Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 717–721

ISBN 978-83-60810-22-4

Classification of Learners Using Linear Regression

Marian Cristian Mihăescu
Software Engineering Department
University of Craiova
Craiova, Romania
Email: mihaescu@software.ucv.ro

***Abstract*—Proper classification of learners is one of the key aspects in e-Learning environments. This paper uses linear regression for modeling the quantity of accumulated knowledge in relationship with variables representing the performed activity. Within the modeling process there are used the experiences performed by students for which it is known the level of accumulated knowledge. The classification of learners is performed at concept level. The outcome is computed as a percentage representing the concept covering in knowledge**

***Keywords*—e-learning, linear regression, learner classification**

I. Introduction

THIS paper addresses the problem of classifying learners according with performed activity. Each learner is described by a set of six parameters. The parameters are of two types. One regards the quality of answers to the test questions and one regards the time in which the answers were provided. The input dataset consists of the data provided by learners who already finished the courses.

The infrastructure on which the analysis process is performed is of hierarchical nature. A discipline is considered to be an aggregate of chapters. Each chapter has an associated concept map [1]. For each concept within the concept map there is associated a set of test questions. During the usage of the e-Learning environment there are recorded necessary actions performed by learners such that a set of parameters may be obtained.

Once a learner obtains a final result for a discipline his experience may be used for building the linear regression classifier. When there are enough learners with complete data regarding their activity we may start using the classifier on new learners.

The classifier may be used for recommendations purposes. Once a student is classified there may be determined a class with better knowledge coverage and thus there may be determined the activities that need more attention such that the student "jumps" into the destination class. The classifier may be also used to predict the knowledge level of a learner at concept level or at discipline level based on learner's activity and on current classification model.

The presented analysis process enables an e-Learning system to be give advice to learners regarding activities that need to be performed or an estimation of the knowledge level at a certain moment in time.

Activity data has been obtained from Tesys [2] e-Learning platform. The data is offline processed by Weka [3] data mining software which is a is a collection of machine learning algorithms for data mining tasks.

The second section presents the state of the art regarding presented issues. The third section presents the infrastructure and methods used in analysis process. The first subsection presents the e-Learning infrastructure that has been used for obtaining activity data. The second subsection presents Concept Maps. The third subsection presents the linear regression technique. Section four presents the analysis process. Section five presents a sample experiment with real data. Finally, conclusions and future works are presented.

II. Related Work

Linear regression is a statistical approach for modeling the relationship between a scalar variable y and one or more variables denoted X [4]. There are many domains in which linear regression is used. Trend line, epidemiology and finance are some of the domains in which linear regression is used.

A trend line represents a trend, the long-term movement in time series data after other components have been accounted for. It tells whether a particular data set (say GDP, oil prices or stock prices) have increased or decreased over the period of time. The term "Trend Analysis Report" is found in many domains an provide important knowledge regarding the analyzed data.

In epidemiology there are studies trying to relate tobacco smoking to mortality. There were performed regression analysis studies in order to reduce spurious correlations when analyzing observational data. The simplest approach builds a regression model in which cigarette smoking is the independent variable of interest, and the dependent variable is lifespan measured in years. Of course, other dependent variables such as socio-economic status may be added to show that the effect of smoking on lifespan is not due to any effect of education or income.

Linear regression is not very much used in e-Learning domain. Educational Data Mining [5] is an emerging discipline, concerned with developing methods for exploring the unique types of data that come from educational settings, and using those methods to better understand students, and the settings which they learn in.

There were performed many studies that used statistical and machine learning algorithms on data provided by e-Learning environments. Some of used algorithms are association rules [6], clustering [7], Bayesian networks [8].

Statistical and machine learning algorithms are used to solve important issues of on-line learning environments. Some of the most addressed issues are simulation and modeling learner's interaction [9, 10], prediction of future performance of learners [11], clustering and classification of learners [7].

III. Employed Infrastructure and Methods

A. The e-Learning Environment

E-Learning systems are mainly concerned with delivery and management of content (e.g., courses, quizzes, exams, etc.). Since we are speaking about a web platform the client is represented by the browser, more exactly by the learner that performs the actions.

Defining the e-Learning infrastructure or the presented purpose represents the first and the most important step. In this phase, all the possible actions that may be performed by a learner need to be presented. There are also identified the resources that are delivered by the e-Learning system. Finally, there are identified the highly complex business logic components that are used when actions are performed by learners.

