EMBEDDING RAPIDMINER MODELS IN JAVA CODE

ZHANAT SERGAZIEVNA SAUKHANOVA
L N Gumilyov Eurasian National University, Mirzoyan Street, 2, Astana, 010008, Kazakhstan
MAGRIPA SERGAZIEVNA SAUKHANOVA
L N Gumilyov Eurasian National University, Mirzoyan Street, 2, Astana, 010008, Kazakhstan
OMARBEKOVA ASSEL SAILAUBEKOVNA
L N Gumilyov Eurasian National University, Mirzoyan Street, 2, Astana, 010008, Kazakhstan

ABSTRACT

Intelligent data analysis, also known as Data Mining, is implemented for a search for new and useful knowledge in large data arrays. Data Mining models can be easily built by means of visual graphic interface of Rapid Miner open-ended software. RapidMiner provides operators for the most well-known algorithms and methods for data analysis and machine learning, including input and output, data pre-processing and visualization. The presented paper discusses problems of embedding of models, which were created using Rapid Miner, into Java codes of other applications.

Keywords: *Data Mining, Rapidminer Operators, Read CSV, Decision Tree, Apply Model, Rapidminer Classes, Java Code.*

1. INTRODUCTION

Nowadays huge amounts of data are stored and processed electronically. In many cases, those data arrays are not used and not analyzed. Search, description and structuring of patterns in those data flows require special mathematical and algorithmic approaches. Currently such approaches are intensively developed in the direction, which is known as intelligent data analysis or DataMining [1].

Methods of DataMining are mainly used for a solution of problems of classification and regression, a search for associative rules and clustering. The classification problem consists of definition of an object's class by its characteristics. The regression problem allows to define a value of a certain object's parameter using its known characteristics. In contrast to the classification problem a parameter's value is not finite set of classes, but a set of real numbers. An aim of a search for associative rules is finding of partial dependencies between objects or events. The clustering problem consists of a search for groups (clusters) and their characteristics in a whole set of analyzed data [2]. Clustering is a key tool for data analysis it is used in many fields. Main K-means methods is widely discussed and implemented in many applications [3].

DataMining on the Internet is used every time, when someone uses Google, or other search engines. The Internet contains a lot of useful data, which can be used by organizations and users in order to improve the decision-making process. On the Internet the major part of web sources provide data in semi-structured form. Access, search and use of data from various sources requires integration of web data. Integration of web data is a complex process because of open, dynamic and heterogeneous nature of web data. Various architectures of integration systems for web data are developed and suggested [4].

In recent years a role of marketing communications increases. Marketing communications allows to transfer messages to consumers in order to make companies' products and services more attractive for a target audience. In one of the previous studies a model for neuron network-classifier for assignment of advertisement for a target group is presented [5].

The presented paper is organized as follows: issues of a creation of model for a classification in RapidMiner environment are discussed and detailed information on embedding of that model in Java code is presented.

2. METHODOLOGY

A process of knowledge extraction from data follows the same schematic as an establishment of laws of physics. Physics is based on experimental studies of phenomena, and the aim of physics is a formulation of laws, which describe those phenomena. Physical studies are based on observation. Synthesis of observations allows to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

formulate hypotheses about general terms of those phenomena. Hypotheses are verified by means experiment. Analysis of experimental data allows to formulate mechanisms. At the first stages of a study mechanisms are predominantly of empirical nature. Analysis of mechanisms and parameters allows to create physical theories that help to explain studied phenomena. Physical theories, in turn, provide a basis for accurate experiments, which are, generally, used to determine limits of their application [6].

Implementation of that approach for DataMining allows to determine methodology and steps of data analysis process for an extraction of new knowledge. The process can be divided into the following stages: data acquisition, data structuring, search for a model, which explains existing data, the model verification using testing data, implementation of the model.

One of the popular software products providing a creation of data analysis models by means of Data Mining, as well as experimental capabilities, is Rapid Miner, produced by company having the same name, which possesses a set of algorithms and methods for data processing and analysis.

2.1. Method For Decisions' Tree Derivation For The Classification Problem

Decision trees method is one of the most popular for a solution of the classification problem and prognosis. The basis of all known methods of decision trees' derivation is Hunt algorithm [7].

Presuming that there are a large number of objects $I = \{i_1, i_2, \dots, i_n\}$. Each object is described by a set of variables:

$$i_j = \{x_1, x_2, \dots, x_h, \dots, x_m, y\},$$

where x_h are independent variables, which values are known, on their basis a value of dependent variable y is determined. Defining a set of independent variables in a form of a vector:

$$X = \{x_1, x_2, \dots, x_h, \dots, x_m\}.$$

Each variable x_h can have values from a certain set of:

$$C_h = \{c_{h1}, c_{h2}, \dots\}.$$

Dependent variable y has a value from a variety $C = \{c_1, c_2, \dots, c_r, \dots, c_k\}$.

