

CoMoVi: a Framework for Data Transformation in Credit Behavioral Scoring Applications Using Model Driven Architecture

Rosalvo Neto *[†], Paulo Jorge Adeodato*, Ana Carolina Salgado*

*Department of Computer Science
Federal University of Pernambuco (UFPE)
Recife-PE, Brazil
{rfon2, pjla, acs}@cin.ufpe.br

Dailton Filho[†], Genival Machado[†]

[†]Department of Engineering Computer
Federal University of Sao Francisco Valley (Univasf)
Juazeiro-BA, Brazil
{dailton, genival}@univasf.edu.br

Abstract—The stage of transforming data in knowledge discovery projects is costly, in general, it takes between 50 and 80% of total project time. This step is a complex task that demands from database designers a strong interaction with experts that have a broad knowledge of the application domain, making the task prone to error. The activities of that border region require a conjugation of database, statistics and system analysis competences. These competences are not ordinarily found in the same project team, whether in academia or in professional environment. The frameworks that aim to systemize this stage have significant limitations when applied to Credit Behavioral Scoring solutions. This paper proposes CoMoVi, a framework inspired in the Model Driven Architecture to systemize this stage in Credit Behavioral Scoring solutions. CoMoVi is composed by a meta-model which maps the domain concepts and a set of transformation rules. In order to validate the proposed framework, a comparative study of performance between frameworks found in literature and the proposed framework applied to a database of a known benchmark was performed. Student’s one-tailed paired t-test showed that CoMoVi gives better performance to a Multilayer Perceptron Neural Network with a confidence level of 95%.

Keywords—*Meta-Modeling; Model Driven Architecture; Credit Behavioural Scoring; Knowledge Discovery*

I. INTRODUCTION

The stage of transforming data is costly, generally consuming between 50 and 80% of the total project time [1]. In this stage data stored in a relational database are prepared and transformed for the application of data mining techniques. Although this stage consumes more than half of a Knowledge Discovery in Databases (KDD) project time, researches in the area are focused mainly on the principal stage of the process, data mining algorithms, and proportionally few works are found in literature concerning the phase of data transformation. There are two general objectives in transformation: data must be transformed in the format that allows the data mining algorithm to be applied, and also enable the analysis necessary to evaluate the results after applying the mining technique [2]. The specific objectives of this stage are: construction of features, features selection and aggregation.

Conventional techniques of data mining like decision tree, artificial neural networks and logistic regression, applied in

KDD process require as input a table containing a row for each object of interest, and a set of columns that describe the characteristics of these objects. The existing frameworks to systemize this stage follow the propositional approach or the multidimensional data mining. The propositional approach transform the multidimensional data representation inside a simple organized relation into a denormalized table in the granularity in which the decision is to be taken, which serves as input to conventional data mining algorithms. The multidimensional data mining approach [3] proposes that knowledge is extracted from each list separately, and later combined, instead of joining the various relationships. Frameworks existing in literature show significant limitations when applied to solutions of Credit Behavioral Scoring.

This work proposes CoMoVi (the name is acronym of Conceptual Modeling Visions), a new framework inspired by Model Driven Architecture (MDA) to systematize the data transformation stage, which takes into account all peculiarities of Credit Behavioral Scoring solutions, embedding automatic knowledge in data view by automatically generating new variables increasing the discriminatory power of the data mining technique. The framework is composed by a meta-model and a set of transformation rules. To validate the proposed framework efficiency a comparative study with the main existing frameworks and CoMoVi was carried out. This comparative study verifies which one provides more discriminatory power to the Multilayer Perceptron Neural Network when applied in solutions of Credit Behavioral Scoring. The comparative study uses an experimental methodology with rigorous statistical base applied to the database from a known benchmark of an international competition organized by PKDD 1999, for a binary classification problem, in order to perform the comparison.

The remainder of this paper is organized as follows. Section 2 presents the problem definition of Credit Behavioral Scoring in relational databases. Section 3 provides a brief presentation of MDA. Section 4 presents the related work to systematize the stage of data transformation. Section 5 details the proposed framework. Section 6 shows the experimental methodology. Section 7 presents the experimental results and their interpretation. Finally, Section 8 concludes this paper and proposes future works.