Each implemented action needs to have an assigned weight. In the prototyping phase, the assignment of weights is performed manually according with a specific setup. This assumes that we have an e-Learning system that is already set up. The main characteristics regard the number of learners, the number of disciplines, the number of chapters per discipline, the number of test/exam questions per chapter and the dimension of the document that is assigned to a chapter. The data that is obtained from analyzing a certain setup will represent the input data for the simulation procedure.

Another type of activities regarding learners are represented by the communication that take place among parties. Each sending or reading of a message is assigned a computed average weight.

A sample e-Learning setup infrastructure may consist of 500 students, 5 disciplines, 5 to 10 chapters per discipline, 10 to 20 test/exam questions.

For this infrastructure here may be established a list of costs for all needed actions that may be performed by learners. The weight assigned to an action takes into consideration the complexity of the action and the dimension of the data that is obtained as response after the query is sent.

For obtaining reasonable weight, a pre-assessment procedure is performed. The simulation tool performs this procedure from a computer that resides in the same network as the server such that response times are minimal. Each request that is composed and issued to the e-Learning platform is measured in terms of time and space complexity. A scaling factor will assign each action a certain weight such that the scenarios that will be created when real time testing starts will have a sound basis.

The pre-assessment procedure firstly loads all the data regarding the analyzed e-Learning platform. This means the data about all managed resources (e.g. disciplines, chapters, quizzes, etc.) are loaded such that the simulation tool may build valid requests for the e-Learning environment.

B. Concept Maps

Concept mapping may be used as a tool for understanding, collaborating, validating, and integrating curriculum content that is designed to develop specific competencies. Concept mapping, a tool originally developed to facilitate student learning by organizing key and supporting concepts into visual frameworks, can also facilitate communication among faculty and administrators about curricular structures, complex cognitive frameworks, and competency-based learning outcomes. To validate the relationships among the competencies articulated by specialized accrediting agencies, certification boards, and professional associations, faculty may find the concept mapping tool beneficial in illustrating relationships among, approaches to, and compliance with competencies [12].

Recent decades have seen an increasing awareness that the adoption of refined procedures of evaluation contributes to the enhancement of the teaching/learning process. In the past, the teacher's evaluation of the pupil was expressed in the form of a final mark given on the basis of a scale of values determined both by the culture of the institution and by the subjective opinion of the examiner. This practice was rationalised by the idea that the principal function of school was selection - i.e. only the most fully equipped (outstanding) pupils were worthy of continuing their studies and going on to occupy the most important positions in society.

According to this approach, the responsibility for failure at school was to be attributed exclusively to the innate (and, therefore, unalterable) intellectual capacities of the pupil. The learning/ teaching process was, then, looked upon in a simplistic, linear way: the teacher transmits (and is the repository of) knowledge, while the learner is required to comply with the teacher and store the ideas being imparted [13].

Usage of concept maps may be very useful for students when starting to learn about a subject. The concept map may bring valuable general overlook of the subject for the whole period of study. It may be advisable that at the very first meeting of students with the subject to include a concept map of the subject.

C. Linear Regression and Weka

Linear regression is the most popular regression model [14]. The goal of this model, is to predict the response to n data points $(x_1, y_1),(x_2, y_2), ...,(x_n,y_n)$ by a regression model given by

$$y = a_0 + a_1 x$$

where a_0 and a_1 are the constants of the regression model.

A measure of goodness of fit, that is, how well $a_0 + a_1 x$ predicts the response variable y is the magnitude of the residual ε_i at each of the n data points.