Hunt algorithm of decision trees' derivation based on "divide and rule" methodology. That method is based on recursive division of a training sample into subsets containing objects, which belong to the same classes.

The study of Quinlan [8] contains a description of the algorithm ID3 for synthesis of decision trees.

Presuming that T is a set of training records and $\{c_1, c_2, \dots, c_k\}$ is a set of classes. Three cases are possible:

- set T contains one or more objects belonging to one class c_j. Decision tree for T is a leaf determining the class c_j;

- set T does not contain any object. Then it is again a leaf, and the class that is associated with a leaf, is selected from another set, which is different from T;

- set T contains objects belonging to different classes. In that case set T breaks down into certain subsets. For that an independent variable x_h is selected, it has values of $c_h^1, c_h^2, \dots, c_h^n$. Set T is divided into subsets $T_1, T_2, \dots, T_n$, where each subset T_i contains all objects having values of c_h^i for a selected attribute. That procedure will continue in a recursive manner until a final set will not have only objects of the same class.

2.2. Development Of Data Analysis Model In Rapid Miner Environment

Let's discuss a method for decision tree deviation for the classification problem. The task is as follows: by means of analysis of clients' activities on a company's web site, determining a probability of buying a new product. Clients of a company are divided into "Innovator", "EarlyAdopter", "EarlyMajority" and "LateMajority" classes depending on terms of of buying a new product. A structured set of data is read from the two Excel files: the first file is used to create a classification model, i.e.the file contains a set of data for training, the created model will be applied for data of the second file. In the presented example, the Excel file has the following headings: User_ID – user's id; Gender – user's gender; Age – age of user; Marital_Status – has two values, "M" – if married, "S" – single; Website_Activity – has three different values: Seldom, Regular, or Frequent; Browsed_Electronics_12Mo – has two values Y/N, depending on viewing of electronic products on a company's website in the past year by user; Bought_Electronics_12Mo – has two values Y/N, depending on buying of electronics through the company's website by user; Bought_Digital_Media_18Mo – has two values Y/N, depending on the buying of digital media (such as MP3 player, excluding electronic book) in the last year by user; Bought_Digital_Books – has two values Y/N, depending on whether user ever bought an electronic book; Payment_Method – has four different values such as bank transfer, Website account, Credit card, Monthly billing; eReader_Adoption – has four different values such as, "Innovator", "Early Adopter", "Early majority" and "Late majority". That column has only training file. Number of records in a training set is 661 [9].

For the task of classification in Rapid Miner software let's use Decision Tree operator. At input operator is provided with a training sample, whic will be used for training. At output operat produces a trained model, which after that can l applied to for new data sets, by means of operat Apply Model for prognosis and a creation of decision tree. Decision Tree operator has tl following main parameters [10]:

- division criterion: information gain, gain ratio, Gini index, accuracy [11];

- minimum size for branching. If a size of subtree is less than a value of a parameter, branching is stopped for that node and "leaf formation" takes place;

- minimum size of a leaf. Each leaf has a size, which is equal to a number of objects in a leaf. If a size of a leaf after a division of a tree or a subtree will be less than a specified value of a parameter, branching process is stopped for a corresponding subtree;

- minimum increase of data. A high value of that parameter will lead to a smaller branching of a tree. If a value of that parameter is too high, there is a possibility that a tree will not branch at all;

- maximum depth. That parameter is used for a restriction of a size of a final decision tree.

Figure 1 shows the classification model and its implementation for the aforementioned problem.

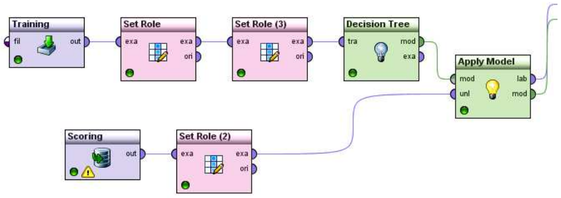


Figure 1. An Implementation Of The Model For A Prognosis Of Term Of Buying Of New Products [9].

In order to embed the model into Java code let's substitute Training and Scoring operators for Read CSV operator. That operator is used for reading of data from Excel files of csv format [12]. In turn, Read CSV operator will lead to a deletion of SetRole operators.

Thus, we obtained model presented in Figure 2.

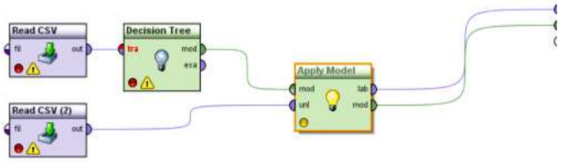


Figure 2. Transformation Of The Model For Embedding Into Java Code.