This work was supported in part by the FAPESB under grant 1047/2013 and by the CAPES under grant 25001019004P6

II. PROBLEM DEFINITION

Credit Scoring and Behavioral Scoring are data mining solutions that help financial institutions to decide whether to grant credit to consumers based on the credit risk of their requests [4]. The goal of these solutions is to assign a “score” which identifies how closely the consumer is to one of two groups: “good” which will eventually meet its financial obligations, or a group of “bad”, whose application should be denied because of their high probability of failing in its commitments with the financial institution. Researches in this area have grown in recent years as a result of the recent financial crisis on a global scale.

Credit scoring is used when a new consumer makes a credit application. Only demographic information such as age, gender, income and other variables are taken into account in assigning the score. Behavioral Credit Scoring is used when a consumer who already has a history of transactions in the database of the institution is requesting credit [5]. In this case, in addition to demographic information, behavioral information is also taken into account, such as timely payment history, arrears, amount of loans, among others. The aim of the solution is to find in the database a profile that separates the good from the bad clients. The output of a solution of Credit Behavioral Scoring is interpreted as the probability of the customer to honor its debt with the institution, in order words, being a good customer.

In a recent study [6], the authors highlight the opportunities for Credit and Behavioral Scoring solutions and describe the processes involved. The first step of the process corresponds to the selection of a sample of clients, ensuring that data regarding their products and consumptions are available at a given point of observation. The period before the observation point is called the performance window. Data contained in the performance window are structured attributes that will be used as input for the solution of Behavioral Credit Scoring. Figure 1 illustrates how data are partitioned according to temporality.

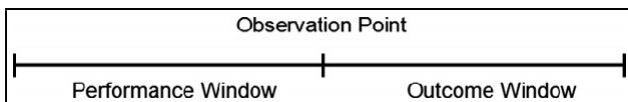


Fig. 1: Temporal Segmentation [6]

The period after the observation point is called the Outcomes Window. Data contained in Outcomes Window are structured attributes that will be used to assess the accuracy of the model, in this window the answering variable (“good” and “bad”) is constructed.

Behavioral Credit Scoring can be described as an instance of a relational classification problem in the domain of credit risk analysis. In a relational classification problem, the data available for the construction of a solution are in a database R containing a given target table, T_a and a set of background tables $T_{b1} \dots T_{bn}$. The background tables have relevant information to the decision problem, however they are not in target table. Each line belonging to T_a includes a single attribute called primary key (row identifier) and a categorical variable y , which represents the concept to be learned or “response variable”. The task of relational classification is finding a $F(x)$ function which maps each x line of the target table for Y

category. Figure 2 illustrates the problem of binary relational classification in the domain of credit risk analysis. The target table is represented by the Loan table on which the status column is the categorical variable that function $F(x)$ has to learn. This variable has two values: good, if the loan was paid on time or bad, otherwise. The background tables are represented by tables that have a relationship with the target table, which is the case in the example of Figure 2 of tables instalment and client.

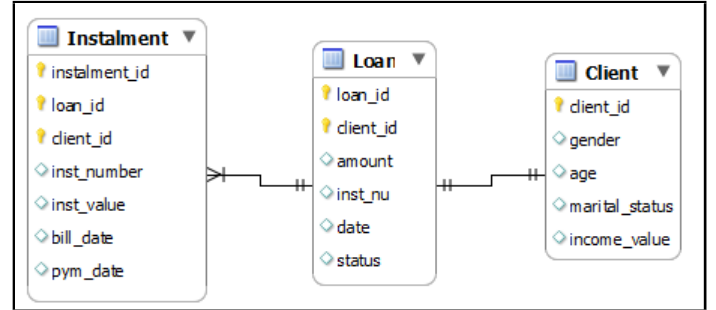


Fig. 2: Example of Relational Schema

III. BASIC CONCEPTS IN MDA TECHNOLOGY

The major step in building an application is the conceptual modeling of the business domain. In this step the mapping of the real world to the model is done by specifying all details involved, including the relationships between entities and restrictions pertaining to the business. The Model Driven Engineering (MDE) is an approach especially focused on modeling techniques. The MDE proposes that conceptual models are used both for documentation as well as for software artifacts. One of the best known initiatives in this context is the Model-Driven Architecture (MDA) proposed by the Object Management Group (OMG) [7].

