Learning Analytics & Educational Data Mining

Sangho Suh

Computer Science, Korea University
Seoul, Republic of Korea

sh31659@gmail.com

Wednesday, June 18, 2016

Abstract

This paper followed CRISP-DM[1] development cycle for building classification models for two different datasets: 'student performance' dataset consisting of 649 instances and 33 attributes; 'Turkiye Student Evaluation' dataset consisting of 5,820 instances and 33 attributes. To avoid confusion, this paper is organized into two parts (Part A, B) where analysis on each dataset is presented separately. Note that the general flow of the paper will abide by the steps shown in the following Table of Contents.

Table of Contents

[1] http://www.sv-europe.com/crisp-dm-methodology/

Introduction

The overall goal of this project is to provide detailed analysis of chosen datasets while building classification models.

For this project, we use the Weka (Waikato Environment for Knowledge Analysis)[2] data mining toolkit. This toolkit provides a library of algorithms and models for classifying and analyzing data.

To ensure accuracy, all development and testing of models will follow the CRISP_DM process.

- Exploration of the problem
- Exploration of the data and its information (meta)
- Data preparation
- Model development
- Evaluating outcomes

Part A. 'Student Performance Data Set'

1.0 Data Exploration of 'Student Performance Data Set'

A-1. Data Set Information:

"This data approach student achievement in secondary education of two Portuguese schools. The data attributes include student grades, demographic, social and school related features) and it was collected by using school reports and questionnaires. Two datasets are provided regarding the performance in two distinct subjects: Mathematics (mat) and Portuguese language (por). In [Cortez and Silva, 2008], the two datasets were modeled under binary/five-level classification and regression tasks. Important note: the target attribute **G3 has a strong correlation with attributes G2 and G1**. *This occurs because G3 is the final year grade (issued at the 3rd period), while G1 and G2 correspond to the 1st and 2nd period grades*. It is more difficult to predict G3 without G2 and G1, but such prediction is much more useful (see paper source for more details)."[3]

The attribute information[4] is as follows.

Attributes for both student-mat.csv (Math course) and student-por.csv (Portuguese language course) datasets:
1 school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira) 2 sex - student's sex (binary: 'F' - female or 'M' - male) 3 age - student's age (numeric: from 15 to 22)

[2] http://www.cs.waikato.ac.nz/ml/weka/
[3] http://mlr.cs.umass.edu/ml/datasets/Student+Performance
[4] http://mlr.cs.umass.edu/ml/datasets/Student+Performance

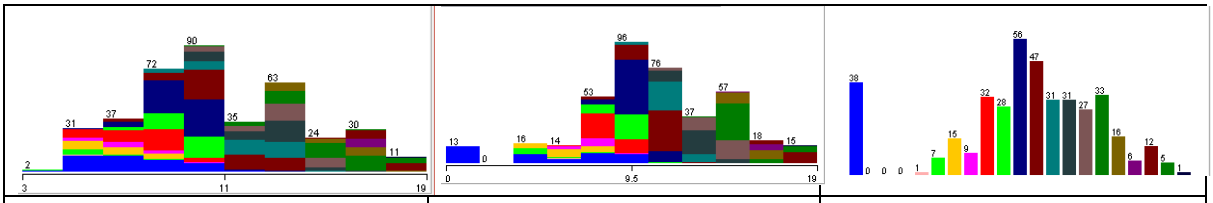
4 address - student's home address type (binary: 'U' - urban or 'R' - rural)
5 famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6 Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7 Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8 Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9 Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10 Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11 reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12 guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13 traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14 studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15 failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16 schoolsup - extra educational support (binary: yes or no)
17 famsup - family educational support (binary: yes or no)
18 paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
19 activities - extra-curricular activities (binary: yes or no)
20 nursery - attended nursery school (binary: yes or no)
21 higher - wants to take higher education (binary: yes or no)
22 internet - Internet access at home (binary: yes or no)
23 romantic - with a romantic relationship (binary: yes or no)
24 famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25 freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26 goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27 Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28 Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29 health - current health status (numeric: from 1 - very bad to 5 - very good)
30 absences - number of school absences (numeric: from 0 to 93)

these grades are related with the course subject, Math or Portuguese:
31 G1 - first period grade (numeric: from 0 to 20)
31 G2 - second period grade (numeric: from 0 to 20)
32 G3 - final grade (numeric: from 0 to 20, output target)

The data set exploration in IPython Notebook as well as attribute information given above provided valuable information regarding the data set. Some of the important discoveries are as follows.

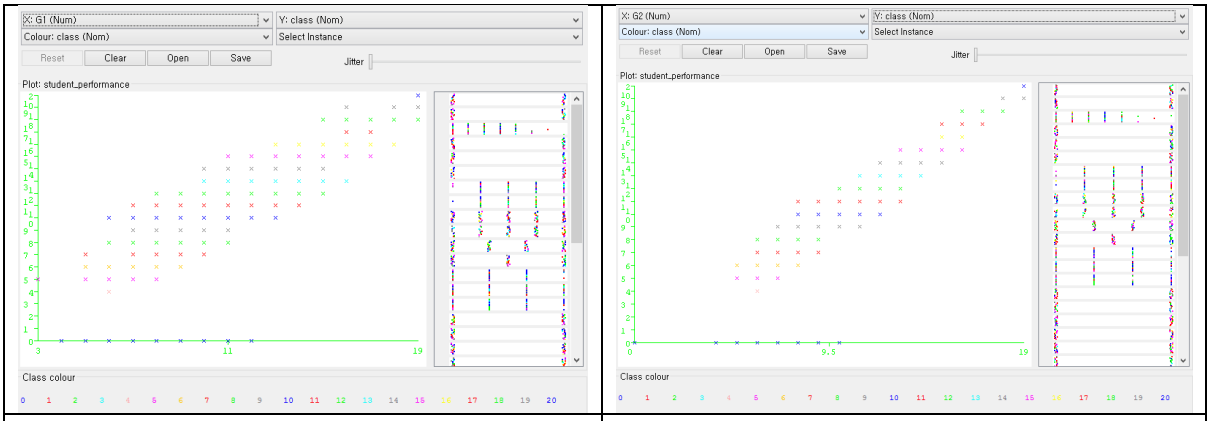
- There is a total of 395 instances and 32 attributes.
- G3 is the output label. In other words, all 32 attributes other than G3 are independent variable predicting dependent variable, G3.
- G3 has a [0, 20] range. If classification model has to predict 1 class out of the 20 possible class labels with only 395 instances, it would be difficult. It seems that the number of class labels should be less in order for classification models to show reasonable accuracy.
- There are no missing values for any of the given attributes. In fact, from reading their paper, I could identify that they also had ‘income’ attribute. In the end, however, they did not include it in the data set, because some left this question blank—as it is probably a sensitive inquiry. In any case, this makes the dataset robust to pre-processing issues.
- There is a mix of numeric and nominal attribute.

- There is a bit of imbalance in the attributes, such as *school*, *address*, *famsize, Pstatus*.
- The attributes, such as G1, G2 and G3, exhibit Gaussian distribution, as shown in Fig. 1.
- Through visualization, G1 and G2 are confirmed to have high correlation with G3, except for very few outliers. Figure 2 illustrates such correlation.