Saving the created model in DTree.rmp file.

2.3. Evaluation Of The Model's Efficiency In Rapidminer Environment

Standard practice in an evaluation of a model's efficiency consists in a division of the used data into a data set, which is used for training, and a data set for testing. Usually training set contains 70-80% of initial data, other data is used for testing [13].

Rapid Miner contains approximately 30 operators for an evaluation of a model's efficiency. One of evaluation operators is X-Validation. X-validation is a nested operator. It consists of two subprocesses: training subprocess and testing subprocess. Training subprocess is used for training and model creation. After a training model is implemented in testing subprocess. A number of records which was correctly and incorrectly predicted by a model is a basis for of a calculation of an evaluation of efficiency of a classification model. On that basis confusion matrix is created. Matrix element f_{ij} represented a number of records of i class,which were predicted by a model as a record of j class. For confusion matrix with 2x2 size an accuracy of a model is calculated using formula (1) [14]:

$$Accuracy = \frac{Number\ or\ correct\ predictions}{Total\ number\ of\ predictions} = \frac{f_{11} + f_{00}}{f_{11} + f_{10} + f_{01} + f_{00}} \quad (1)$$

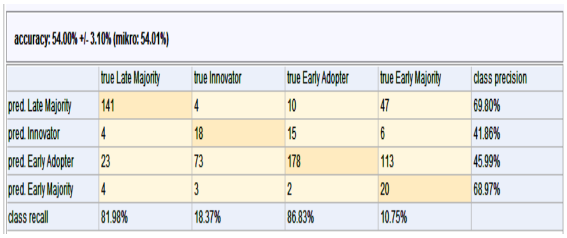
Similarly, a productivity of a model can be represented in terms of frequency of error, which is defined using the following formula (2):

$$Error\ rate = \frac{Number\ of\ wrong\ predictions}{Total\ number\ of\ predictions} = \frac{f_{10} + f_{01}}{f_{11} + f_{10} + f_{01} + f_{00}} \quad (2)$$

Performance operator can be used for a calculation of a model's accuracy.

Let's calculate an evaluation of efficiency of a model by means of two following division criteria: Gain ratio and Gini index.

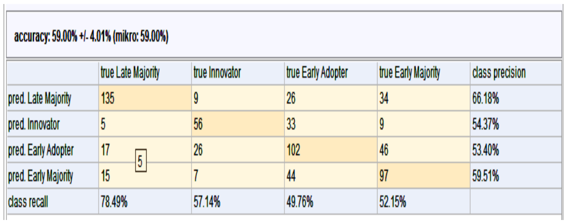
A confusion matrix and an accuracy of a model for corresponding criteria of a division is presented in Figures 3 and 4.



accuracy: 54.00% +/- 3.10% (mikro: 54.01%)

	true Late Majority	true Innovator	true Early Adopter	true Early Majority	class precision
pred. Late Majority	141	4	10	47	69.80%
pred. Innovator	4	18	15	6	41.86%
pred. Early Adopter	23	73	178	113	45.99%
pred. Early Majority	4	3	2	20	68.97%
class recall	81.98%	18.37%	86.83%	10.75%	

Figure 3. An Accuracy Of A Model And A Confusion Matrix For Gain Ratio Division Criterion.



accuracy: 59.00% +/- 4.01% (mikro: 59.00%)

	true Late Majority	true Innovator	true Early Adopter	true Early Majority	class precision
pred. Late Majority	135	9	26	34	66.18%
pred. Innovator	5	56	33	9	54.37%
pred. Early Adopter	17	26	102	46	53.40%
pred. Early Majority	15	7	44	97	59.51%
class recall	78.49%	57.14%	49.76%	52.15%	

Figure 4. An Accuracy Of A Model And A Confusion Matrix For Gini Index Division Criterion.

In the study of Kovahi [19] it is demonstrated that an accuracy of a model for a cross-validation is calculated using the equation (3):

$$acc_{cv} = \frac{1}{n} \sum_{<v_i, y_i> \in D} \delta(\mathrm{I}(D \backslash D_{(i)}, v_i), y_i) \, . \quad (3)$$

where $\delta(i,j) = 1$, if $i = j$, and in other cases is equal to 0, $\mathrm{I}(D \backslash D_{(i)}, v_i)$ means, that classifier I for a tested set of data D\D$_i$ is applied for a record v_i, which was not marked with a class.

Dispersion of an evaluated accuracy is calculated using the equation (4):

$$\bar{\sigma}^2 = acc_{cv}(1 - acc_{cv})/n \, , \quad (4)$$

where n is a number of records in a set of data.

