Predicting Student Performance with Data from an Interactive Learning System

Ana Gonçalves
argonalves@ua.pt
Ana Tomé
ana@ua.pt
Luís Descalço
luisd@ua.pt
1 Center for Research & Development in Mathematics and Applications
University of Aveiro
Aveiro, PT
2 Institute of Electronics and Informatics Engineering of Aveiro
University of Aveiro
Aveiro, PT

Abstract

Nowadays Interactive Learning Systems have been developed to provide students with new forms of practicing concepts. In this work we propose to predict if the student fails or succeeds in the introductory mathematics course based on the information collected by an interactive learning platform. The predicting models are based on binary support vector machines (SVM). As some of the collected data sets are unbalanced the study was conducted with suitable strategies to train this binary classifier.

1 SIACUA application and collected data

SIACUA - Sistema Integrativo de Aprendizagem por Computador, Universidade de Aveiro - is a web application designed to support autonomous study. For each subject there is defined a concept map with questions associated to each concept. The system is supplied with parametrized questions from PmatE (pmate.ua.pt) and MEGUA (cms.ua.pt/megua) projects. It implements a user model based on Bayesian networks. The student chooses a subject and the system presents a sequence of related questions. After every student response the system provides a feedback. The student gets to know if the answer is correct, and in case it is not, he also gets the solution. In case of the MEGUA set the student also receives a detailed guidance of how to solve the problem. The student may also choose only to see the solution without answering the question. For a detailed SIACUA description please refer to Fonseca master thesis [3]. During a student session with SIACUA, the following information is stored: the student identification, the ID of each question and information related with student’s question/answers. The latter comprises: the time-stamp for each question, the time elapsed between question presentation and student answer; elapsed time for solution visualization; the type of student answer (0-incorrect;1-correct; 2-solution visualization). This work proposes a machine learning based methodology to interpret the interaction of students with SIACUA. The goal of the work is to see if the behavior of the students with the system is related with the success of the student in the course. Therefore it is proposed a feature extraction block which describes the student in terms of platform’s use. Afterward a classifier is used to predict if the student approves or fails. Naturally this scenario leads to unbalanced data sets, as it is expected that the majority of students with SIACUA. The goal of the work is to see if the behavior of the students with the system is related with the success of the student in the course. Therefore it is proposed a feature extraction block which describes the student in terms of platform’s use. Afterward a classifier is used to predict if the student approves or fails. Naturally this scenario leads to unbalanced data sets, as it is expected that the majority of students that were submitted to final evaluation succeed. Then, the SVM classifiers will be studied, however using methodologies to train with unbalanced data sets.

2 Data set

Only the students that were submitted to final evaluation were considered. In Table 1 it is presented the number of students per group (that were approved or that failed) and discipline.

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Approved</th>
<th>Failed</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cálculo 2</td>
<td>187</td>
<td>140</td>
<td>327</td>
</tr>
<tr>
<td>Cálculo 3</td>
<td>250</td>
<td>62</td>
<td>312</td>
</tr>
</tbody>
</table>

Table 1: Students by discipline and group.

Note that, for Cálculo 2, the cardinality of Approved and Failed sets are similar, while for Cálculo 3 the Approved subset is about 80% of students.

2.1 Feature extraction

Each student of the data set is represented as a vector \(x\) then forming data point in a space of dimension \(M\). The vector has dimension \(M = 42\) and each vector entry corresponds to one characteristic computed with the data collected by the platform. Examples of characteristics are the number of days the student interacts with the platform, the number of answers, the average time to answer questions and so on. Analysing the collected data by day perspective, it became possible to have, for instance, the number of answers a student gives during assessment and regular days.

