the analysed cases all the predictions will follows the trend of the corresponding real sales values by observing a different ripple behaviour. In the particular cases of Fig. 8, Fig. 9 and Fig. 11 are observed a decrease of sales.

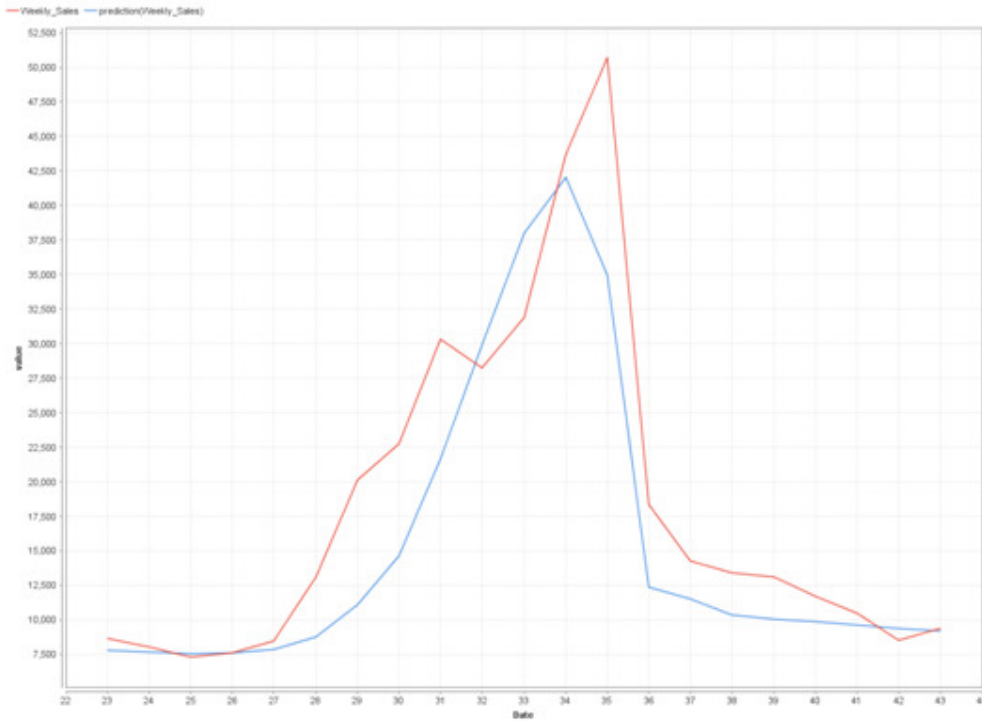


Figure 8. Department 3: predicted weekly sales (blue plot) compared with real sales (red plot).