G1	G2	G3

Figure 1. Grade distribution



G1 vs G3 (class)	G2 vs G3 (class)

Figure 2. Correlation between G1,G2 and G3

2.0 Data Pre-processing for 'Student Performance Data Set'

2.1 Change the format from CSV to ARFF

The downloaded data came in csv and R format. Thus, in order to use the data set in Weka, it was pre-processed with python in IPython notebook.

The following image is the data as it came in csv format.

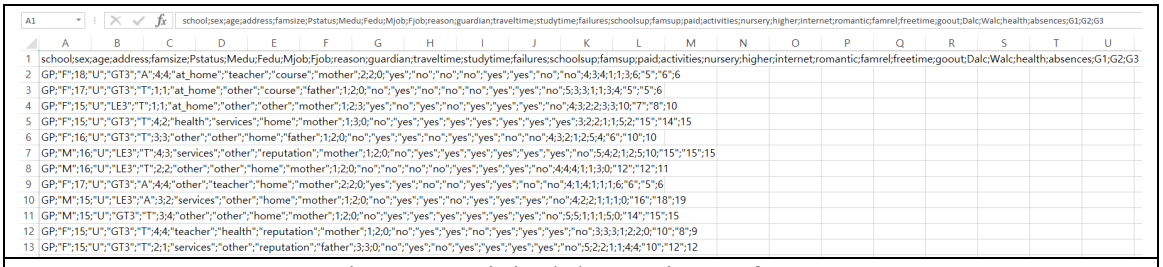


Figure 3. Original dataset in csv format

With references to python and syntactic structure specified by attribute-relation file format(arff)[5], which was developed by University of Waikato to use in Weka, the dataset was transitioned to the following file in arff format.

```
@RELATION student_performance

@ATTRIBUTE school {GP, MS}
@ATTRIBUTE sex {M, F}
@ATTRIBUTE age REAL
@ATTRIBUTE address {U, R}
@ATTRIBUTE famsize {GT3,LE3}
@ATTRIBUTE Pstatus {A,T}
@ATTRIBUTE Medu REAL
@ATTRIBUTE Fedu REAL
@ATTRIBUTE Mjob {at_home,health,other,services,teacher}
@ATTRIBUTE Fjob {at_home,health,other,services,teacher}
@ATTRIBUTE reason {course,home,other,reputation}
@ATTRIBUTE guardian {father,mother,other}
@ATTRIBUTE traveltime REAL
@ATTRIBUTE studytime REAL
@ATTRIBUTE failures REAL
@ATTRIBUTE schoolsup {yes,no}
@ATTRIBUTE famsup {yes,no}
@ATTRIBUTE paid {yes,no}
@ATTRIBUTE activities {yes,no}
@ATTRIBUTE nursery {yes,no}
@ATTRIBUTE higher {yes,no}
@ATTRIBUTE internet {yes,no}
@ATTRIBUTE romantic {yes,no}
@ATTRIBUTE famrel REAL
@ATTRIBUTE freetime REAL
@ATTRIBUTE goout REAL
@ATTRIBUTE Dalc REAL
@ATTRIBUTE Walc REAL
@ATTRIBUTE health REAL
@ATTRIBUTE absences REAL
@ATTRIBUTE G1 REAL
@ATTRIBUTE G2 REAL
@ATTRIBUTE class {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20}

@DATA
GP,"F",18,"U","GT3","A",4,4,"at_home","teacher","course","mother",2,2,0,"yes","no","no","no","yes","yes","no","no",4,3,4,1,1,3,6,"5","6",6
GP,"F",17,"U","GT3","T",1,1,"at_home","other","course","father",1,2,0,"no","yes","no","no","no","yes","yes","no",5,3,3,1,1,3,4,"5","5",6
GP,"F",15,"U","LE3","T",1,1,"at_home","other","other","mother",1,2,3,"yes","no","yes","no","yes","yes","yes","no",4,3,2,2,3,3,10,"7","8",10
GP,"F",15,"U","GT3","T",4,2,"health","services","home","mother",1,3,0,"no","yes","yes","yes","yes","yes","yes","yes",3,2,2,1,1,5,2,"15","14",15
GP,"F",16,"U","GT3","T",3,3,"other","other","home","father",1,2,0,"no","yes","yes","no","yes","yes","no","no",4,3,2,1,2,5,4,"6","10",10
GP,"M",16,"U","LE3","T",4,3,"services","other","reputation","mother",1,2,0,"no","yes","yes","yes","yes","yes","yes","no",5,4,2,1,2,5,10,"15","15",15
GP,"M",16,"U","LE3","T",2,2,"other","other","home","mother",1,2,0,"no","no","no","no","yes","yes","yes","no",4,4,4,1,1,3,0,"12","12",11
GP,"F",17,"U","GT3","A",4,4,"other","teacher","home","mother",2,2,0,"yes","yes","no","no","yes","yes","no","no",4,1,4,1,1,1,6,"6","5",6
GP,"M",15,"U","LE3","A",3,2,"services","other","home","mother",1,2,0,"no","yes","yes","no","yes","yes","yes","no",4,2,2,1,1,1,0,"16","18",19
GP,"M",15,"U","GT3","T",3,4,"other","other","home","mother",1,2,0,"no","yes","yes","yes","yes","yes","yes","no",5,5,1,1,1,5,0,"14","15",15
GP,"F",15,"U","GT3","T",4,4,"teacher","health","reputation","mother",1,2,0,"no","yes","yes","no","yes","yes","yes","no",3,3,3,1,2,2,0,"10","8",9
GP,"F",15,"U","GT3","T",2,1,"services","other","reputation","father",3,3,0,"no","yes","no","yes","yes","yes","yes","no",5,2,2,1,1,4,4,"10","12",12
GP,"M",15,"U","LE3","T",4,4,"health","services","course","father",1,1,0,"no","yes","yes","yes","yes","yes","yes","no",4,3,3,1,3,5,2,"14","14",14
```

Figure 4. Transformation into arff format

The following image shows that it was successfully loaded to Weka.