3 Classification with SVM

Support Vector Machine (SVM) is a reliable two class classifier based on the decision equation

\[
g(z) = \sum_i \alpha_i y_i K(x_i, z) + b \Rightarrow \begin{cases} 
  g(z) > 0, & y = 1 \\
  g(z) < 0, & y = -1
\end{cases}.
\]  

(1)

The parameters determined by classifier training are: \(\alpha_i\), the support vectors \(x_i\) and labels \(y_i\) and \(b\). The support vectors are the elements of the training set that have associated nonzero \(\alpha_i\). \(K(x_i, z)\) is the kernel function. Two possible kernels were considered: the linear kernel

\[
K(x, z) = x^T z
\]

(2)

which determines that the separation plane is a hyperplane; and the RBF kernel

\[
K(x, z) = \Phi^T(x) \Phi(z) = \exp\left(\frac{-\|x-z\|^2}{2\gamma}\right),
\]

(3)

in which case the separation surface is not linear.

4 Linear SVM and unbalanced data

The hyperplane determined by using unbalanced training data sets is in most of the cases more close to the majority class [1, 5, 7]. Different approaches have been proposed to deal with unbalanced data sets. In the following subsections some are described: Different Error Costs (DEC), Synthethic Over-sampling + DEC and \(z\)-SVM.

4.1 DEC

The Different Error Costs (DEC) proposed by Veropoulos et al. [7] assigns different error costs to the misclassification of objects from each class. The linear SVM with smooth margin minimization problem is given by

\[
\min \left( \frac{1}{2} w^T w + C \sum_i \xi_i \right) \text{ with } y_i (w^T x_i + b) \geq 1 - \xi_i, \xi_i \geq 0.
\]

(4)

where \(\xi\) are slack variables for the misclassified elements. The \(C\) parameter controls the tradeoff between the number of misclassified objects and the margin width and it is an user defined parameter.

Let \(C^+\) and \(C^-\) be the error costs for misclassification of negative and positive class objects, respectively. The expression for SVM with smooth margin can be re-written as

\[
\min \left( \frac{1}{2} w^T w + C^+ \sum_i \xi_i + C^- \sum_i \xi_i + b \right).
\]

(5)
subject to the same conditions as in (4). After training using the dual form of the defined optimization problem the Lagrangians and support vectors are available. Note that in the case of linear kernel it is possible to compute the normal to the hyperplane \( w \) that characterizes the decision surface (e.g. the hyperplane can be computed using the support vectors and Lagrangian values [4]). While in the the case of RBF, the support vectors and Lagrangian values have to be stored to the test phase.

### 4.2 Synthetic Over-sampling + DEC

The experimental results presented by Akbani et al. [1] suggest that the application of synthetic over-sampling techniques before training DEC have better performance. Two techniques have been studied: Synthetic Minority Over-sampling Technique proposed by Chawla et al. [2], known as SMOTE and SMOTE SVM proposed by Nguyen et al. [6].

In SMOTE, the new instances are generated over the line segments that connect the nearest neighbors of the minority class. In SMOTE SVM, first an SVM classifier is trained on the original data set. Then, new instances are generated based on the minority class support vectors of the resulting model. This way, the new instances are generated along the decision boundary. Figure 1 illustrates the results obtained by both techniques.

![Figure 1: Synthetic over-sampling by SMOTE and SMOTE SVM.](image)

Observing the middle graphic in figure 1 it can be seen that in the data set resulting of over-sampling with SMOTE, the minority class is denser having its instances uniformly distributed over the original instances area. While the data set resulting of over-sampling with SMOTE SVM (right graphic) has the new minority class instances concentrated along the decision boundary, where they are critical for estimating the optimal decision boundary.

### 4.3 z-SVM

Imam et al. [5] presented z-SVM. With this strategy, first an SVM classifier is trained. Then, the decision boundary is adjusted in order to correct the bias towards the majority class. Rewriting equation 1 by separating the support vectors from each class and then, magnifying the \( \alpha \) values associated with the minority class support vectors by a small positive constant \( z \), the z-SVM classifier is obtained:

\[
\begin{equation}
\begin{aligned}
g(z) &= \sum_{i:y_i=0} \alpha_i y_i K(x_i, z) + \sum_{i:y_i=1} \alpha_i y_i K(x_i, z) + b \Rightarrow \\
&= \begin{cases} 
\quad g(z) > 0, & y = 1 \\
\quad g(z) < 0, & y = -1 
\end{cases}
\end{aligned}
\end{equation}
\]

(6)

The \( z \) value is experimentally determined and is the one that maximizes the geometric mean \( g \). Note that the minority class is considered as the positive one.