MDA is a way to develop software transforming an input model in an output artifact that may be another model or source code. These models can be Platform Independent Models (PIM) and Platform Specific Models (PSM). The transformation process is performed by a processing device following transformations rules. The rules of transformations specify how to generate a target model from a source model. To transform a given model to another model, the transformation rules map the source model using the target meta-model. The MDA provides the Meta Object Framework (MOF) for specifying meta-models and the Model Transformation Language (MTL) for specifying the rules that will be used to transform an input model into an output model. Output models are often source code.

IV. RELATED WORKS

Research involving MDA and automatic code generation has been growing in recent years. Many tools and frameworks have been proposed and developed for different applications. However, according to our literature survey no proposal automates the data transformation stage in KDD projects. Among the closest researches it is possible to highlight: [8] and [9]. In [8], the authors proposed a framework based on MDA for mapping conceptual models of operational databases for Data Warehouses (DW). The framework consists in a meta-model

for specifying the conceptual model of the operational database and a set of rules for automatically generating the SQL script used in the construction of DW. In [9], the authors propose the construction of a software component based on MDA to systematize the analysis and visualization of academic information from management information systems. The authors' proposal is the semi-automatic construction of a DW in the field of higher education, thus facilitating the decision making of managers in the area. The component is based on three stages: multidimensional data modeling, data extraction and data visualization.

The two approaches found in literature applied in the stage of data transformation in a project of knowledge discovery are: propositionalization and multidimensional data mining. The approach of propositionalization transforms the multidimensional representation of data within a simple interface organized in a denormalized table in the granularity at which the decision is intended to be made. A distinct approach called multidimensional mining data [3] suggests that knowledge is extracted from each list separately, and later combined.

Relational Aggregations (RelAggs) [10] is the main framework for data transformation which follows the propositional line. In their approach the idea of aggregation, commonly used in the area of Data Warehouse is applied. Aggregation is an operation that replaces a set of values for a single value that summarizes the properties of these sets. For numerical values, simple descriptive statistics are used, such as maximum, medium and minimum value, for categorical values mode (most frequent value) can be used. RelAggs was adopted as a mechanism for transforming the Weka [1] platform, one of the main free tools for data mining. In [10] the performance of the main frameworks of transformation which follow the line of propositionalization was evaluated, and the authors concluded that RelAggs provides better performance when compared to the initial frameworks based in Inductive Logic Programming (ILP).

The Correlation-based Multiple View Validation (CbMVV) [11] is the main framework for data transformation which follows the multidimensional data mining approach. This approach is divided into three steps: firstly the relationships between tables are represented in the form of a graph, paths through this graph are combinations of visions between the target table and the background tables. In order to ensure the generation of non-cyclical paths, repeating paths are not allowed, and every path always starts from the target relation. The second step is to select the views which are relevant to the problem. For this, the authors propose an algorithm that calculates the relevance of visions by index calculation, taking into account the correlation between the attributes of vision, and also the correlation between attributes and the target concept "response variable". Views that have the lowest correlation with each other, and the highest correlation with the target, are selected. After selecting the relevant views, the algorithm enters the third and last stage in which a classifier is built for each view, and finally, a last classifier is constructed by using as input the responses from the individual classifiers of each vision. In [11], the authors demonstrated that this framework provides a higher predictive power for the data mining algorithm, when compared with ILP-based frameworks.