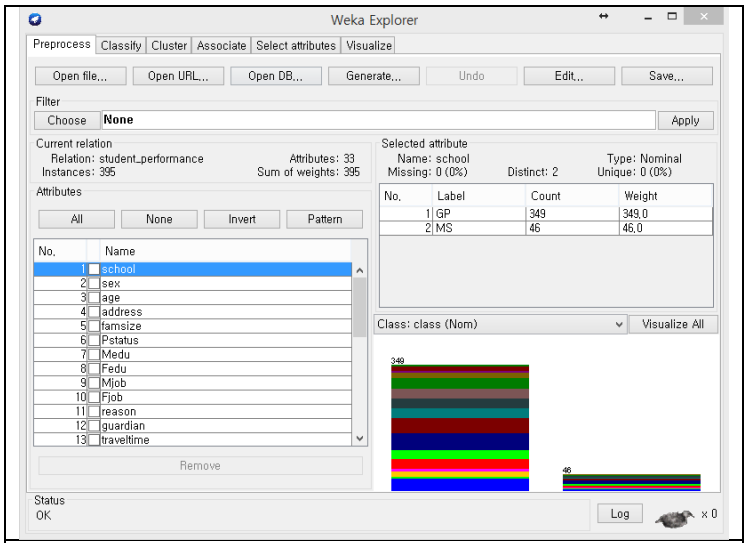


Figure 5. Student Performance Data Set on Weka

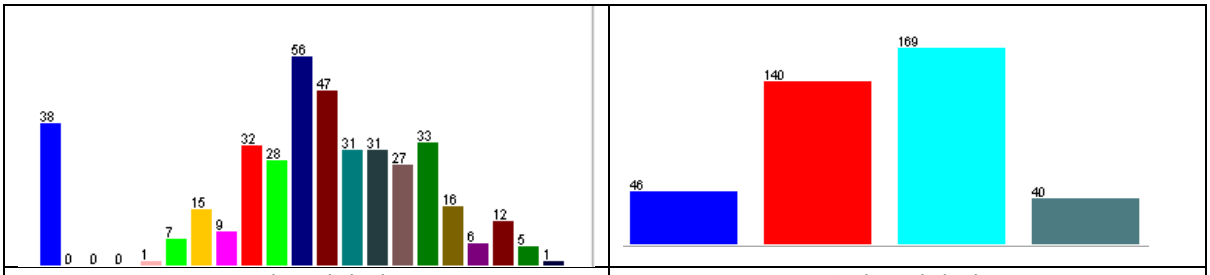
[5] http://www.cs.waikato.ac.nz/ml/weka/arff.html

2.2 Change the number of target class clusters

Initially, the target output class ranges from 0 to 20, and there are 21 clusters (cf. Figure 4). This is an unreasonable setting for the classification task, because it makes it extremely difficult to classify—remember that the number of instances we have is only 395. As a result, I have mapped a group of clusters to a few clusters [1,4], as indicated in Table 1 and Figure 6. This now makes classification task a reasonable task.

Range of initial class	New cluster number
0 ~ 5	1
6 ~ 10	2
11 ~ 15	3
16 ~ 20	4

Table 1



21 class labels | 4 class labels

Figure 6. Number of target class clusters

2.3 Remove outliers from G1 vs G3 and G2 vs G3 graph

Since Fig.2 confirms that G1 and G2 have high correlation with G3, I hypothesized that with G1 and G2 alone, it may be possible to get good enough result. Thus, it was assumed that removing the outliers in G1 vs G3 graph would help classification. However, removing the outliers on G1 vs G3 graph resulted in rather significant loss of accuracy, as shown in Table 1. This showed that G1 and G2 may not assume such a significant factor in helping to predict G3.

Dataset	Instances	Accuracy (J48)
Original	395	79.49%
Outliers (G1 vs G3) removed	357	38.38%
Outliers (G2 vs G3) removed	370	39.73%

Table 2

3.0 Classification Models for 'Student Performance Data Set'

In order to find a classifier algorithm(s) to best generalize the data, this section concentrates on identifying various classifiers, identifying which work better than others and choosing the most efficient algorithms and further refining their parameters further to

increase their generalization accuracy. Note that all the accuracy was calculated using 10-fold cross validation.

3.1 Benchmark Models

Several models were chosen and applied to the sample dataset. These models include, Naive Bayes, k-nearest neighbor, Logistic regression, J4.8, RandomForest, OneR, JRip, ZeroR. Each algorithm was applied using its default parameters. *K*-nearest neighbor's *k* value was chosen by user, and this chosen value is displayed on each table where appropriate. The algorithm with best accuracy is underlined.

Model	*Accuracy*
Naïve Bayes	73.92%
k-nearest neighbor (k=4)	44.30%
Logistic regression	42.78%
J4.8	79.49%
JRip	81.01%
RandomForest	81.01%
Multi-Layer Perceptron	67.59%
ZeroR (baseline)	42.78%

3.2 Attribute Selection

As mentioned above, first reasonable assumption starts with identifying the extent to which G1 and G2 have influence on G3. However, as we have confirmed in Figure 2, G1 and G2 alone cannot be a significant factor for predicting G3.

In order to understand which attributes play an important role, we referred to a tree structure generated by J4.8, as shown in Fig. 5. In detail, the tree had 30 leaves and 59 as its size. The analysis reveals that G2 is the most significant attribute (as expected); it is the root node (cf. Figure 7). This is reasonable, as G2 is the test score students receive before G3. Obviously, students who do well on the previous test will do well on the next test, as it can be assumed that the test contents may be related. Even if not, it is a good indicator that a student is preparing for his or her exams well.

In any case, examination of the tree suggested that attributes, such as *age*, *activities*, *failures*, have minimal influence—you can see them at the lowest node, while attributes, *G2, absences, G1, traveltime, famrel*, are the most important attributes.

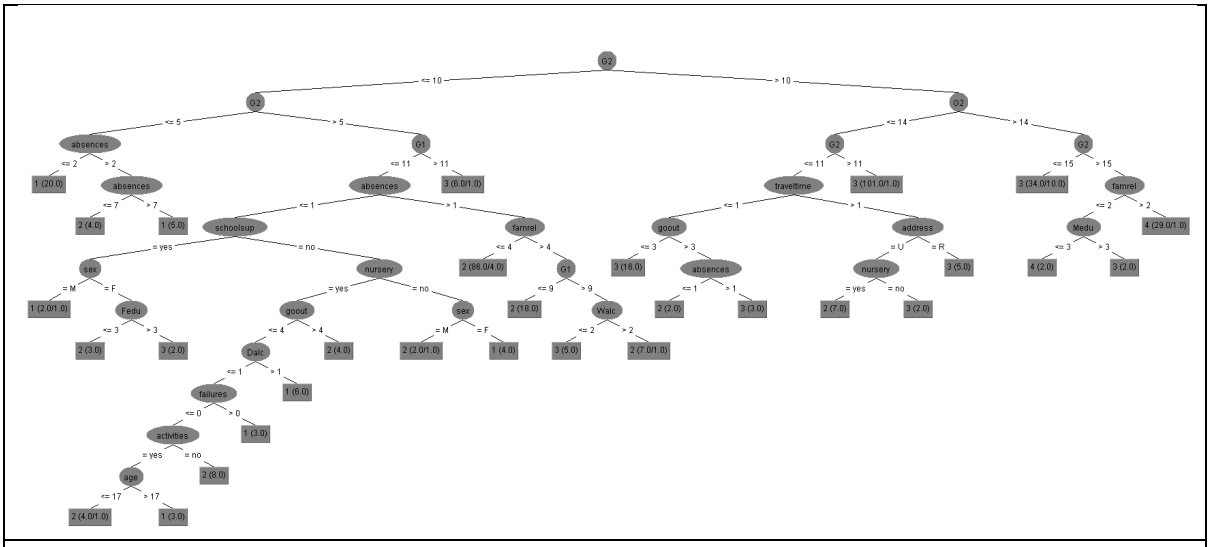


Figure 7. Tree generated by J4.8

Only using these significant attributes, we attempted classifying seven main models with default values and recorded their progress. The algorithm that has performed the best is underlined.