### 5 Results and Discussion

The simulations used the python software package scikit-learn\(^1\) and the python library imbalanced-learn\(^2\) for the synthetic over-sampling. Three data normalizations were considered: no normalization (none), min-max and z-score normalizations.

#### 5.1 Evaluation

There are several performance measures. The usual one is accuracy, the proportion of objects correctly classified. Considering only the negative class object, the accuracy among these objects is the proportion of negative class objects correctly classified and designated specificity \( s \). The proportion of positive class objects correctly classified is the sensitivity or recall \( r \). For unbalanced data sets it is usual to use the geometric mean between specificity and recall as a performance measure

\[
g = \sqrt{r \times s}.
\]

#### 5.2 Performance

Table 2 presents a summary of the results obtained for the Cálculo 3 data set.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>none</th>
<th>min-max</th>
<th>z-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM linear</td>
<td>0.785</td>
<td>0.304</td>
<td>0.801</td>
</tr>
<tr>
<td>SVM RBF</td>
<td>0.801</td>
<td>0</td>
<td>0.801</td>
</tr>
<tr>
<td>DEC</td>
<td>0.664</td>
<td>0.656</td>
<td>0.670</td>
</tr>
<tr>
<td>SMOTE+DEC</td>
<td>0.603</td>
<td>0.644</td>
<td>0.628</td>
</tr>
<tr>
<td>S.SVM+DEC</td>
<td>0.641</td>
<td>0.643</td>
<td>0.715</td>
</tr>
<tr>
<td>z-SVM</td>
<td>0.385</td>
<td>0.422</td>
<td>0.401</td>
</tr>
</tbody>
</table>

Table 2: Accuracy and geometric mean \( g \) for unbalanced data.

It seems to be an advantage in the use of classifiers adapted for unbalanced data sets, with the exception of z-SVM. Recall that z-SVM’s starting point is the decision boundary obtained by applying linear SVM. And, as shown by results in table 2, this model is much biased towards the majority class. So, it is not a surprise that the decision boundary adjustment provided by the z-SVM approach does not offer so good results.

DEC and SMOTE SVM + DEC classifiers are the ones with highest accuracy and geometric mean. So, for the studied data sets, the approaches for unbalanced data based on different error costs for misclassified objects present better results. For original data and min-max normalized data, the DEC classifier is the one with highest accuracy and geometric mean. For z-score normalized data, SMOTE SVM + DEC approach presents the best results. These results also suggest that is not always worth it the extra processing for data generation, before applying DEC classifier.

Although the Cálculo 2 data set is quite balanced, the approaches for unbalanced data sets were also tried. As expected there is no advantage in using these techniques on this set. The accuracy and \( g \) values are close to 0.72 for linear SVM and slightly better for RBF SVM. With unbalanced strategies, including the z-SVM, those values decrease (less than 3 %). These results are according to what is reported in other works [1, 5].

### 6 Conclusion

Despite the application of strategies to deal with unbalanced data sets, the obtained results are fragile. It is believed that the amount of collected data is not enough to fully cover the diversity of the student behavior. So, concerning the work primary goal, it is not yet possible to predict about a student success. Hopefully, it becomes possible to determine a reliable prediction model as more data is collected.

The preliminaries results presented in this work show that the features are relevant for decision making and strategies for unbalanced data sets improve the classification of linear SVM.

### References


\(^1\)http://scikit-learn.org

\(^2\)http://github.com/scikit-learn-contrib/imbalanced-learn
variáveis. Master’s thesis, Departamento de Matemática, Universi-
dade de Aveiro, 2014.


improved classification of imbalanced data. In Proceedings of the
19th Australian Joint Conference on Artificial Intelligence: Advances


[7] K. Veropoulos, C. Campbell, and N. Cristianini. Controlling the sen-
sitivity of support vector machines. In Proceedings of the Interna-