Based on frameworks found in literature, important charac-

teristics to the processing stage of data transformation in KDD projects applied in behavioral databases, like Credit Behavioral Scoring were identified:

- **Independence of mining technique (IMT):** This feature identifies whether the framework can be applied to any technique of data mining, since some frameworks mix the stages of data processing with the stage of data mining and, so, are specific to a certain technique.
- **Support the temporal segmentation (STS):** This feature identifies whether the framework addresses how to perform temporal segmentation during construction of behavioral variables, in other words, taking into consideration the partitioning in performance window and outcome window. This partitioning of data is essential to the success of the project using historical data as Credit Behavioral Scoring, since the use of available data on the performance window as input variables for the data mining technique makes all the project invalid.
- **Knowledge Acquisition (KA):** This feature identifies whether the framework addresses how to embed expert knowledge in the construction of variables during the stage of transforming the data to improve the discriminatory power of the data mining technique.

In Table 1, the "N" concept was attributed to frameworks that do not address the identified characteristic. And the "Y" concept for those addressing it in detail.

TABLE I: Comparison of Frameworks

Features	Frameworks		
	CbMVV	RelAggs	Based in ILP
IMT	Y	Y	N
STS	N	N	N
KA	N	N	N

V. PROPOSED FRAMEWORK

Following the MDA approach this paper proposes CoMoVi, a framework for systematizing the stage of data transformation in KDD projects on the domain of Credit Behavioral Scoring. The framework takes into consideration the temporal peculiarities of the domain and embeds knowledge in the data vision building new variables to be used as input to a data mining technique. Figure 3 shows the architecture of CoMoVi.

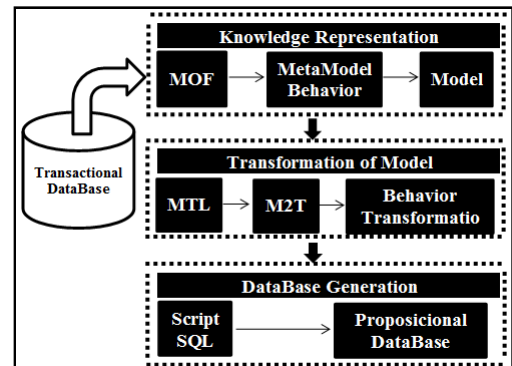


Fig. 3: Architecture of Proposed Framework

CoMoVi's architecture consists of three layers: in the first layer, called Knowledge Representation, the Behavior meta-model was defined using the MOF specification of MDA, and serves as the basis for the construction of specific models, which are adapted to the peculiarities of each database, however always following the concepts and rules defined by the meta-model. In the second layer, called the Model Transformation, the models generated by the Knowledge Representation layer are used as input to the module Transformation Behavior, which is responsible for automating the creation of SQL code for generating databases in propositional form. This module was written in MOF Model To Text (M2T) [12] which is an OMG approach to transform models into text artifacts. The third and last layer, called Database Generation, receives as input SQL codes generated by the Model Transformation layer and produces the database in propositional format that will be used as input to the data mining algorithm.

In order to embed expert knowledge and to support the temporal segmentation of data required by the Credit Behavioral Scoring the concept of behavioral Recency, Frequency and Monetary (RFM) analysis was introduced to CoMoVi meta-model [13]. The objective of the analysis is to distinguish clients based on three behavioral variables:

- **Recency (R):** Period of time since last purchase. It is the interval between the last transaction and present reference time. The lower this value is the more valuable the customer is;
- **Frequency (F):** Number of transactions in a given period until a present reference time. The higher this frequency is, the more valuable the customer is;
- **Monetary (M):** Total amount of money paid by the customer over a given period of time. The higher this value is, the more valuable the customer is.

In literature it is common to find studies using RFM variables as input for data mining techniques. In [14], the authors showed the importance of using RFM variables in building intelligent systems for e-commerce applications. The authors used the RFM variables as input to identify profiles of ecommerce users in a case study with one of the largest retail stores in Taiwan. In [15], the authors proposed a system of Customer Relationship Management (CRM) using RFM variables as input to clustering algorithms. The aim of the study was to identify niches with levels of customer loyalty to the institution.