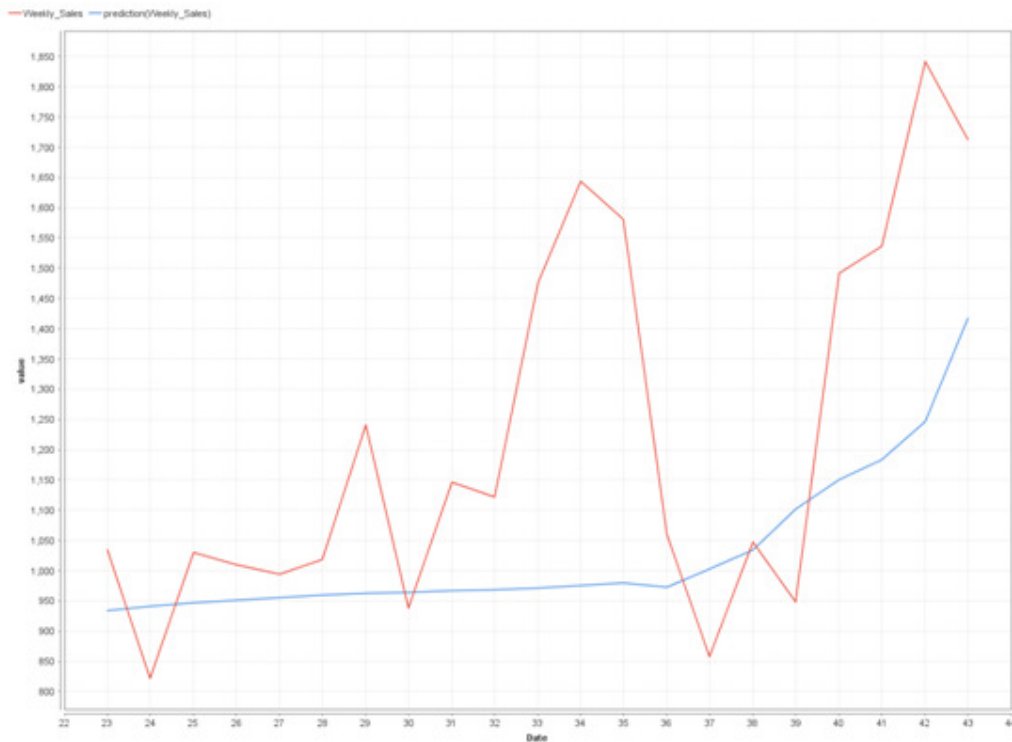


Figure 9. Department 9: predicted weekly sales (blue plot) compared with real sales (red plot).



Figure 10. Department 7: predicted weekly sales (blue plot) compared with real sales (red plot).

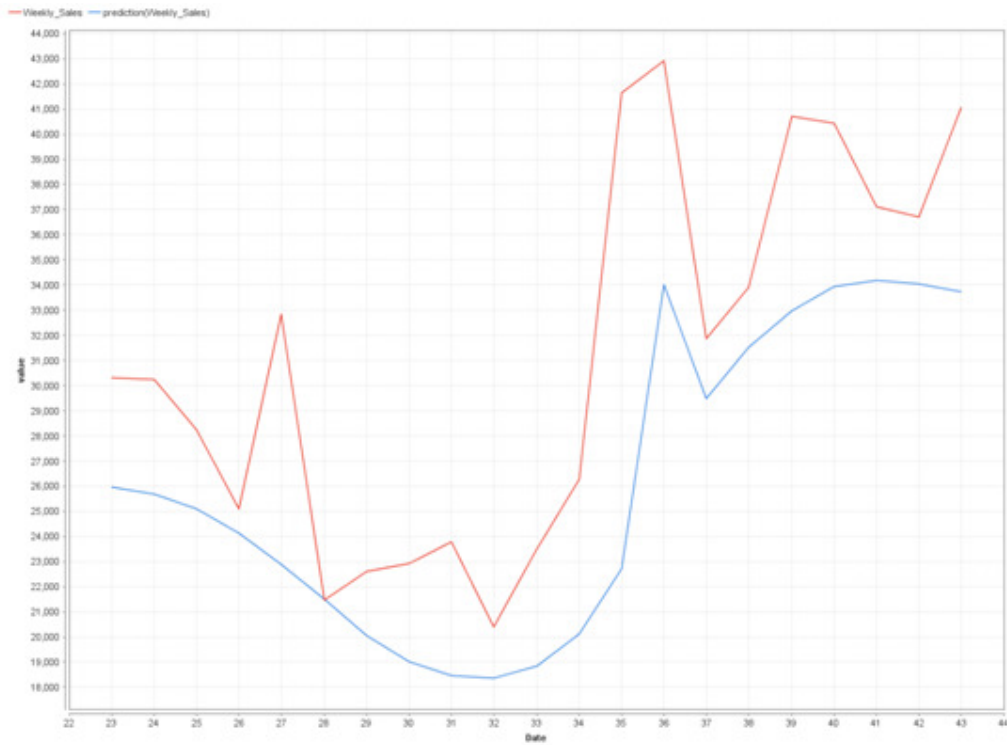


Figure 11. Department 27: predicted weekly sales (blue plot) compared with real sales (red plot).