Model	*Accuracy*
Naïve Bayes	80.76%
k-nearest neighbor (k=4)	76.96%
Logistic regression	82.03%
J4.8	81.77%
JRip	81.77%
Random Forest	83.79%
Multi-Layer Perceptron	78.73%
ZeroR(baseline)	42.78%

Naïve Bayes has increased by more than 6%, while k-nearest neighbor has increased from 44.30% to 76.96%. Also, logistic regression increased from 42.78% to 82.03%, with J4.8 seeing approximately 2% increase as well. JRip improve by little (0.76%), but overall, the accuracy has all increased in main models. The bold style indicates that the accuracy has increased. Fig. 8 is the tree generated by J4.8 with only five most significant attributes. The number of leaves is 31 and its size is 61, which is not much different from the tree in Fig. 7. But it clearly shows that those five attributes are good discriminants.

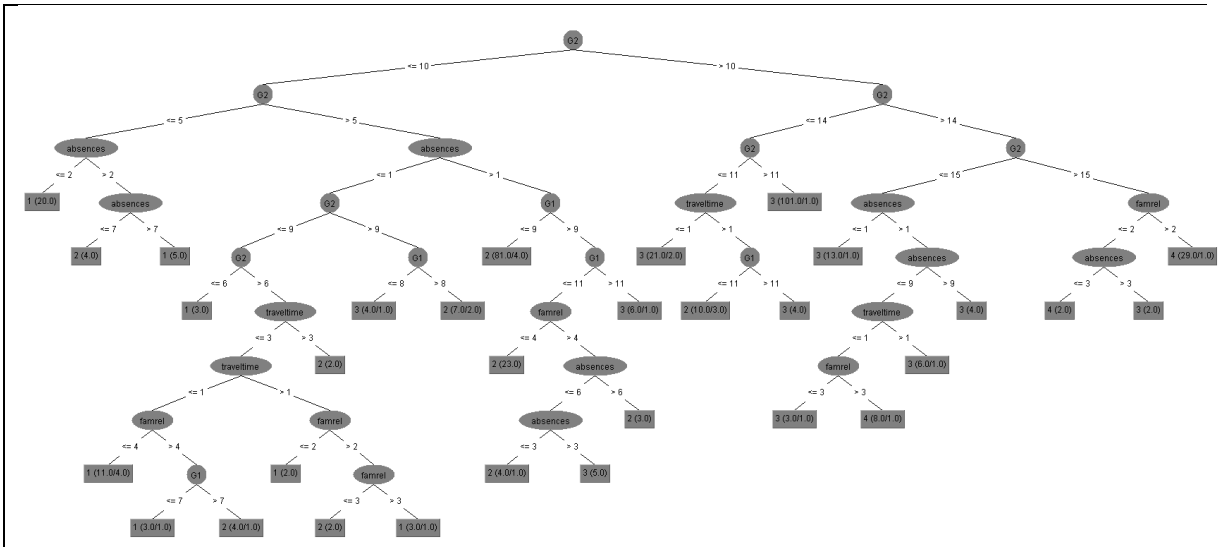


Figure 8. Tree generated by J4.8 with five most significant attributes only

3.3 Model Development

3.3.1 Naive Bayes

When we experimented with all 32 attributes, Naïve Bayes showed **73.92%** in accuracy. With five most significant attributes, its accuracy increased to **80.76%**.

To see if estimation can be improved, we set *useKernelEstimator* parameter to true. The accuracy increased to **81.52%**. After that, we applied discretization on the model by setting *useSupervisedDiscretization* parameter to true (with *useKernelEstimator* at false). The accuracy was even higher at **82.28%**.

3.3.2 K-nearest Neighbor

As for K-nearest neighbor, the accuracy with k=1 was **41.77%**, and the accuracy after k=4 did not improve much. Since k-means (at k=1) is equivalent to logistic regression and ZeroR, it is understandable that those three methods outputted accuracy in the similar range (i.e. accuracy within **40~43%**).

After removing all insignificant attributes and running K-nearest neighbor(k=1) with five most significant attributes, the accuracy jumped to **76.96%**. K-nearest neighbor at k=10 was even higher at **79.49%**.

Note that even with different distance weighting schemes, the accuracy of K-nearest neighbor with all 32 attributes never went beyond **43%**. In other words, reducing the number of variables seem to be the necessary setting if K-nearest neighbor is to be used. Refer to Table 3 for detailed accuracy record.

	Accuracy (32 attributes)	Accuracy (5 attributes)
Standard	44.30%	76.96%
1/distance	42.78%	75.94%
1-distance	42.78%	76.96%
Number of attributes	32 attributes	5

Table 3

3.3.3 Logistic Regression

After testing with five most significant attributes only, the accuracy drastically increased to **82.02%**, which is the best out of all the models we tested so far. To see if we can improve by varying ridge parameter, we experimented and was able to see that accuracy only decreased and that the default value is optimal for getting the best accuracy, as shown in Table 4.

Ridge parameter	**Accuracy**
1 x 10-8	**82.02%**
1 x 10-4	**82.02%**
1	**80.50%**
10	**73.67%**
20	**69.62%**
Table 4	

3.3.4 Decision Trees

Decision Tree was able to improve by getting rid of less significant attributes. Its accuracy increased from 79.49% to 81.77%, albeit minimal. We experimented with complexity control to see if the performance can be improved. As indicated in Table 5, setting unpruned parameter to true increased accuracy to **82.78%**, which is above the best accuracy by far set by logistic regression above.

Complexity Control	Parameter Value	Accuracy
unpruned	False	81.77%
unpruned	**True**	82.78%
minNumObj (unpruned==true)	3	82.78%
minNumObj	4	81.2658%
Table 5		

3.3.5 JRip

JRip consistently performed well, regardless of whether how many attributes and which attributes were used. The accuracy was in the range of 81.01% ~ 81.77%. We changed the number of folds to see if further improvement can be made.

Folds	Accuracy
1	81.26%
2	81.26%
3	81.77%
4	81.77%
5	81.77%
10	81.77%
Table 6	

No significant improvement was observed with changes in number of folds.

3.3.6 Random Forest

Random Forest is known for its powerfulness. In addition to powerful decision tree representation, it is capable of generalizing well. Indeed, it exhibited the best performance (**83.79%**) by far. Even with various experimentation with parameters in different models, it was not enough to beat Random Forest.

3.3.7 Multi-Layer Perceptron

With attribute selection, Multi-Layer Perceptron improved from 67.59% to 78.73%. Multi-Layer Perceptron is a very powerful algorithm suitable for complex non-linear functions. The reason the performance increase is not as high as any other or that the performance is not shown to be the best seems to rely on the fact that the classifier for this dataset does not have to be complex. This result is reasonable, because the relationship between the influential attributes and student performance is most likely linear.

4.0 Model Selection

Though no single model stands above all the other models by high margin, it is clear that Random Forest beats all the other models in performance. Since Random Forest is robust to overfitting, it is a satisfying choice for our classifier.

5.0 Evaluation & Conclusion for 'Student Performance Data Set'

The experiment with the student performance data set was gratifying in that it provided me with a chance to take a shot at educational data mining. While this experiment and examination was very extensive, I think much more interesting insights can still be mined from this data set. I will leave this as a part of future work.