A prognosis of belonging of tested records to certain classes can be considered as a binomial experiment. Presuming that a tested set contains N records, X is a number of correctly predicted records and p is a confident accuracy of a model. In a case of modeling of a prognosis task as a binomial experiment, X has a binomial distribution with an average value of Np and dispersion Np(1-p). It is known that empiric accuracy acc=X/N also has a binomial distribution with an average value of p and dispersion p(1-p)/N. Although binomial distribution can be used for an evaluation of a confidence interval for an accuracy of a model acc, it is often approximated by means of normal distribution for a sufficiently large N. Confidence interval for an accuracy of a model is calculated using the following equation (5):

$$\frac{2n * acc + Z^2_{\alpha/2} \pm Z_{\alpha/2}\sqrt{Z^2_{\alpha/2} + 4n * acc - 4n * acc^2}}{2(n + Z^2_{\alpha/2})} \quad (5)$$

Let's discuss an issue of statistical significance of an observed deviation of models accuracies. For that let's use k-tuple cross-validation. A set of data is divided into k equal parts. Implementing each classifier for a creation of a model for k-1 parts and testing it for a remained k part. That stage is repeated k times, each time using other part of a testing set.

Presuming that there are M1i model with Decision Tree classifier and Gini index criterion and M2i model with Decision Tree classifier and gain ratio criterion for iteration I of k-tuple cross-validation. The difference between error frequencies is as follows:

$$d_i = e_{1i} - e_{2i}$$

Presuming that $\bar{d}$ is an average value of differences of error frequencies and $\bar{\sigma}$ is dispersion calculated using the equation (6):

$$\bar{\sigma}^2 = \frac{\sum_{i=1}^{k}(d_i - \bar{d})^2}{k - 1} \quad (6)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A deviation of average $\bar{d}$ from an expected value, which is zero, is verified using the statistics (7):

$$t = \frac{|\bar{d}|}{\bar{\sigma}} \sqrt{k} \qquad (7)$$

For a two-sided evaluation of a significance level α value t is compared with a value $t_{(1-\alpha/2),k-1}$, which is obtained from the table t-of t distribution. For $t > t_{(1-\alpha/2),k-1}$ a difference between frequencies of evaluations for various models can be stated.

Using T-test operators for a calculation of t-statistics. A process of a calculation of t-statistics is presented in Figures 5-7.

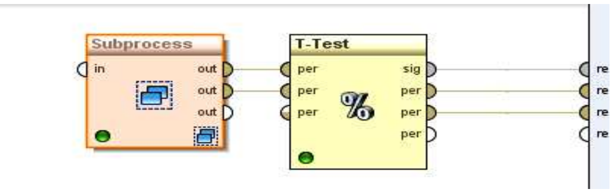


Figure 5. The Main Process For A Calculation Of T-Statistics.

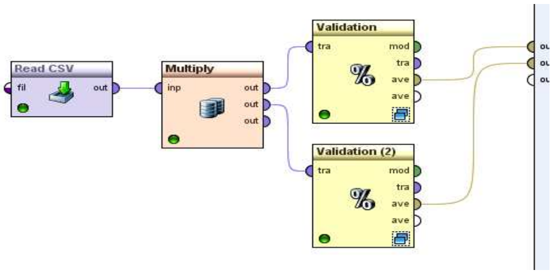


Figure 6. Subprocess For An Organization Of Cross-Validation.

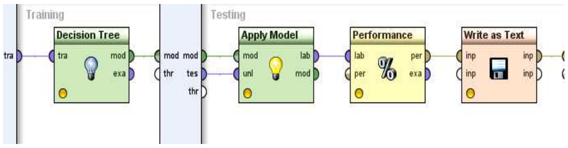


Figure 7. Subprocess Of Training, Testing And Efficiency Calculation.

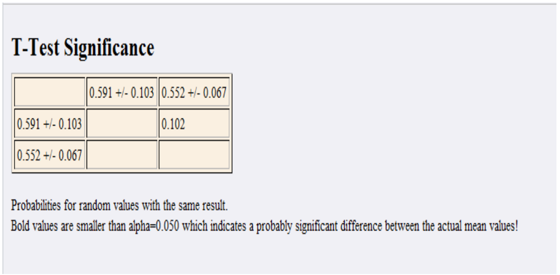
In order to obtain intermediate results in a subprocess of testing Write as Text operator was used. By means of that operator results of iterations of K-tuple cross-validation are written in a file.

Table 1 presented intermediated results of a calculation of classification_error parameter for 30-tuple cross-validation of Decision Tree classifier with Gini index and gain ratio criteria.