The meta-model proposed in this paper was defined taking into account the peculiarities of the temporal segmentation and also creating new variables based on RFM analysis to embed expert knowledge. The meta-model proposed can be seen in Figure 4. The first element of the meta-model is Granularity, which represents the decision granularity of the project. This element has the "date" attribute, that represents the concept of Observation Point, used to divide the variables in a priori and a posteriori. The a priori variables represent the knowledge that happened before the Observation Point, so that they can be used as input for the data mining algorithm. The a posteriori variables represent the knowledge that happened after the Observation Point, so that they cannot be used as input, however they will be used as performance evaluation

variables such as, for instance, the response variable. The Entity element represents the background tables. This element is composed by a set of Fields elements which represent the characteristics of entities. The Relationship element is the relationship between the project granularity and another Entity. This type of relationship has one to one cardinality. The RelationshipTemp element represents the temporal relationship between the project granularity and another Entity. This type of relationship has cardinality one to many. The RelationshipTemp element has an attribute of type "date", which represents the date field of the Entity with greater cardinality in the relationship. This date field is mandatory, because it is from it that the temporal segmentation will be performed. The RelationshipTemp element has an attribute of type "fResume". This fResume attribute represents the field of the Entity from RelationshipTemp that will be used for building new RFM variables and descriptive statistics. The elements PerformanceWindow and OutcomeWindow represent the concepts of temporal segmentation defined in Section II. Each "Window" element has an attribute of type "Month". The Month attribute is an array that represents the interval of months for which RFM variables will be built, for example: if the instanced model from this meta-model uses a PerformanceWindows with two values for month, for example 6 and 12, variables will be built of type: frequency of transaction performed in the last 6 months and frequency of transaction performed in the last 12 months.

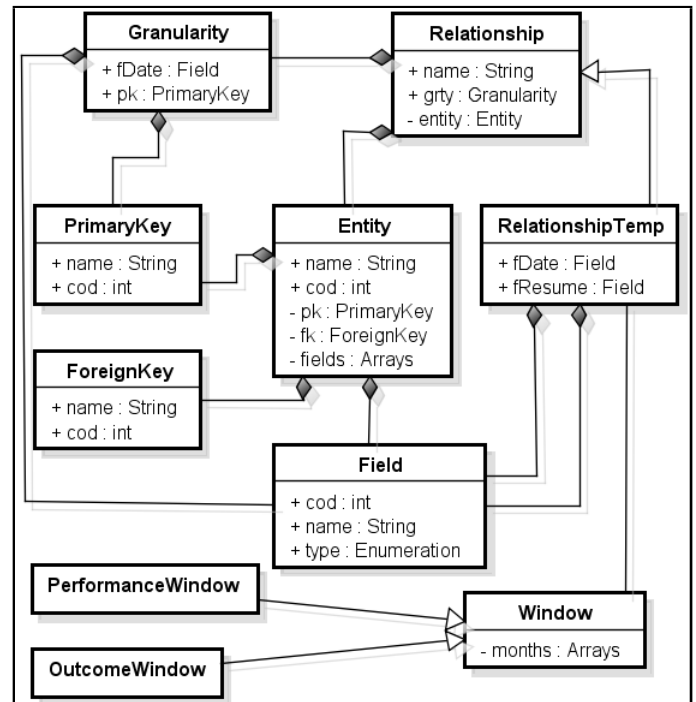


Fig. 4: Proposed Metamodel

After instantiating a model using the proposed meta-model, a set of model to text transformations is run in order to provide the SQL code that will generate the view to be used as input for a data mining algorithm. Three templates are executed. The first template RegisterData builds views with information of the backgrounds relations who have one to one relationship with the entity of granularity.