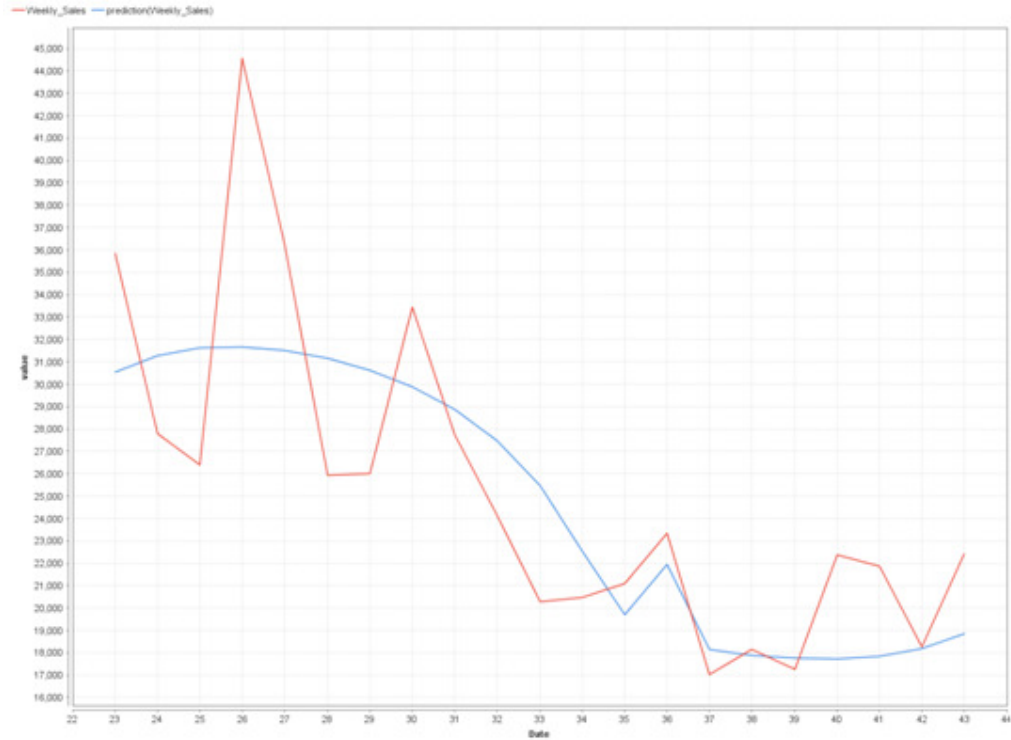


Figure 12. Department 11: predicted weekly sales (blue plot) compared with real sales (red plot).



Figure 13. Department 34: predicted weekly sales (blue plot) compared with real sales (red plot).

The proposed plots behave as graphical dashboards thus providing a decision support system oriented on strategic marketing and business intelligence: the sales prediction can be considered

in order to plan warehouse stock replenishment, to propose promotions or to change in general the marketing strategy.

3.2. Performance comparison of different data mining algorithms predicting sales

In this section is validated the choice of the ANN algorithm by comparing performance results of different data mining algorithms able to predict sales. We describe below the compared algorithms.

A Decision Tree algorithm implements a tree like collection of nodes generating a decision on values affiliation to a class where each node behaves as a splitting rule for a defined attribute. The algorithm is mainly based on the classification process separating values belonging to different classes: for regression it separates values in order to reduce the error in an optimal way for the selected parameters representing recursive classification criteria. During the data process the building of new nodes is repeated until the achievement of the stopping criteria setted before. The prediction is performed on the class label attribute.

Another algorithm adopted for the comparison is the k-Nearest Neighbor algorithm. This method is based on the comparison of an unknown dataset with a k training dataset representing the nearest neighbors of the unknown dataset. Different approaches such as the Euclidean distance can be used to calculate the distance between the unknown dataset and the training one. Concerning data pre-processing it is recommended to normalize data before the launch of k-Nearest Neighbor algorithm. The k-Nearest Neighbor algorithm will classify the unknown dataset by assigning a majority score of the found neighbors.

Also the Random Forest method as been applied for the comparison. This approach is an ensemble of a certain number of random trees. As for Decision Tree, each node of a tree represents a splitting rule for one specific attribute. Only a sub-set of attributes, specified with the subset ratio criterion, is considered for the splitting rule selection. This approach is suitable for the classification process by separating values belonging to different classes until a stop criterion occurs. Each random tree generates a prediction for each input dataset by following the branches of the tree according with the splitting rules and evaluating the 'leaf' positions and conditions. The resulting model is a model obtained from a scoring process of all the created random trees. In this approach the 'pruning' procedure can reduce the model complexity. Important parameters to tune the data processing are the minimal leaf size and split ratio. A good parameter choice for the classification data processing is minimal leaf size of 2. The last algorithm executed for the performance comparison is the Support Vector Machine –SVM-. Here is a basic description of the SVM. The standard SVM read an input dataset and predicts specific attributes. SVM approach behaves as a non-probabilistic binary linear classifier. Given a set of training dataset the SVM training algorithm will build a model that assigns new dataset into different categories. An SVM model is the data representation of the dataset as points in space. These points are mapped so that the dataset values appartaining to separate categories are divided by clear gaps. The SVM constructs a hyperplane or set of hyperplanes in a high- or infinite- dimensional space, suitable for classification or regression. The mapping used by the SVM schemes are designed to ensure that dot products may be processed in the original space, by defining them in terms of a kernel function $K(x,y)$.