Table 1. Intermediate Results Of Cross-Validation.

i	classification_error		*i*	classification_error	
	gini index	**gain ratio**		**gini index**	**gain ratio**
1	36.36%	54.55%	**16**	30.43%	39.13%
2	45.45%	36.36%	**17**	45.45%	40.91%
3	54.55%	40.91%	**18**	40.91%	40.91%
4	40.91%	54.55%	**19**	59.09%	40.91%
5	59.09%	40.91%	**20**	31.82%	50.00%
6	31.82%	50.00%	**21**	36.36%	45.45%
7	45.45%	36.36%	**22**	45.45%	36.36%
8	50.00%	50.00%	**23**	45.45%	54.55%
9	27.27%	36.36%	**24**	54.55%	54.55%
10	31.82%	50.00%	**25**	59.09%	40.91%
11	40.91%	36.36%	**26**	40.91%	45.45%
12	13.64%	40.91%	**27**	31.82%	45.45%
13	40.91%	40.91%	**28**	27.27%	50.00%
14	40.91%	50.00%	**29**	40.91%	59.09%
15	36.36%	45.45%	**30**	40.91%	36.36%

t-statistics is calculated for 30-tuple validation and with significance level of $\alpha = 0.05$, the result are presented in Figure 8.



T-Test Significance

	0.591 +/- 0.103	0.552 +/- 0.067
0.591 +/- 0.103		0.102
0.552 +/- 0.067		

Probabilities for random values with the same result.
Bold values are smaller than alpha=0.050 which indicates a probably significant difference between the actual mean values!

Figure 8. Results of t-statistics calculation.

An increase of repetition factor of cross-validation increase a probability of error of 1st type, i.e. a deviation from zero hypothesis, while actually it shouldn't be declined. In a case observed difference between methods is statistically significant, an additional ANOVE (ANalysis Of VAriance) test can be implemented in order to define, whether zero hypothesis should be refuted or not. The process of connecting of ANOVA test is presented in Figure 9.

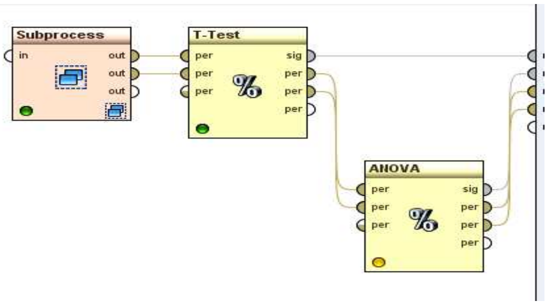
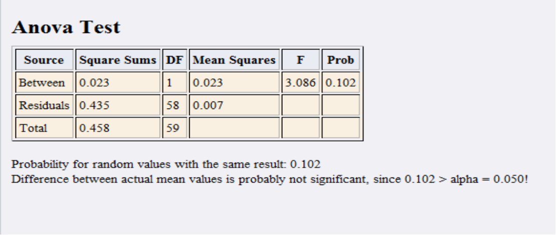


Figure 9. The Process Calculations Of T-Test, And Anova Test

The results of Anova test are presented in Figure 10.



Anova Test

Source	Square Sums	DF	Mean Squares	F	Prob
Between	0.023	1	0.023	3.086	0.102
Residuals	0.435	58	0.007		
Total	0.458	59			

Probability for random values with the same result: 0.102
Difference between actual mean values is probably not significant, since 0.102 > alpha = 0.050!

Figure 10. The Results Of Anova Test.

As a result of aforementioned, in further implementing a model, which was created by means of Decision Tree operator with Gini index division criteria.

2.4. Embedding Of The Model Into Java Application

For specialists developing various information system an integration of systems with models built using RapidMiner could provide many advantages from analytical point of view.

2.4.1. Reading and pre-processing of data

Java application will be created by means of Eclipse integrated development environment [15]. For an embedding of a model, in the first place, connecting rapidminer.jar library to the project. Table 2,represents formats of main classes of RapidMiner, which will be used for further coding [10]:

Table 2. Formats Of Constructors And Methods

No.	Imported package	Class, interface	Format of constructor, method
1	com.rapidminer.Process	public class Process extends java.lang.Object implements java.lang.Cloneable	public Process(java.io.File file) throws java.io.IOException , XMLException
2	com.rapidminer.Process	public class Process extends java.lang.Object implements java.lang.Cloneable	public Operator getOperator(java.lang.String name)
3	com.rapidminer.Process	public class Process extends java.lang.Object implements java.lang.Cloneable	public final IOContainer run() throws OperatorException
4	com.rapidminer.operator.Operator	public abstract class Operator extends java.lang.Object implements ConfigurationListener, PreviewListener, LoggingHandler, ParameterHandler	public void setParameter (java.lang.String key, java.lang.String value)
5	com.rapidminer.operator.Operator	public abstract class Operator extends java.lang.Object implements ConfigurationListener, PreviewListener, LoggingHandler, ParameterHandler	public void setListParameter(java.lang.String key ,java.util.List list)
6	com.rapidminer.operator.IOContainer	public class IOContainer extends java.lang.Object implements java.io.Serializable	public IOObject getElementAt(int index)
7	com.rapidminer.example.ExampleSet	public interface ExampleSet extends ResultObject, java.lang.Cloneable , java.lang.Iterable<Example>	ExampleTable getExampleTable()
8	com.rapidminer.example.table	public interface ExampleTable extends java.io.Serializable	DataRow getDataRow(int index)
9	com.rapidminer.example.table	public interface ExampleTable extends java.io.Serializable	Attribute getAttribute(int i)
10	com.rapidminer.example.table.DataRow	public abstract class DataRow extends java.lang.Object implements java.io.Serializable	public double get(Attribute attribute)