$$E_i = y_i - (a_0 + a_1 x_i)$$

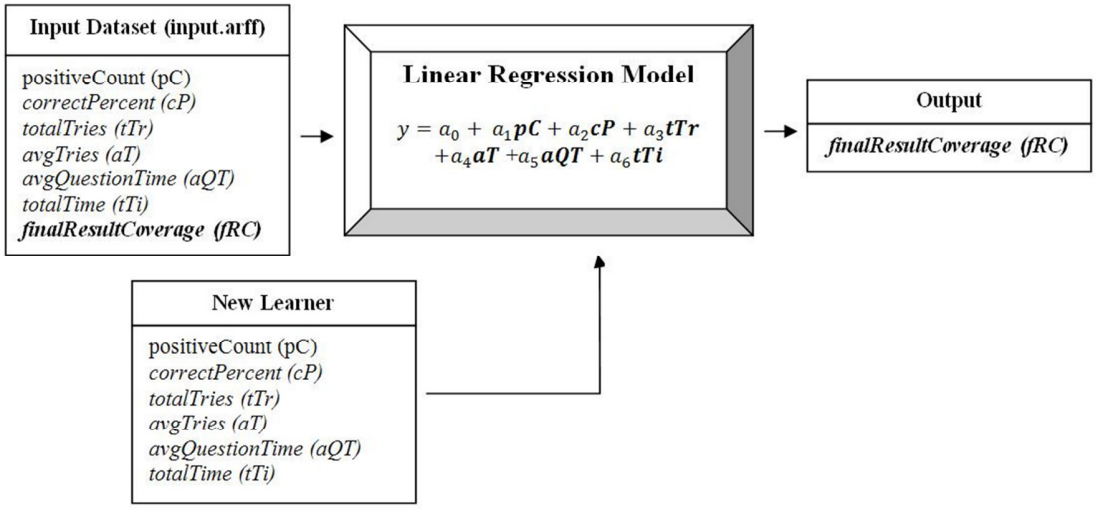

Figure 1. The analysis process

Ideally, if all the residuals ε_i are zero, one may have found an equation in which all the points lie on the model. Thus, minimization of the residual is an objective of obtaining regression coefficients.

The most popular method to minimize the residual is the least squares methods, where the estimates of the constants of the models are chosen such that the sum of the squared residuals is minimized, that is minimize $\sum_{i=1}^{n} E_i^2$.

Linear regression may be performed on models that have one input variable or multiple input variables. For the multiple input variables the equation has the form:

$$y = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n$$

Linear regression finds the parameter values (for the weights $a_1, \ldots, a_n$ and constant a_0 that minimize the sum of the squares of the differences between the actual and predicted y values.

Weka [15] is a collection of machine learning algorithms. It includes schemes for classification, numeric prediction, meta-schemes and clustering. Linear regression is one of the implemented numeric prediction schemes. Weka uses *arff* file format which require declaration of @RELATION (associates a name with the dataset), @ATTRIBUTE (specifies the name and attribute of an attribute) and @DATA (denotes the start of data segment).

The preprocessing phase in Weka is represented by the necessary actions that load the data. Once the data is loaded there may be performed a linear regression on the dataset. In order to perform this analysis the *LinearRegression* must be chosen. It may be found under *Classify* tab right at *functions* leaf. Finally, the last step to creating our model is to choose the dependent variable (the column we are looking to predict).

IV. Analysis Process

The analysis process has four phases. Firstly, the overall procedure is described. This means that the goal of the process is clearly defined. Than, there are defined the parameters that characterize each instance that build up the dataset. The third step is represented by the effective running of the analysis procedure and obtaining results. Finally, the results are interpreted.

The goal of the analysis process is to predict the knowledge coverage at concept level. A discipline is supposed to be composed of chapters and each chapter has an associated concept map. Each concept has an associated set of quiz questions. We suppose that we have a dataset consisting of past experience of learners regarding quizzes answered related to analyzed concept. For these learners there is known the final result coverage of the concept. Having this data modeled our goal is to estimate the concept coverage using the activity performed by the analyzed learner.

The parameters that characterize each instance are:

positiveCount – represents the number of correctly answered questions;

correctPercent – represents the percentage of correctly answered questions from the total number of questions;

totalTries – represents the total number of tries (answered questions);

avgTries – represents the medium number of tries per question;

avgQuestionTime – represents, on average, how long (in minutes) it takes for a student to answer a question;

totalTime – represents the total time spent on testing;

finalResultCoverage – represents the final coverage of the concept. This value is obtained from the final examination data and represents the dependent variable. The value of this variable is known for all learners that participate in building the model. The value of this parameter will be predicted for new learners that provide values only for first six variables.

Figure 1 presents the analysis process. It may be observed that the input dataset is represented by *input.arff* file. In this file resides the data regarding the activity

performed be learners that is used for building the linear regression model. When the input data is fed to the linear regression model builder the final result coverage variable is set as dependent variable. Once the model is created, it may be used to predict the value of the dependent variable provided that values for all other parameters are given. This constitutes the input provided by the learner whose final result coverage needs to be predicted.