Listing 1: Code of template RegisterData

```
[template public RegisterData (rs: Relationship)]
CREATE VIEW [rs.name/] AS SELECT
[rs.grty.name/].[rs.grty.pk.name/]
[for(s: String | rs.entity.fields.name)] [rs.entity.
name/].[s/] [/for]
FROM [rs.entity.name/] , [rs.grty.name/]
WHERE [rs.grty.name/].[rs.grty.pk.name/] =
[rs.entity.name/].[rs.entity.pk.name/] [/template]
```

The second template, Behavior, constructs behavioral views from one to many relationships between background relationships and the granularity entity, the resulting View has two special attributes: APRIORI indicating whether the information may be used as input data for a data mining technique, and the DAYS field that tells how many days has that information from the Observation Point. This field will be used to calculate the Recency variable, as well as the partitioning of RFM variables in periods of months. The third and final template Windows constructs a set of new views containing the expert's knowledge by calculating RFM variables and descriptive statistics from the behavioral information generated by the template Behavior. The codes of templates are shown below.

Listing 2: Code of template Behavior

```
[template public Behavior (rst: RelationshipTemp)]
CREATE VIEW BEHAVIOR_[rst.name/] AS SELECT
[rst.grty.name/].[rst.grty.pk.name/],
([rst.fDate.name/] < [rst.grty.fDate.name/]) AS
APRIORI,
([rst.fDate.name/] - [rst.grty.fDate.name/]) AS
DAYS,
[for(s: String | rst.entity.fields.name)] [rst.
entity.name/].[s/] [/for]
FROM [rst.entity.name/],[rst.grty.name/]
WHERE [rst.grty.name/].[rst.grty.pk.name/] =
[rst.entity.name/].[rst.entity.fk.name/] [/template]
```

Listing 3: Code of template Windows

```
[template public Windows (pw: PerformanceWindows)]
[for(a: String | pw.months.first(2))]
CREATE VIEW RFM_[pw.rst.name/]_[a.trim()/] AS
SELECT [pw.rst.entity.fk.name/],
max(days) AS Recency_[a.trim()/],
count(*) AS Freq_[pw.rst.fResume.name/]_[a.trim()/],
sum([pw.rst.fResume.name/]) AS
Monetary_[a.trim()/],
max([pw.rst.fResume.name/]) AS
[pw.rst.fResume.name/]_max_[a.trim()/],
min([pw.rst.fResume.name/]) AS
[pw.rst.fResume.name/]_min_[a.trim()/],
avg([pw.rst.fResume.name/]) AS
[pw.rst.fResume.name/]_avg_[a.trim()/]
FROM BEHAVIOR_[pw.rst.name/]
WHERE APRIORI IS TRUE AND
(DAYS > 0 AND DAYS < [a.trim()/]*30 )
GROUP BY [pw.rst.entity.fk.name/] [/for] [/template]
```

VI. EXPERIMENTAL METHODOLOGY

In order to validate the effectiveness of the proposed framework, an experimental study comparing the main proposed and existing frameworks in literature was performed. The RelAggs framework was chosen as representative of the propositional approach and the CbMVV framework as representative of the

multidimensional approach. The same data mining technique was applied to the databases generated by the frameworks to verify which data transformation framework provides greater discriminatory power for the data mining technique. The technique chosen was one of the most popular in the area of artificial intelligence and very used for Credit Behavioral Scoring solutions, the Artificial Neural Networks (ANN) Multi Layer Perceptron (MLP) [16]. The study was conducted over a public database of known benchmarks used in an international competition organized by PKDD [17]. The data describe the customers of a Czech bank with their bills, credit cards, loans, transactions on their accounts and aspects of the regions where customers and bank branches are located. Figure 5 shows the relational schema of the database.

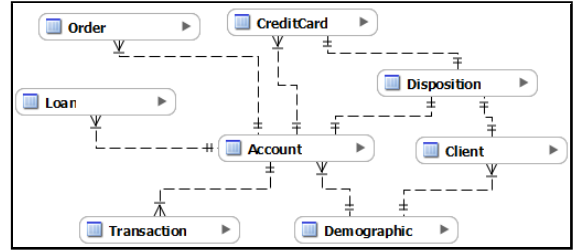


Fig. 5: Relational Schema of PKDD

The comparison was performed using the cross validation stratified k-fold framework ($k = 10$), repeated 10 times to set the confidence intervals as recommended by the authors [1]. The performance evaluation metric used was the statistical maximum value of the Kolmogorov-Smirnov's curve (KS2) using MLP as a technique for data mining. The KS2 is a non-parametric statistical method used to measure the adhesion between functions of accumulated distributions [18]. In binary classification problems the KS2 curve is the difference between two cumulative distribution functions of each class having score as the independent variable. The one-tailed paired Student's t-test was applied to verify if there is statistically significant difference between the neural networks using the three frameworks. The test setup used in this study is detailed below.