Below are the results obtained, testing the different algorithms and making an average of the 5 best results on 20 possible model configurations.

Table 2. Performance comparison of different data mining results.

Algorithm	Correlation (min / max %)	Average Absolute Error (\$)	Relative Average Error (%)
1: Deep Learning (Neural Network)	73% / 97,4%	2000 +/- 1250	12,9% +/- 9,9%
2: Gradient Boosted Trees	51% / 76%	4200 +/- 4300	13% +/- 11%
3: SVM	45% / 55%	47000 +/- 22000	100%
4: k-NN	58% / 62%	8900 +/- 5400	28.5% +/- 11.5%
5: Decision Trees	44% / 69%	13200 +/- 6600	25.33% +/- 12%
6: Random Forest	59% / 65%	7500 +/- 7500	14% +/- 11%

In Fig.14 are summarised the results of Table 2 concerning Correlation and relative average error. In Fig. 15 is reported the comparison of average absolute errors. Table 2, Fig. 14 and Fig. 15 prove that Deep Learning algorithm is best method for predicting sales.

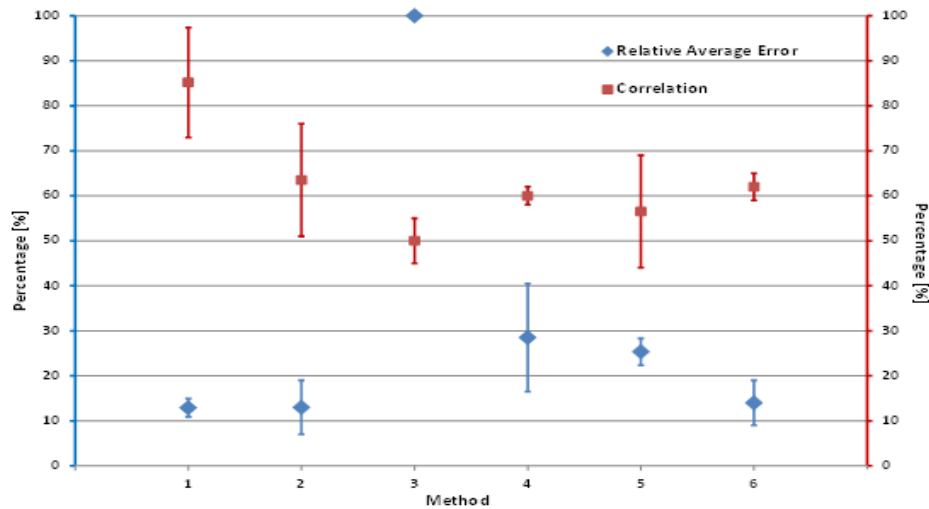


Figure 14. Comparison of correlations and relative average errors versus the methods indicated in table 2.

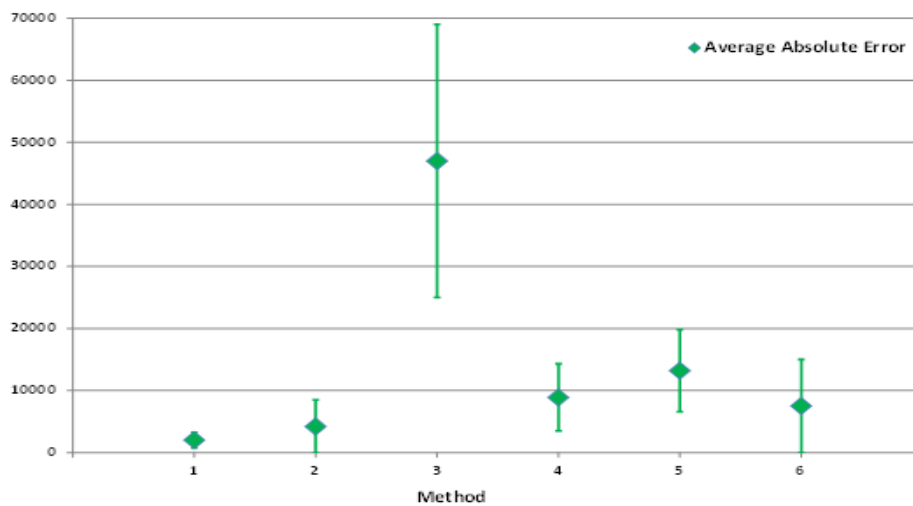


Figure 15. Comparison of average absolute errors versus the methods indicated in table 2