Reference

P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7. [Web Link[6]]

S. Harvey. Mining Information from US Census Bureau Data.

[6] http://www3.dsi.uminho.pt/pcortez/student.pdf

Part B. 'Turkiye Student Evaluation Data Set'

1.0 Data Exploration for 'Turkiye Student Evaluation Data Set'

A. Student Evaluation Data Set

A-1. Data Set Information:

"This data set contains a total 5820 evaluation scores provided by students from Gazi University in Ankara (Turkey). There is a total of 28 course specific questions and additional 5 attributes."[7]

The attribute information[8] is as follows.

Name of attribute	Comment	Possible values
instr	Instructor's identifier	{1,2,3}
class	Course code	{1-13}
nb.repeat	Number of times the student is taking this course	{1,2,3}
attendance	Code of the level of attendance	{0,1,2,3}
difficulty	Level of the difficulty of the course	{1,2,3,4,5}
Q1	The semester course content, teaching method and evaluation system were provided at the start.	{1,2,3,4,5}
Q2	The course aims and objectives were clearly stated at the beginning of the period.	{1,2,3,4,5}
Q3	The course was worth the amount of credit assigned to it.	{1,2,3,4,5}
Q4	The course was taught according to the syllabus announced on the first day of class.	{1,2,3,4,5}
Q5	The class discussions, homework assignments, applications and studies were satisfactory.	{1,2,3,4,5}
Q6	The textbook and other courses resources were sufficient and up to date.	{1,2,3,4,5}
Q7	The course allowed field work, applications, laboratory, discussion and other studies.	{1,2,3,4,5}
Q8	The quizzes, assignments, projects and exams contributed to helping and learning.	{1,2,3,4,5}
Q9	I greatly enjoyed the class and was eager to actively participate during the lectures.	{1,2,3,4,5}
Q10	My initial expectations about the course were met at the end of the period or year.	{1,2,3,4,5}
Q11	The course was relevant and beneficial to my professional development.	{1,2,3,4,5}
Q12	The course helped me look at life and the world with a new perspective.	{1,2,3,4,5}
Q13	The instructor's knowledge was relevant and up to date.	{1,2,3,4,5}
Q14	The instructor came prepared for classes.	{1,2,3,4,5}
Q15	The instructor taught in accordance with the announced lesson plan.	{1,2,3,4,5}
Q16	The instructor was committed to the course and was understandable.	{1,2,3,4,5}

[7] http://mlr.cs.umass.edu/ml/datasets/Turkiye+Student+Evaluation#
[8] http://mlr.cs.umass.edu/ml/datasets/Turkiye+Student+Evaluation#

Q17	The instructor arrived on time for classes.	{1,2,3,4,5}
Q18	The instructor has a smooth and easy to follow delivery/speech.	{1,2,3,4,5}
Q19	The instructor made effective use of class hours.	{1,2,3,4,5}
Q20	The instructor explained the course and was eager to be helpful to students.	{1,2,3,4,5}
Q21	The instructor demonstrated a positive approach to students.	{1,2,3,4,5}
Q22	The instructor was open and respectful of the views of students about the course.	{1,2,3,4,5}
Q23	The instructor encouraged participation in the course.	{1,2,3,4,5}
Q24	The instructor gave relevant homework assignments/projects, and helped/guided students.	{1,2,3,4,5}
Q25	The instructor responded to questions about the course inside and outside of the course.	{1,2,3,4,5}
Q26	The instructor's evaluation system (midterm and final questions, projects, assignments, etc.) effectively measured the course objectives.	{1,2,3,4,5}
Q27	The instructor provided solutions to exams and discussed them with students.	{1,2,3,4,5}
Q28	The instructor treated all students in a right and objective manner.	{1,2,3,4,5}

The data set exploration in IPython Notebook as well as attribute information given above provided valuable information regarding the data set. Some of the important discoveries are as follows.

- There is a total of 5,820 instances and 33 attributes.
- 'nb.repeat' ought to be the output label we predict, the rest are the independent variables.
- There are no missing values for any of the given attributes, so there is no concern or need for pre-processing the data.
- All attributes are numeric.

As seen from Fig. 9, most students have 're-taken' the course only once, which seems reasonable. But this may make our plan to build an interpretable, acceptable classifier a bit difficult, since the distribution is too skewed. We will confirm later in Section 4.0 but this skewed distribution may lead a very simple classifier, ZeroR, to exhibit good enough classification accuracy.

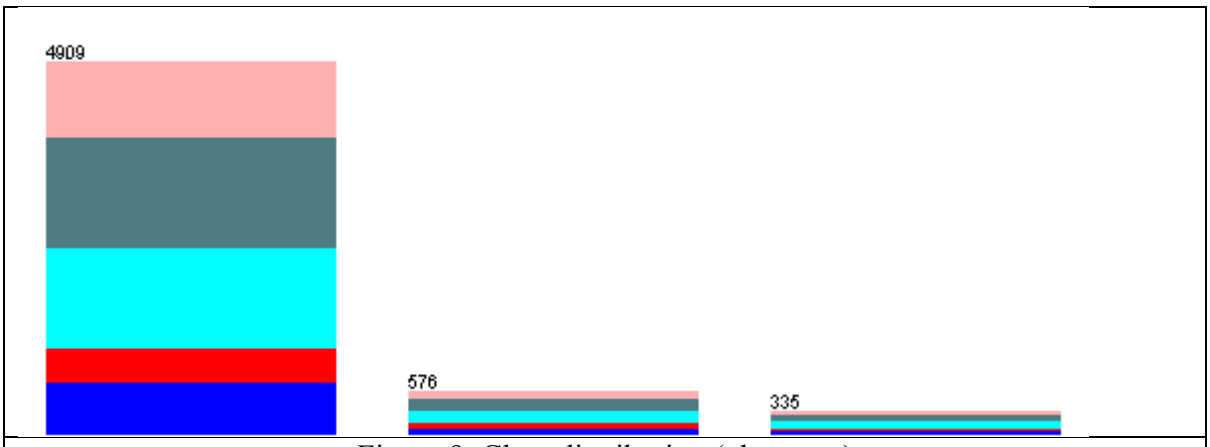


Figure 9. Class distribution (nb.repeat)

Note that the distributions of answers from Q1 to Q28 display similar distribution type (Fig. 10), another factor that may make it difficult for us to select particular attributes for attribute selection.