- **Null Hypothesis:** $\mu_d = \mu_1 - \mu_2$
- **Alternative Hypothesis:** $\mu_1 > \mu_2$

where

- μ_1 is the average maximum KS2 for a neural network using CoMoVi;
- μ_2 is the average maximum KS2 for a neural network using an existing framework.

VII. EXPERIMENTAL RESULTS

The simulations were performed according to the experimental setup described in Section VI for each one of the three frameworks, resulting in 10 testing sets, all statistically independent. CoMoVi provided greater predictive power for the neural network in 8 of the 10 sets of tests as shown in Figure 6, which shows the results obtained in the experiment. Table 2 shows the summary of results obtained in the one-tailed paired t-test. Since p-value is less than 0.05, we conclude

that all three frameworks provide different processing results. Specifically, data indicate that CoMoVi produces, on average, higher discriminatory power for the MLP network than RelAggs and CbMVV frameworks with a confidence level of 95%.

The results show that frameworks following the propositional approach (RelAggs and CoMoVi) outperform the CbMVV framework, which follows the approach of multidimensional data mining, in performance. The performance difference can be justified by appropriate choices of the metric for evaluating the performance and the artificial intelligence technique used in the stage of data mining, which in this work are more suitable for the domain of credit risk analysis. However, the most plausible explanation is the reduced functional capacity of the data mining algorithm caused by the input space sampling inherent to the approach of multidimensional data mining. This approach creates many local solutions with partial views of the problem, while the propositional approach builds a view with all variables using the whole functional capacity of the neural network, that is an universal approximator of functions.

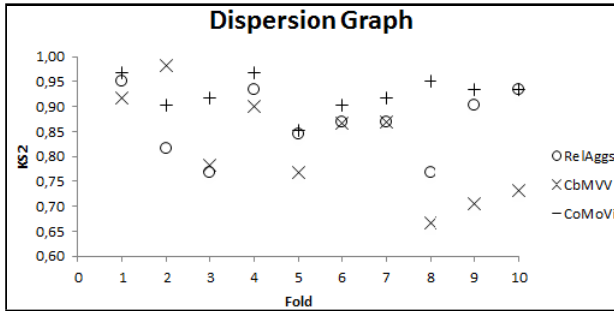


Fig. 6: Dispersion Graph

TABLE II: Summary of Results

μ_1	μ_2	Lower Limit	μ_d	Upper Limit	p -value
CoMoVi	RelAggs	0,0169	0,0554	∞	0,0133
CoMoVi	CbMVV	0,0427	0,1054	∞	0,0065

VIII. CONCLUSION

This paper presented a new framework inspired by MDA to systematize the stage of data transformation in KDD projects in the domain of Credit Behavioral Scoring. The framework is composed by a meta-model that maps key concepts of the domain and a set of transformation rules, which generate SQL code from models instantiated by the proposed meta-model. In comparison with the main existing frameworks, the experimental study showed that CoMoVi produces better performance to the technique of ANN when applied to a benchmark. The difference in performance can be explained by the construction of new variables generated by CoMoVi, based on RFM analysis within slide windows, which embeds new knowledge for the technique of data mining in the form of input variables.

Among the main contributions of the proposed framework the highlights are: 1) providing greater discriminatory power for the technique of data mining to build new variables based on RFM analysis, 2) minimizing errors in the calculation

of behavioral variables by automating temporal segmentation within the meta-model; 3) easing the use by using models for specifying data views, 4) Platform independence and technical data mining. As a future work, this study will be expanded to check the power of CoMoVi generalization in databases of different domains of credit risk analysis.