Accuracies of the compared algorithms are represented by the error bars of Fig. 14 and Fig. 15: the smaller error bar amplitude will correspond to the best accuracy. Concerning computational cost only few seconds are necessary to execute all the workflow algorithms (from 0.5 sec. to 1 min. and 30 sec.). The workflows have been executed by a laptop having the following characteristics: Intel(R) 122 Core(TM) i3-403U CPU, 1.9 GHz, 8 GB RAM, 64 bit operative system. In Fig. 16 and Fig. 17 are illustrated some example of output screenshot indicating computational costs of the Deep Learning algorithm and other parameters such as training speed. The good accuracy and computational cost of the Deep Learning algorithms proves that it can be adopted also in other recent application fields such as tourism [27] and stock trend prediction [28].

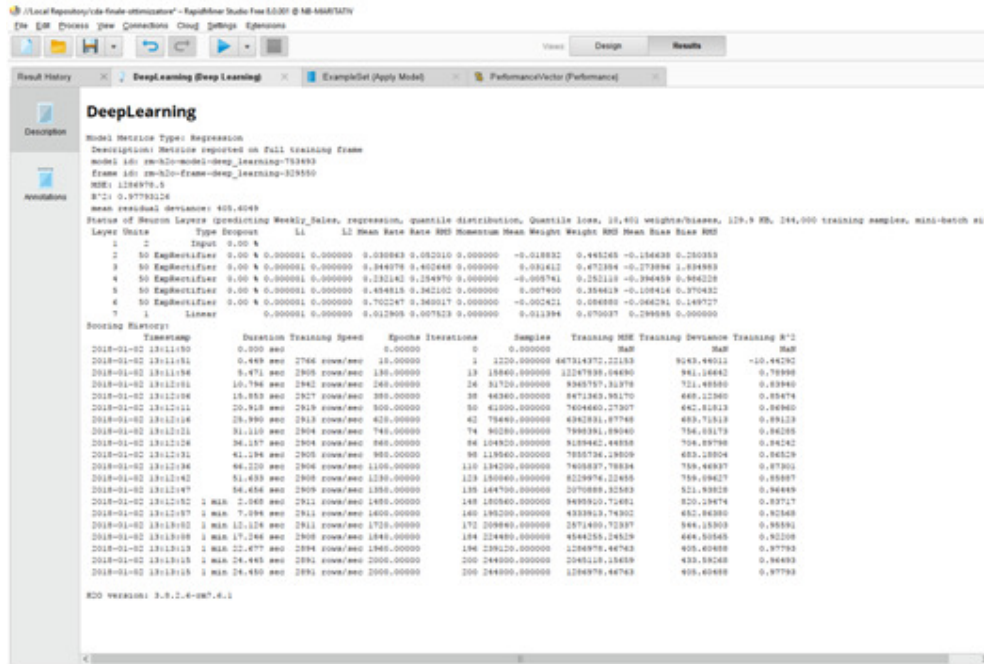


Figure 16. Computational costs of deep learning algorithm.

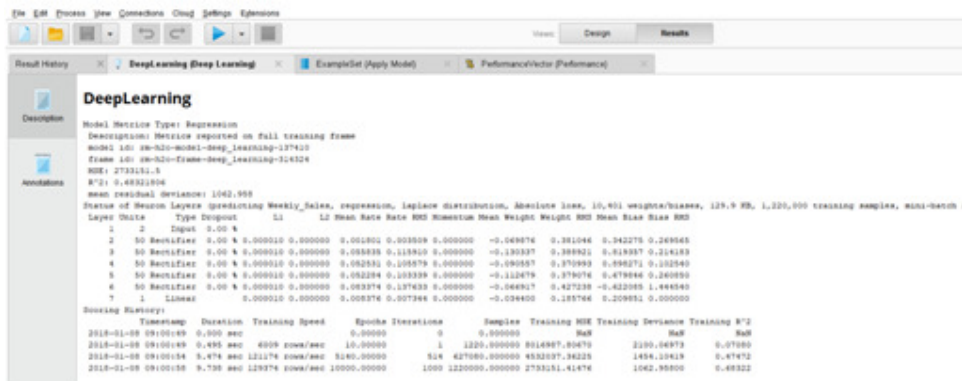


Figure 17. Computational costs of deep learning algorithm.