Exclusion processing should be considered during coding java.io.IO Exception, XML Exception, Operator Exception.

Launching RapidMiner in Java application:

```
RapidMiner.setExecutionMode(RapidMiner.ExecutionMode.COMMAND_LINE);

RapidMiner.init();
```

One of reading methods of a previously created model into an own's application is a creation of a new object of Process class:

```
Process process = newProcess(new File("C:\\Users\\Administrator\\.RapidMiner5\\repositories \\DecisionTrees\\DTree.rmp"));
```

By means of getOperator() method of Process class reading two operators from a process. A parameter of the method is an operator's name of String type, a method is returning an object of Operator type. In our case that will be as follows:

```
Operator readCSV = process.getOperator("Read CSV");

Operator resdCSV2 = process.getOperator("Read CSV (2)");
```

Setting necessary parameters for those Read CSV operators. For that using setParameter() method of Operator class. A first argument of setParameter() method defines a name of a parameter, a second one – a value of a parameter. Let's represent a first argument using corresponding statistic fields of CSVExampleSource class, for a second argument – by means of string literals.

For a start, let's define a value of CSV_FILE parameter, which is specifying a name of Excel file of csv-format for data reading.

```
readCSV.setParameter(com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_CSV_FILE, "C:\\Chapter10DataSet_Training.csv");

readCSV2.setParameter(com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_CSV_FILE, "C:\\Chapter10DataSet_Scoring.csv");
```

Further, defining parameters, such as using of a first string of a table for names of attributes, column separator and deletion of initial and following empty symbols.

```
readCSV.setParameter (com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_FIRST_ROW_AS_NAMES, "true");

readCSV.setParameter (com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_COLUMN_SEPARATORS, ",");

readCSV.setParameter (com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_TRIM_LINES, "true");
```

Setting of parameters for meta data is carried out using setListParameter() method of Operator class.

Let's create parameterTypeList collection of list<string[ ]> type [16] for an implementation of a second argument of setListParameter( ) method. In that object regular and special attributes will be defined, i.e. their names, types of data and a role according to headings of columns in the Excel table [17], [18].

```
List<String[]> parameterTypeList = newArrayList<String[]>();

parameterTypeList.add(newString[] { "0", "User_ID.true.integer.id" });

parameterTypeList.add(newString[] { "1", "Gender.true.binominal.attribute" });

parameterTypeList.add(newString[] { "2", "Age.true.integer.attribute" });

parameterTypeList.add(newString[] { "3", "Marital_Status.true.nominal.attribute" });

parameterTypeList.add(newString[] { "4", "Website_Activity.true.nominal.attribute" });

parameterTypeList.add(newString[] { "5","Browsed_Electronics_12Mo.true.binominal.attribute" });

parameterTypeList.add(newString[] { "6","Bought_Electronics_12Mo.true.binominal.attribute" });

parameterTypeList.add(newString[] { "7","Bought_Digital_Media_18Mo.true.binominal.attribute" });

parameterTypeList.add(newString[] { "8", "Bought_Digital_Books.true.binominal.attribute" });

parameterTypeList.add(newString[] { "9", "Payment_Method.true.nominal.attribute" });

parameterTypeList.add(newString[] { "10", "eReader_Adoption.true.nominal.label" });
```

In our case there are two special attributes. attribute User_ID, which has role id, eReader_Adoption attribute, which has role label for a classification of a data set. A created collection is used for a definition of meta data of ReadCSV operator:

readCSV.setListParameter(com.rapidminer.operator.nio.CSVExampleSource.PARAMETER_META_DATA, parametertyplist);

In the same manner setting parameters for the second file. That file lacks a column with name eReader_Adoption. A created model must define data of eReader_Adoption column for Chapter10DataSet_Scoring.csv file for a company's clients.