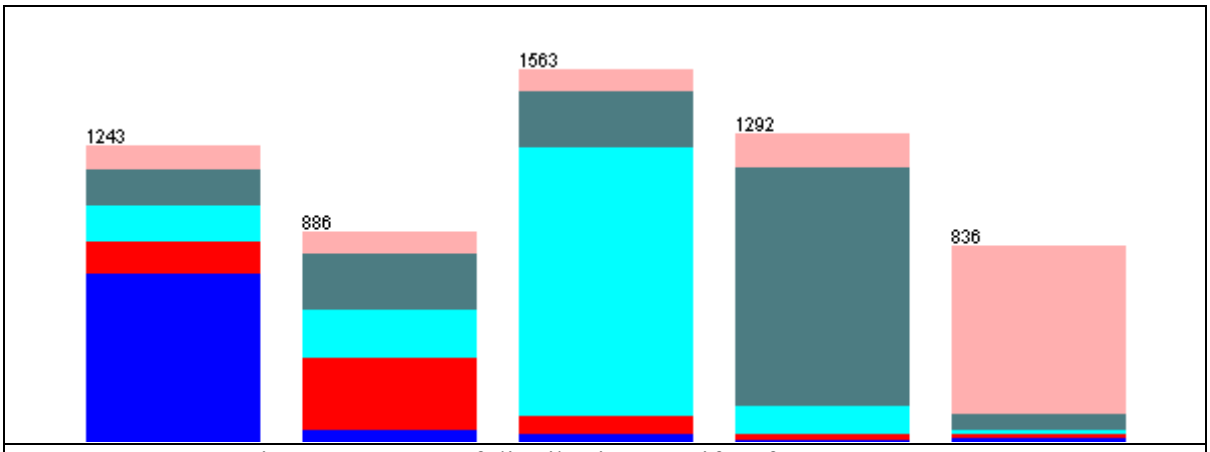


Figure 10. Type of distribution manifest from Q1 to Q28

In order to numerically assess whether they really exhibit similar pattern, I first computed percentage value on each nominal label (i.e. {1,2,3,4,5}) for each attribute from Q1 to Q28. Then, I calculated mean and standard deviation for values from Q1 to Q28. As shown in Table 7, it is clear that the attributes, Q1 ~ Q28, display similar distribution.

	Mean	Standard Deviation (STDEV)	Range (1 STDEV)
1	15.32	1.76	(13.6,17.1)
2	11.93	1.82	(10.11,13.8)
3	28.73	1.03	(27.7,29.7)
4	27.23	2.31	(24.9,29.5)
5	16.78	1.75	(15.0,18.5)

Table 7

2.0 Data Pre-processing for 'Turkiye Student Evaluation Data Set'

2.1 Change the format from CSV to ARFF

The downloaded data came in csv and R format. Thus, in order to use the data set in Weka, it was pre-processed with python in IPython notebook.

The following image is the data as it came in csv format.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
	instr	class	nb.repeat	attendanc	difficulty	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q1
1	1	1	2	1	0	4	3	3	3	3	3	3	3	3	3	
2	2	1	2	1	1	3	3	3	3	3	3	3	3	3	3	
3	3	1	2	1	2	4	5	5	5	5	5	5	5	5	5	
4	4	1	2	1	1	3	3	3	3	3	3	3	3	3	3	
5	5	1	2	1	0	1	1	1	1	1	1	1	1	1	1	
6	6	1	2	1	3	3	4	4	4	4	4	4	4	4	4	
7	7	1	2	1	1	3	4	4	4	4	4	4	4	4	4	
8	8	1	2	1	1	3	5	5	5	5	5	5	5	5	5	
9	9	1	2	1	1	3	4	4	4	4	4	4	4	4	4	
10	10	1	2	1	4	4	4	4	4	4	4	4	4	4	4	
11	11	1	2	1	3	4	3	3	3	3	3	3	3	3	3	
12	12	1	2	1	3	3	3	3	3	3	3	3	3	3	3	
13	13	1	2	1	2	3	4	4	4	4	4	4	4	4	4	
14	14	1	2	1	1	2	1	1	1	1	1	1	1	1	1	
15	15	1	2	1	2	3	1	1	1	1	1	1	1	1	1	
16	16	1	2	1	3	4	5	5	4	4	5	5	4	4	5	
17	17	1	2	1	3	4	1	1	1	1	1	1	1	1	1	

Figure 11. Original dataset in csv format

With references to python and syntactic structure specified by attribute-relation file format(arff)[9], which was developed by University of Waikato to use in Weka, the dataset was transitioned to the following file in arff format.

```
@relation student_evaluation

@attribute instr {1,2,3}
@attribute class {1,2,3,4,5,6,7,8,9,10,11,12,13}
@attribute nb.repeat {1,2,3}
@attribute attendance {0,1,2,3,4}
@attribute difficulty {1,2,3,4,5}
@attribute Q1 {1,2,3,4,5}
@attribute Q2 {1,2,3,4,5}
@attribute Q3 {1,2,3,4,5}
@attribute Q4 {1,2,3,4,5}
@attribute Q5 {1,2,3,4,5}
@attribute Q6 {1,2,3,4,5}
@attribute Q7 {1,2,3,4,5}
@attribute Q8 {1,2,3,4,5}
@attribute Q9 {1,2,3,4,5}
@attribute Q10 {1,2,3,4,5}
@attribute Q11 {1,2,3,4,5}
@attribute Q12 {1,2,3,4,5}
@attribute Q13 {1,2,3,4,5}
@attribute Q14 {1,2,3,4,5}
@attribute Q15 {1,2,3,4,5}
@attribute Q16 {1,2,3,4,5}
@attribute Q17 {1,2,3,4,5}
@attribute Q18 {1,2,3,4,5}
@attribute Q19 {1,2,3,4,5}
@attribute Q20 {1,2,3,4,5}
@attribute Q21 {1,2,3,4,5}
@attribute Q22 {1,2,3,4,5}
@attribute Q23 {1,2,3,4,5}
@attribute Q24 {1,2,3,4,5}
@attribute Q25 {1,2,3,4,5}
@attribute Q26 {1,2,3,4,5}
@attribute Q27 {1,2,3,4,5}
@attribute Q28 {1,2,3,4,5}

@data
1,2,1,0,4,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
1,2,1,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
1,2,1,2,4,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5,5
1,2,1,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3
1,2,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
```

Figure 12. Transformation into arff format

The following image shows that it was successfully loaded to Weka.

[9] http://www.cs.waikato.ac.nz/ml/weka/arff.html

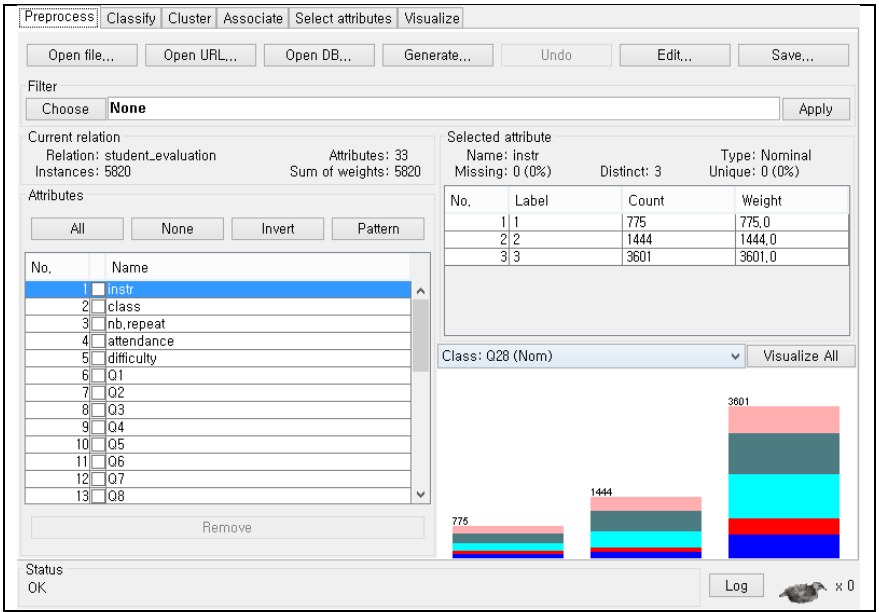


Figure 13. Student Evaluation Data Set on Weka

3.0 Classification Models for 'Turkiye Student Evaluation Data Set'

In order to find a classifier algorithm(s) to best generalize the data, this section concentrates on identifying various classifiers, identifying which work better than others and choosing the most efficient algorithms and further refining their parameters further increase their generalization accuracy.