This setup uses only normalized and continuous type parameters. That is why the linear regression is chosen from the area of supervised learning algorithms. An important aspect regards the fact that the output variable *finalResultCoverage* is not computed as a formula that takes into consideration the other parameters. The output or predicted variable is obtained from real life examples and thus there is no clear (mathematical) prior dependency between this variable and the rest of variables. This approach makes the experiment to have real consistency regarding the learning process.

V. SAMPLE EXPERIMENT

The goal of the experiment has already been presented in the analysis process section. The structure of the data is presented in the first section of the *input.arff* file where the attribute names and types are presented. All attributes are of numeric type. This is a constraint imposed by the *LinearRegression* procedure implemented by Weka.

```
@RELATION activity

@ATTRIBUTE positivCount NUMERIC
@ATTRIBUTE correctPercent NUMERIC
@ATTRIBUTE totalTries NUMERIC
@ATTRIBUTE avgTries NUMERIC
@ATTRIBUTE avgQuestionTime NUMERIC
@ATTRIBUTE totalTime NUMERIC
@ATTRIBUTE finalResultCoverage NUMERIC
```

The @data section provides the actual data values. The actual number of learners that are used for building the model is 40. Here is a sample of the dataset corresponding to four learners.

```
@DATA
120,90,133,12.3,45.6,100,71
110,65,71.5,10.3,25.6,180,52
331,27,67,15.8,31.6,56,43
63,92,80,22.6,15.6,36,34
93,31,126,41.6,75.6,60,10
87,62,12,33.8,41.3,41,76
…
```

The first line from @DATA section represents a learner who answered correctly to 120 questions, with a correct percentage f answered questions of 90%, a total number of tryes of 130 and an average of 12.3 number of tries per question. Regarding the time values, the student has an average of 45.6 seconds per question with a total time spent on line of 100 hours. This time represents the whole time spent on-line within the e-Learning environment. It consists of time spent for testing and for study. At the final examination the discussed concept had a coverage of 71%.

The file with data is set as input data in the preprocessing phase. Than, the *LinearRegression* algorithm is chosen from the functions implemented under *Classify* tab. The *finalResultCoverage* is set as dependent variable. Running in Weka is performed by clicking the *Start* button. Finally, the classifier output is obtained. The output of this analysis process is presented below and represents the linear regression model.

```
=== Run information ===
Scheme:       weka.classifiers.functions.LinearRegression
Relation:     activity
Instances:    40
Attributes:   7
              positivCount
              correctPercent
              totalTries
              avgTries
              avgQuestionTime
              totalTime
              finalResultCoverage
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===
Linear Regression Model
finalResultCoverage =
     0.2556 * correctPercent +
    -0.401  * totalTries +
     0.267  * totalTime +
    40.1076

Time taken to build model: 0.03 seconds
=== Cross-validation ===
=== Summary ===
Correlation coefficient                  0.2336
Mean absolute error                     22.6838
Root mean squared error                 28.1816
Relative absolute error                103.337  %
Root relative squared error            109.6054 %
Total Number of Instances               40
```

The above result presents the obtained coefficients of the variables representing the regression output. The interpretation of the pattern regarding the obtained model is the of great importance. Firstly, *positivCount*, *avgTries* and a*vgQuestionTime* parameters do not matter. WEKA uses only columns that statistically contribute to the accuracy of the model. It will throw out and ignore columns that don't help in creating a good model. So this regression model is telling us that the number of correctly answered questions doesn't affect the final coverage of the concept.

The total time spent and the correct percentage of correctly answered questions are the parameters that matter.

High number of tries reduce the final coverage of the concept. WEKA is telling us that if learner has a large number of tries his final coverage will be lower. This can be seen by the negative coefficient in front of the *totalTries*

parameter. The model is telling us that every additional try of answering questions reduces the final coverage of the concept by 0.4 percent.

Now, that we have a model we can use it. Let us suppose we have a learner that just had some time spent reading and answering test questions regarding a concept. At this point he may want to know his knowledge coverage of this poncept. All we have to do is to feed his data into the model and the the concept coverage will be determined.

The values for the classified learner are: 70 in correctPercent, 60 in totalTries and 50 in totalTime.

Applying the formula we obtain:

70*0.2556 + 60*(-0.401) + 50*0.267 + 40.1076 = **47.2896**

The obtained result needs interpretation. It means that the discussed concept is covered 47.2896 percent by the learner. This is a predicted value obtained by the linear regression model taking into account the previous experiences offered by 40 learners and the current activity performed by the learner for which the prediction is performed.