4. CONCLUSION

The goal of the proposed paper is the study of a complete workflow predicting sales. The discussion has been developed following the main steps of data pre-processing and data treatment. The applied approaches are able to address the reader to understand and to apply RapidMiner workflow. The study followed a typical problem concerning the processing of a data source having more information where only some attributes are important for the prediction. The proposed study supports the reader to learn the approaches used for data pre-processing which characterize a predicting data mining process. The results proves that Deep Learning algorithm based on artificial neural network is the best method for predicting sales. Concerning this method an high correlation and low absolute and relative errors are observed. The Gradient Boosted Trees algorithms provides a good performance if compared with other approaches such as SVM, k-NN, Decision Trees and Random Forest and could represent an alternative approach able to predict sales. The study has been developed within the framework of an industrial project oriented on the data processing of output data of different software such as CRM and ERP. The model can be also applied in other different fields by considering other dataset types to process and other application scenarios.

ACKNOWLEDGEMENTS

The work has been developed in the frameworks of the Italian projects: “Algoritmi di Intelligenza Artificiale Applicati agli Output dei Software Gestionali per la Predizione di vendite: ‘SALES INTELLIGENCE’ [Artificial Intelligence Algorithms for Sales Prediction Applied to the Outputs of Management Software: ‘SALES INTELLIGENCE’]”. The authors would like to thank the following collaborators of the research team: G. Birardi, A. Colonna, R. Cosmo, M. Le Grottaglie, A. Leogrande, A. Lombardi, A. Lorusso, S. F. Massari, G. Meuli, D. D. Romagno, P. Scagliusi, and V. Vitti.

REFERENCES

- [1] Penpece D., & Elma O. E. (2014) “Predicting Sales Revenue by Using Artificial Neural Network in Grocery Retailing Industry: A Case Study in Turkey”, *International Journal of Trade Economics and Finance*, Vol. 5, No. 5, pp435-440.
- [2] Thiesing F. M., & Vornberger, O. (1997) “Sales Forecasting Using Neural Networks”, *IEEE Proceedings ICNN'97, Houston, Texas, 9-12 June 1997*, pp2125-2128.
- [3] Zhang, G. P. (2003) “Time series forecasting using a hybrid ARIMA and neural network model”, *Neurocomputing*, Vol. 50, pp159–175.
- [4] Sharma, A., & Panigrahi, P. K. (2011) “Neural Network based Approach for Predicting Customer Churn in Cellular Network Services”, *International Journal of Computer Applications*, Vol. 27, No.11, pp0975–8887.
- [5] Kamakura, W., Mela, C. F., Ansari A., & al. (2005) ” Choice Models and Customer Relationship Management,” *Marketing Letters*, Vol. 16, No.3/4, pp279–291.
- [6] Smith, K. A., & Gupta, J. N. D. (2000) “Neural Networks in Business: Techniques and Applications for the Operations Researcher,” *Computers & Operations Research*, Vol. 27, No. 11–12, pp1023-1044.
- [7] Chattopadhyay, M., Dan, P. K., Majumdar, S., & Chakraborty, P. S. (2012) “Application of Artificial Neural Network in Market Segmentation: A Review on Recent Trends,” *Management Science Letters*, Vol. 2, pp425-438.