2.4.2. Obtainment of results of the model's operation

After all parameters for ReadCSV operators are set, the process can be launched:

IOContainer ioResult = process.run();

The result of the process's launch is an object of IOContainer class. RapidMiner's ApplyModel operator has two output ports: lab (label) and mod (model). It means that using getElementAt() method of IOContainer class and type conversion we can obtain an object, which realizes ExampleSet interface and an object of TreeModel class.

ExampleSet exampleSet = (ExampleSet) ioResult.getElementAt(0);

System.out.println(exampleSet);

A production of exampleSet object demonstrates that a resulting object, which realizes ExampleSet interface is supplemented with following special fields:

confidence_Late Majority = #11: confidence(Late Majority) (real/single_value)

confidence_Innovator = #12: confidence(Innovator) (real/single_value)

confidence_Early Adopter = #13: confidence(Early Adopter) (real/single_value)

confidence_Early Majority = #14: confidence(Early Majority) (real/single_value)

prediction = #10: prediction(eReader_Adoption) (nominal/single_value)/values=[Late Majority, Innovator, Early Adopter, Early Majority]

In order to correctly operate an object's data it is necessary to know a sequence of an object's creation, which is presented in figure 7.

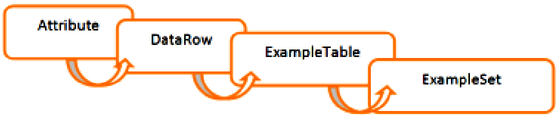


Figure 11. A Sequence Of A Creation Of An Object, Which Realizes Exampleset Interface.

By means of inversely using that sequence necessary information can be obtained, e.g. in order to obtain User_ID using values of prediction(eReader_Adoption) field, the following code can be used:

ExampleTable exsmpleTable = exampleSet.getExampleTable();

Attribute userID = exsmpleTable.getAttribute(0);

Attribute prediction = exsmpleTable.getAttribute(10);

for (int row = 0; row < exsmpleTable.size(); row++) {

data processing code for marketing communications by means exsmpleTable.getDataRow(row).get(userID) and exsmpleTable.getDataRow(row).get(prediction)

}

treeModel object is a model of Decision Tree:

com.rapidminer.operator.learner.tree.TreeModel treeModel = (TreeModel) ioResult.getElementAt(1);

That decision tree can be used for new sets of data. For a creation of new ExampleSet using getAttributes() method of exsmpleTable object. Number 5 is related with additional fields, which were presented above. The following is a fragment of the application to be used:

Attribute[] attrs = new Attribute [exsmpleTable.getAttributes().length-5];

for (int i=0; i<attrs.length; i++){

attrs[i]= exsmpleTable.getAttribute(i);}

MemoryExampleTable table = new MemoryExampleTable(attrs);

DataRowFactory factory = new DataRowFactory(DataRowFactory.TYPE_DOUBLE_ARRAY, '.');

String[] data = {"1342", "F","35", "M", "Frequent","No","No","No","No","Bank Transfer"};

```
    DataRow dataRow = factory.create(data, attrs);
     table.addDataRow(dataRow);
    ((PredictionModel)
resultSet2).apply(table.createExampleSet());
```

3. CONCLUSION

Thus, we discussed currently the most powerful complex Data Mining and statistic analysis system – Rapid Miner, which possesses a significant number of algorithms for processing and analysis including processing of large data arrays.

Certain aspects of embedding of Rapid Miner in Java code are also discussed. Methods for reading and pre-processing of data for an application of the created model are presented.

The further processing of results of the created model allows to solve communication marketing problems: buyers attraction, «wining» their attention from competitors, creation of awareness about a product among potential clients and entering new market segments. Knowing a prognosis for a target audience, a company can plan a transition of necessary information in order to achieve possible (desired) reaction.

The novelty of the presented work consists of the proposed methodology of complex application of RapidMiner as a tool for experiments, which character is analogous to establishment of laws of physics. The following stages of determinations of objective regularities in arrays of structured data are defined: application of various methods of analysis for establishment of relationships, evaluation of effectiveness of models, analysis of statistical significance of observed deviations of models' accuracy and embedding of a model in Java code for further implementation.

Due to advances in technology of writing and storage of data and increase of stored volume, data definition of useful knowledge, which is necessary for decision making, manifests practical importance of the results' application. In the discussed example use of new RapidMiner in Java applications allows to defined buying attitude of customers, which, in turn, allows to organize in applications, for example, automated messaging for promotion of marketing and pricing programs depending on customers' characteristics.

REFERENCES:

[1] Han, J., & Kamber, M. (2006). *Data Mining: Concepts and Techniques.* Morgan Kaufmann Publishers.

[2] Barsegjan, A., Kuprijanov, M., Stepanenko, V., & Holod I. (2004). *Methodologies and models for data analysis: OLAP u Data Mining.* Saint-Petersburg: BHV-Petersburg publishing house.