3.1 Benchmark Models

Several models were chosen and applied to the sample dataset. These models included. Naive Bayes, k-nearest neighbor, Logistic regression, J4.8, RandomForest, OneR, KStar, JRip, ZeroR. Each algorithm was applied using its default parameters. *K*-nearest neighbor's *k* value was chosen by user, and this chosen value is displayed on each table where appropriate. The algorithm with best accuracy is underlined.

Model	*Accuracy*
Naïve Bayes	55.72%
k-nearest neighbor (k=4)	84.02%
Logistic regression	83.76%
J4.8	84.35%
RandomForest	83.09%
JRip	84.34%
Multi-Layer Perceptron	83.04%
ZeroR (baseline)	84.34%

3.2 Attribute Selection

The optimal tree generated by J4.8 has only one node, confirming that blindly selecting a single class (i.e. nb.repeat == 1) without taking any additional information (i.e. attributes)

into account still gives you the best performance. Also, as discussed in Section 1.0, the attribute distribution for Q1 to Q28 shows similar pattern. This implies that particular attributes from the pool (Q1 to Q28) may not be good discriminants. Indeed, as shown in Fig. 14, the patterns visible in red panel corroborates the similar in those attributes. Thus, I experimented by removing all the questionnaire attributes (Q1 to Q28) and tested the performance with major algorithms. Thus, the attributes used were 'instr', 'class', 'attendance', 'difficulty.'

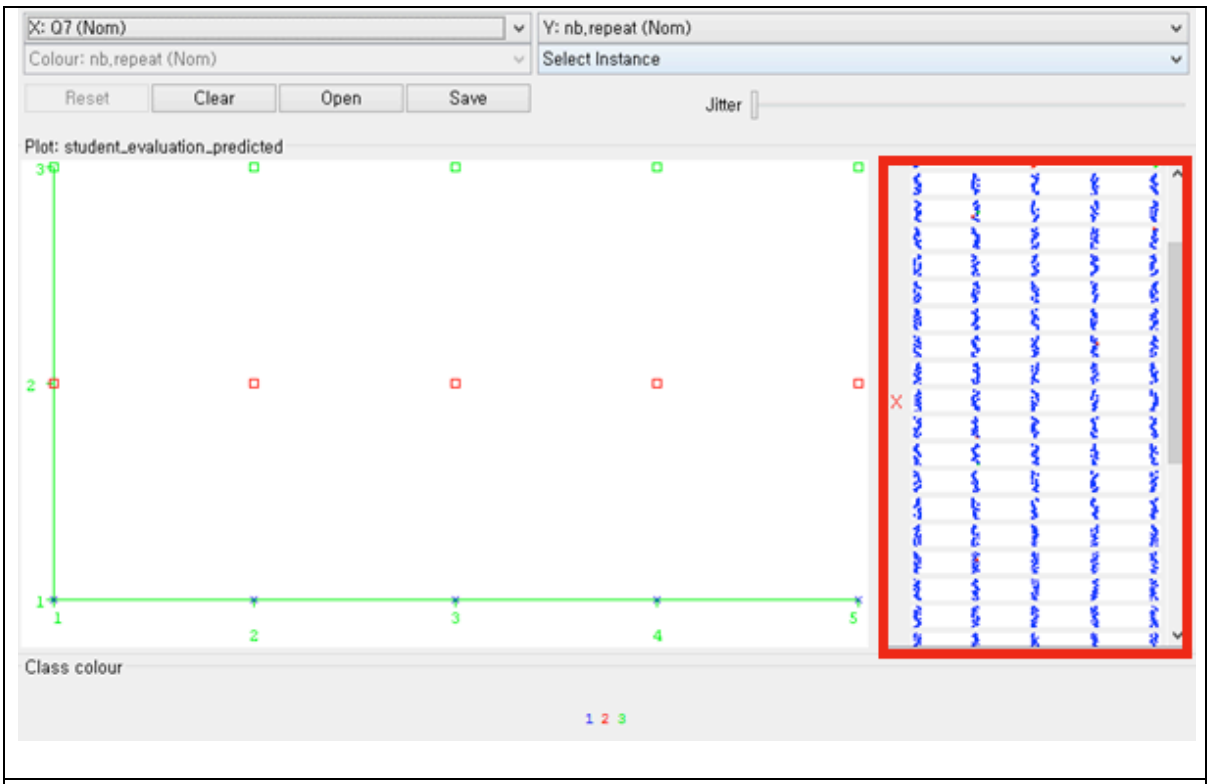


Figure 14. Classifier errors visualized

The following is a list of classification accuracy measured using seven main models with default values. The algorithm that has performed the best is underlined.

Model	*Accuracy (computing time)*
Naïve Bayes	84.24%
k-nearest neighbor (k=4)	84.16%
Logistic regression	84.33% (5 sec)
J4.8	84.34%
JRip	84.34%
Random Forest	83.64% (2 sec)
Multi-Layer Perceptron	83.73%
**ZeroR(baseline)*	84.34%

After removing insignificant attributes, Naïve Bayes increased by almost 30%, while k-nearest neighbor has seen only 0.14% increase in performance. Also, logistic regression increased by only 1%, with J4.8 and JRip practically seeing no improvement at all. Random

Forest improved by 0.6%. Overall, the accuracy has all increased in main models. The bold style in models indicates that the accuracy has increased.

3.4 Model Development

3.4.1 Naive Bayes

When we experimented with 32 attributes, the accuracy was at 55.72%. However, after using just 4 main attributes, Naïve Bayes showed 30% increase to boast the accuracy at **84.24%**. Given the nature of Naïve Bayes, this clearly demonstrates that the questionnaire attributes (Q1 to Q28) do not serve as supporting information for building a classifier.

3.4.2 K-nearest Neighbor

Compared to when we used 32 attributes, K-nearest Neighbor showed better performance with just 4 attributes at **84.16%**.

	Accuracy (33 attributes)	Accuracy (4 attributes)
Standard	85.39%	84.04%
1/distance	85.44%	84.04%
1-distance	85.46%	84.04%

Table 8

3.4.3 Logistic Regression

Building a logistic model is computationally very expensive. It took about 5 minutes to construct a model with 10-fold cross validation. Since it does not have significant advantage over all the other models, this model has a serious drawback when compared to other models.

Note that compared to using larger numbers of attributes, using fewer number of attributes gives computing time advantage while assuring even higher performance. When we computed using 32 attributes, it took 45 seconds, but with 4 attributes, the computational time was only 4 seconds.