VI. Conclusions and Future Work

This paper strives to use a simple data mining technique to obtain knowledge regarding a learner. The goal is to create a model that may be used to predict the knowledge coverage for a learner at concept level. The main outcome of this procedure is that is produces important information for the learner. The created procedure may be adapted for virtually any e-Learning environment with the proper adjustment of the parameters.

Performing such an analysis process needs several things. Firstly, an e-Learning environment is needed. There is also needed data regarding learner's performed activities in a structured manner. This dataset represents the input data for the modeling technique.

A modeling technique is needed. In this paper it is used Linear Regression Modeling implemented by Weka. The outcome of the process is a set of parameters (the coefficients of the linear equation) that may be used to compute a dependent variable.

Once the model is obtained it may be used to compute the value of the dependent variable for a learner that provides values all other parameters. An interpretation of the parameters is needed.

Future works may improve and generalize this procedure. The coverage level may be taken into consideration at chapter or even discipline level.

Another important approach may take into consideration shifting towards discrete values for output variable and even for the other parameters. This may open the window for using other supervised learning algorithms from the area of classifiers.

Obtained results may be used for building a recommender system that runs along the e-Learning system.

VII. Acknowledgment

This work was supported by the strategic grant POSDRU/89/1.5/S/61968, Project ID61968 (2009), co-financed by the European Social Fund within the Sectorial Operational Program Human Resources Development 2007 – 2013.

References

[1] J. D. Novak and A. J. Cañas, "The Theory Underlying Concept Maps and How to Construct and Use Them", *Technical Report IHMC CmapTools,* 2006.

[2] D. D. Burdescu and M.C. Mihăescu, "Tesys: e-Learning Application Built on a Web Platform", *Proceedings of International Joint Conference on e-Business and Telecommunications,* Setubal, Portugal , pp. 315-318, 2006.

[3] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update", SIGKDD Explorations, Volume 11, Issue 1, 2009.

[4] A. K. Kaw and E. E. Kalu, *Numerical Methods with Applications,* http://www.autarkaw.com, second edition, 2010.

[5] C. Romero and S. Ventura, "Educational Data Mining: A Survey from 1995 to 2005", *Expert Systems with Applications,* 33(1), pp. 135-146, 2007.

[6] E. García, C. Romero, S. Ventura, T. Calders, "Drawbacks and solutions of applying association rule mining in learning management systems", *Proceedings of the International Workshop on Applying Data Mining in e-Learning,* pp. 1-10, 2008.

[7] R. Nugent, N. Dean, E. Ayers, "Skill Set Profile Clustering: The Empty K-Means Algorithm with Automatic Specification of Starting Cluster Centers", *Proceedings of The 3rd International Conference on Educational Data Mining,* pp. 151-160, 2010.

[8] N. Khodeir, N. M. Wanas, N. M. Darwish, N. Hegazy, "Inferring the Differential Student Model in a Probabilistic Domain Using Abduction inference in Bayesian networks", *The 3rd International Conference on Educational Data Mining,* pp. 299-300, 2010.

[9] M. Mavrikis, "Data-driven modelling of students' interactions in an ILE", *The 1st International Conference on Educational Data Mining,* pp. 87-96, 2008.

[10] H. Jeong and G. Biswas, "Mining Student Behavior Models in Learning-by-Teaching Environments", *First International Conference on Educational Data Mining,* Montreal, pp. 127-136, 2008.

[11] H. F. Yu et. al., Feature Engineering and Classifier Ensemble for KDD Cup 2010, *JMLR Workshop and Conference Proceedings,* Invited Paper of KDD Cup 2010 Winner, 2010.

[12] E. McDaniel, B. Roth, M. Miller, "Concept Mapping as a Tool for Curriculum Design", *The Journal of Issues in Informing Science and Information Technology,* Volume 2, pp. 505-313, 2005.

[13] L. Vecchia, M. Pedroni, "Concept Maps as a Learning Assessment Tool", The Journal of Issues in Informing Science and Information Technology, Volume 4.,pp. 307-312, 2007.

[14] A. Kaw, E. Kalu, *Numerical Methods with Applications,* 2010.

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations,* Volume 11, Issue 1, 2009.