- [8] Berry, J. A. M., & Linoff, G. S. (2004) "Data Mining Techniques For Marketing, Sales, and Customer Relationship Management", Wiley, Second Edition.
- [9] Buttle, F. (2009) "Customer Relationship Management Concepts and Technologies", Elsevier, Second Edition.
- [10] Thomassey, S. (2014) "Sales Forecasting in Apparel and Fashion Industry: A Review", Springer, chapter 2.
- [11] Massaro, A. Barbuzzi, D., Vitti, V., Galiano, A., Aruci, M., Pirlo, G. (2016) "Predictive Sales Analysis According to the Effect of Weather", Proceeding of the 2nd International Conference on Recent Trends and Applications in Computer Science and Information Technology, Tirana, Albania, November 18 - 19, pp53-55.
- [12] Parsons, A.G. (2001), "The Association between Daily Weather and Daily Shopping Patterns", Australasian Marketing Journal, Vol. 9, No. 2, pp78-84.
- [13] Steele, A.T., (1951) "Weather's Effect on the Sales of a Department Store", Journal of Marketing Vol. 15, No. 4, pp436-443.
- [14] Murray, K. B., Di Muro, F., Finn, A., & Leszczyc, P. P. (2010) "The Effect of Weather on Consumer Spending", Journal of Retailing and Consumer Services, Vol. 17, No.6, pp512-520.
- [15] Massaro, A., Galiano, A., Barbuzzi, D., Pellicani, L., Birardi, G., Romagno, D. D., & Frulli, L., (2017) "Joint Activities of Market Basket Analysis and Product Facing for Business Intelligence oriented on Global Distribution Market: examples of data mining applications," International Journal of Computer Science and Information Technologies, Vol. 8, No.2 , pp178-183.
- [16] Aguinis, H., Forcum, L. E., & Joo, H. (2013) "Using Market Basket Analysis in Management Research," Journal of Management, Vol. 39, No. 7, pp1799-1824.
- [17] Štulec, I, Petljak, K., & Kukor, A. (2016) "The Role of Store Layout and Visual Merchandising in Food Retailing", European Journal of Economics and Business Studies, Vol. 4, No. 1, pp139-152.
- [18] Otha, M. & Higuci, Y. (2013) "Study on Design of Supermarket Store Layouts: the Principle of "Sales Magnet"", World Academy of Science, Engineering and Technology, Vol. 7, No. 1, pp209-212.
- [19] Shallu, & Gupta, S. (2013) "Impact of Promotional Activities on Consumer Buying Behavior: A Study of Cosmetic Industry", International Journal of Commerce, Business and Management (IJCBM), Vol. 2, No.6, pp379-385.
- [20] Al Essa, A. & Bach, C. (2014)" Data Mining and Knowledge Management for Marketing", International Journal of Innovation and Scientific Research, Vol. 2, No. 2, pp321-328.
- [21] Kotu, V., & Deshpande B. (2015) "Predictive Analytics and Data Mining- Concepts and Practice with RapidMiner" Elsevier.
- [22] Wimmer, H., Powell, L. M. (2015) "A Comparison of Open Source Tools for Data Science", Proceedings of the Conference on Information Systems Applied Research. Wilmington, North Carolina USA.
- [23] Al-Khoder, A., Harmouch, H., "Evaluating Four Of The most Popular Open Source and Free Data Mining Tools", International Journal of Academic Scientific Research, Vol. 3, No. 1, pp13-23.
- [24] Gulli, A., & Pal, S. (2017) "Deep Learning with Keras- Implement neural networks with Keras on Theano and TensorFlow," Birmingham -Mumbai Packt book, ISBN 978-1-78712-842-2.

- [25] Kovalev, V., Kalinovsky, A., & Kovalev, S. (2016) “Deep Learning with Theano, Torch, Caffe, TensorFlow, and deeplearning4j: which one is the best in speed and accuracy?” Proceeding of XIII Int. Conf. on Pattern Recognition and Information Processing, 3-5 October, Minsk, Belarus State University, pp99-103.
- [26] “Walmart Recruiting - Store Sales Forecasting” 2018. [Online]. Available: <https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting/data>
- [27] Huang, H.-C. & Hou, C.-I.. (2017) “Tourism Demand Forecasting Model Using Neural Network”, International Journal of Computer Science & Information Technology (IJCSIT), Vol. 9, No. 2, pp19-29.
- [28] Kalyani, J., Bharathi, H. N., & Rao, J. (2016) “Stock Trend Prediction Using News Sentiment Analysis”, International Journal of Computer Science & Information Technology (IJCSIT), Vol. 8, No. 3, pp67-76.

Corresponding Author

Alessandro Massaro: Research & Development Chief of Dyrecta Lab s.r.l.