[3] Usman, D., & Mohamad, I. (2013). A Novel Center Point Initialization Technique for K-Means Clustering Algorithm. *Modern Applied Science, 9*(7). ISSN 1913-1844 E-ISSN 1913-1852. DOI: 10.5539/mas.v7n9p10

[4] Hajmoosaei, A., & Abdul-Kareem, S. (2008). Web data integration system: Approach and case study. *Business Information Systems, 7*, 410-423. DOI: 10.1007/978-3-540-79396-0_36

[5] Abrahams, S., Coupey, E., Zhong, E., et al. (2013, June 15). Audience targeting by B-to-B advertisement classification: A neural network approach. *Expert Systems with Applications, 8*(40), 2777-2791. DOI:10.1016/j.eswa.2012.10.068

[6] Pechenizkiy, M., Puuronen, S., & Tsymbal, A. (2005). On the Use of Information Systems Research Methods in Datamining. *Information Systems Development,* 487-499. Retrieved February 8, 2014, from http://link.springer.com/chapter/10.1007/0-387-28809-0_42. DOI 10.1007/0-387-28809-0_42

[7] Hunt, E., Marin, J., & Stone, P. (1966). *Experiments in induction*. New York: Academic Press.

[8] Quinlan, J. (1986). Induction of Decision Trees. *Machine Learning, 1*, 81-106. Kluwer Academic Publishers, Boston. DOI: 10.1007/BF00116251

[9] North, M. (2012). *Data Mining for the Masses.* Retrieved February 8, 2014, from www.rapidminer.com/wp-content/uploads/2013/10/DataMiningForTheMasses.pdf.

[10] *Rapid Miner: Operator Reference Manual.* (2014). Retrieved February 8, 2014, from www.rapidminer.com/wp-content/uploads/2014/10/RapidMiner-5-Operator-Reference.pdf.

[11] Tan, P., Steinbach, M., & Vipin, K. (2005). *Classification: Basic Concepts, Decision Trees, and Model Evaluation.* Addison-Wesley. Retrieved February 8, 2014, from www-users.cs.umn.edu/~kumar/dmbook/ch4.pdf.

[12] Berube, D. (2007). *Practical Ruby Gems.* Retrieved February 8, 2014, from http://link.springer.com/book/10.1007/978-1-4302-0193-9. DOI 10.1007/978-1-4302-0193-9

[13] Manne, S., Muddana, S., Sohail, A., & Fatima, S. (2014). Features Selection Method for Automatic Text Categorization: A Comparative Study with WEKA and Rapid Miner Tools. *ICT and Critical Infrastructure. Proceedings of the 48th Annual Convention of Computer Society of India, Vol. II, Advances in Intelligent Systems and Computing, 249*, 115-123. Retrieved February 8, 2014, from http://link.springer.com/chapter/10.1007/978-3-319-03095-1_13. DOI 10.1007/978-3-319-03095-1_13

[14] Graczyk, M., Lasota, T., & Trawiński, B. (2009). Comparative Analysis of Premises Valuation Models Using KEEL, Rapid Miner, and WEKA. Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems. *Lecture Notes in Computer Science, 5796*, 800-812. Retrieved February 8, 2014, from http://link.springer.com/chapter/10.1007/978-3-642-04441-0_70. DOI 10.1007/978-3-642-04441-0_70

[15] Chen, M., Mao, S., Zhang, Y., Leung, V. (2014). Big Data Analysis. *SpringerBriefs in Computer Science*, 51-58. Retrieved February 8, 2014, from http://link.springer.com/chapter/10.1007/978-3-319-06245-7_5. DOI 10.1007/978-3-319-06245-7_5

[16] Shen, K., Schimpf, J., Novello, S., & Singer, J. (2002). Practical Aspects of Declarative Languages. *Lecture Notes in Computer Science, 2257*, 262-279. Retrieved February 8, 2014, from http://link.springer.com/chapter/10.1007/3-540-45587-6_17. DOI 10.1007/3-540-45587-6_17

[17] Cañas, D. (2011). *Embedding Rapidminer as a library in an application.* Retrieved February 8, 2014, from www.dacamo76.com/blog/2011/07/22/embedding-rapidminer-as-a-library-in-an-application.

[18] *How to Extend Rapid Miner: Guide.* (2014). www.rapidminer.com/wp-content/uploads/2014/10/RapidMiner-extensions.pdf

[19] Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Appears in the International Join Conference on Artificial Intelligence (IJCAI).*

[20] *Rapid Miner 4.2 Class Documentation.* (2001-2008). Retrieved February 8, 2014, from http://www-ai.cs.uni-dortmund.de/LEHRE/VORLESUNGEN/MLRN / WS0809/rm-api/index.html.