REFERENCES

- [1] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.
- [2] S. Chakrabarti, E. Cox, E. Frank, R. H. Gting, J. Han, X. Jiang, M. Kamber, S. S. Lightstone, T. P. Nadeau, R. E. Neapolitan, D. Pyle, M. Refaat, M. Schneider, T. J. Teorey, and I. H. Witten, *Data Mining: Know It All*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2008.
- [3] L. Cao, H. Zhang, Y. Zhao, D. Luo, and C. Zhang, "Combined mining: Discovering informative knowledge in complex data," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 41, no. 3, pp. 699–712, June 2011.
- [4] N.-C. Hsieh, "Hybrid mining approach in the design of credit scoring models," *Expert Syst. Appl.*, vol. 28, no. 4, pp. 655–665, 2005.
- [5] N. Sarlija, M. Bencic, and M. Zekic-Susac, "Comparison procedure of predicting the time to default in behavioural scoring," *Expert Syst. Appl.*, vol. 36, no. 5, pp. 8778–8788, 2009.
- [6] K. Kennedy, B. M. Namee, S. Delany, M. O. Sullivan, and N. Watson, "A window of opportunity: Assessing behavioural scoring," *Expert Systems with Applications*, vol. 40, no. 4, pp. 1372–1380, 2013.
- [7] O. M. G. (OMG), "Catalog of omg modeling and metadata specifications," Tech. Rep., 2008. [Online]. Available: http://www.omg.org/technology/documents/modeling_spec_catalog.htm
- [8] L. Zepeda, E. Cecena, R. Quintero, R. Zatarain, L. Vega, Z. Mora, and G. G. Clemente, "A mda tool for data warehouse," in *Proceedings of the 2010 International Conference on Computational Science and Its Applications*, ser. ICCSA '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 261–265.
- [9] J. Xie, X. Li, L. Wang, and Y. Niu, "A mda-based campus data analysis and visualization framework," in *Proceedings of the 2011 Third International Workshop on Education Technology and Computer Science - Volume 02*, ser. ETCS '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 118–121.
- [10] M.-A. Krogel and S. Wrobel, "Facets of aggregation approaches to propositionalization," in *Work-in-Progress Track at the Thirteenth International Conference on Inductive Logic Programming (ILP)*, T. Horvath and A. Yamamoto, Eds., 2003.
- [11] H. Guo and H. Viktor, "Multirelational classification: a multiple view approach," *Knowledge and Information Systems*, vol. 17, no. 3, pp. 287–312, 2008.
- [12] L. Rose, N. Matragkas, D. Kolovos, and R. Paige, "A feature model for model-to-text transformation languages," in *Proceedings of the 2012 Modeling in Software Engineering*, ser. MiSE '12. Zurich, Switzerland: IEEE Computer Society, 2012, pp. 57–63.
- [13] M. Y. Lee, A. S. Lee, and S. Y. Sohn, "Behavior scoring model for coalition loyalty programs by using summary variables of transaction data," *Expert Syst. Appl.*, vol. 40, no. 5, pp. 1564–1570, 2013.
- [14] Y.-L. Chen, M.-H. Kuo, S.-Y. Wu, and K. Tang, "Discovering recency, frequency, and monetary (rfm) sequential patterns from customers purchasing data," *Electronic Commerce Research and Applications*, vol. 8, no. 5, pp. 241–251, 2009.
- [15] C.-H. Cheng and Y.-S. Chen, "Classifying the segmentation of customer value via {RFM} model and {RS} theory," *Expert Systems with Applications*, vol. 36, no. 3, Part 1, pp. 4176–4184, 2009.
- [16] S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 2009.
- [17] P. Berka, "Guide to the financial data set," in *PKDD 2000 Discovery Challenge*, 2000, pp. 87–92.
- [18] W. Conover, *Practical nonparametric statistics*, 3rd ed. New York: Wiley, 1999.