Time (32 attributes)	Time (4 attributes)
45 seconds	4 seconds

Table 9

I experimented with different ridge parameters but was not successful at improving the current performance.

Ridge parameter	Accuracy
1 x 10-8	**84.33%**
1 x 10-4	**84.33%**
1	**84.33%**
10	**84.33%**
100	**84.31%**
Table 10	

3.4.4 Decision Trees

Since Decision Tree follows ZeroR structure, it obviously sees no improvement in performance. As a result, the performance is the same with ZeroR at **84.34%**. Although it was assumed to be the case that changing the structure from ZeroR would only mean decrease in performance as shown in Table 11, I continued with the experiment of changing complexity parameters to see how it affects performance. As expected, the performance only decreased.

Complexity Control	Parameter Value	Accuracy
unpruned	False	84.34%
unpruned	**True**	83.98%
minNumObj (unpruned==false)	3	84.34%
minNumObj (unpruned==false)	4	84.34%
Table 11		

3.4.5 JRip

Like Decision Tree, JRip saw no improvement in performance. I experimented with different folds but saw no significant improvement in accuracy, as shown in Table 12.

Folds	Accuracy
1	84.34%
2	84.30%
3	84.34%
4	84.34%
5	84.30%
10	84.34%
Table 12	

3.4.6 Random Forest

Random Forest witnessed minor improvement, increasing from **83.09%** to **83.64%**. Random Forest is very powerful, because it is robust to overfitting. In this case, however, since the dataset distribution is very skewed and simple, Random Forest seemed to be more than enough.

3.4.7 Multi-Layer Perceptron

The computational time with 32 attributes was too costly. It took almost 45 minutes to just calculate accuracy with 10-fold cross-validation. Since it did not produce an outstanding performance (obviously, since the classifier does not need to be complex), Multi-Layer Perceptron is definitely not a fitting algorithm for this data set. As shown in Table 13, however, it was a wise decision to select few important attributes, because it saved us 40 minutes with even better performance.

Time (32 attributes)	Time (4 attributes)
Approx. 45 minutes.	Approx. 4 minutes
Table 13	

4.0 Model Selection

In this experiment, it was clear that no algorithm can outperform the baseline method, ZeroR, which performes at **84.34%**. So the best model is ZeroR in this case.

5.0 Evaluation & Conclusion for 'Turkiye Student Evaluation Data Set'

Identifying and understanding what each attribute means was more interesting than the actual experiment itself, because the result was too obvious and did not propose any interesting insight. It would have been better, therefore, to have chosen a different attribute as a class. As a part of future work, it will be interesting to choose a different class, such as 'attendance', 'instr', and examine the relationship between these classes and other attributes.

Reference

Hajizaden et al. (2014). Analysis of factors that affect students' academic performance - Data Mining Approach. IJASCSE, Volume 3. Issue 8.

Gunduz, G. & Fokoue, E. (2013). UCI Machine Learning Repository [Web Link[10]]. Irvine, CA: University of California, School of Information and Computer Science.

[10] http://mlr.cs.umass.edu/ml/datasets/Turkiye+Student+Evaluation#

APPENDIX

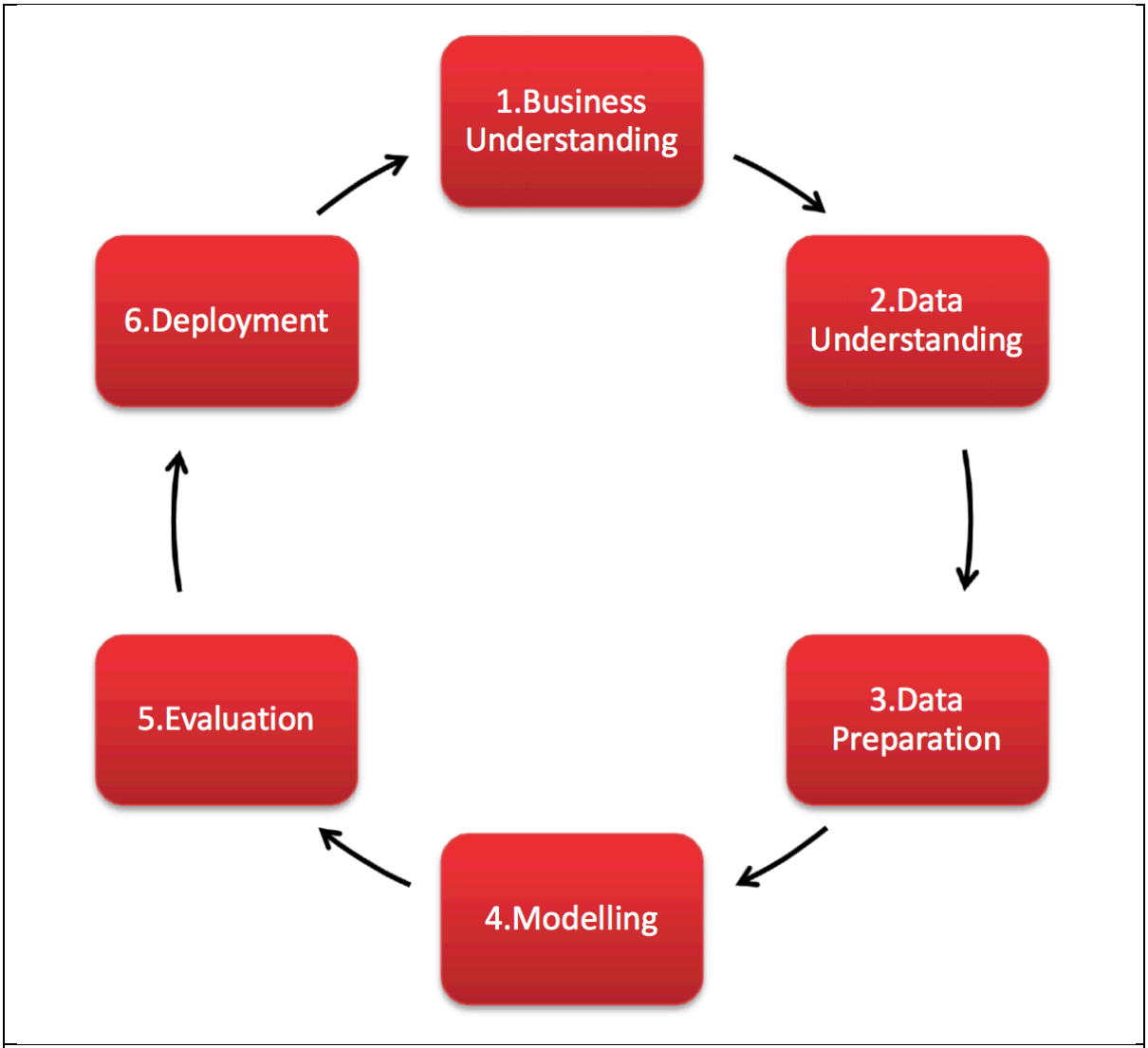


Figure A. CRISP-DM
(Cross-industry process for data mining)[11]

[11] http://www.sv-europe.com/crisp-dm-methodology/