Project Id: 16

Student dropout prediction using ML at secondary level in India

GitHub link: https://github.com/yaswanthkoravi/SMAI-project

Team members:

Name	Roll No
Yaswanth	2018202011
Gaurav	2018201027
Sarat	2018202013
Apurva Siruvolu	2018900022

Main goal : Data analysis and to provide classification algorithm which can able to classify the data with best accuracy.

***Problem definition* :** Using machine learning we have to perform binary classification on student dropout data to predict whether a particular student will dropout or not. First, after collecting data, we perform feature analysis to find out most dominant features in dataset and then use various classification algorithms on the extracted features to figure out best algorithm for this task.

***Results of the project* :** Extract the dominant features in dataset (figuring out the main reasons for student dropout) and comparing how various classification algorithms able to perform classification.

Tasks :

Name	Task
Yaswanth	KNN, SVM, Logistic regression (dataset1) and Gradient Boosting, neural networks (dataset 2)
Gaurav	Data analysis,KNN and logistic regression (dataset 2)
Sarat	Neural Networks, Gradient Boosting (dataset 1)
Apurva Siruvolu	Data analysis and SVM (dataset2)

Motivation for the project:

The power of education in empowering an individual and the society at large cannot be understated.

- While the enrollment in higher educational institutions has been rising steadily (from 10% in 2005 to 25% in 2015 -MHRD) the numbers are still abysmal.

- There is also significant drop after 10th standard due to various factors. This is even more troublesome as the the droppers can't really get any job these days without completing their schooling.

- We analyse a dataset from BODWAD district of Maharashtra and try to analyze the various factors which are responsible.

Data pre-processing :

One hot encoder representation used in :

- Social category
- Gender
- Religion

Standardization of data used in :

- Attendance
- Exam marks

Metric considered in model evaluation:

Since, a student who is likely to drop out classified as 'retained' is worse than vice-versa, the agenda is to select the model that provides sufficiently low false negatives with reasonable number of false positives. We use weighted accuracy in which the cost of True Positive Rate is higher than that of True Negative Rate.

$$\text{so weighted accuracy} = 0.7(TP/P) + 0.3(TN/N)$$

Train-test split :

Training data=10415 (60%)

Testing data=6944 (40%)

Model is evaluated and optimal parameters are found using 10 fold cross validation.

Total number of students: 17359

Number of features: 10

Attribute names : Gender,SocialCategory, Religion, BPL, Disadvantaged, FreeEducation, Attendance, Homeless, ExamMarks, Disablility, Dropout.

Number of students who were left: 3729
Number of students who were not left: 13630

Deploying Machine Learning Algorithms :

The supervised machine learning algorithms have been deployed for the binary classification of students into the categories of likely to ’drop-out’ or retain. Supervised algorithms work by learning the mapping function of input variables to output variables . This means that the algorithm learns the relationship between input and output variables and when the new data comes in it iteratively predicts the output till the error is reduced to an acceptable limit. The dataset acquired from U-DISE is randomly divided into training and testing set in 3:2 ratio. The algorithms used for the above task are Logistic Regression, K-Nearest Neighbours, Support Vector Machine, Neural Networks and Gradient Boost classifier.

Models :

KNN :

Parameters :

- K values : 1 to 50

Results :

- Optimal parameter values : k = 9
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5203	266
Actual positive	648	827

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(827/1475) + 0.3*(5203/5469)
  = 67.7%

Logistic regression:
Parameters:

- C values : 0.01, 0.1, 1, 10, 100
- Regularization : L1, L2

Results:

- Optimal parameter values : C = 10
- Regularization: L1
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5190	279
Actual positive	698	777

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(777/1475) + 0.3*(5190/5469)
  = 65.3%

SVM :

Parameters :

- C values : 0.01, 0.1, 1, 10,100
- Kernels : linear, rbf, poly

Results :

- Optimal parameter values : C = 100, Kernel = rbf
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5298	171
Actual positive	731	744

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(744/1475) + 0.3*(5298/5469)
  = 64.3%

Neural Networks :

Parameters :

- Hidden layers : 1, 2, 3
- Neurons : 3, 6, 9
- Activation : tanh, relu, sigmoid

Results :

- Optimal parameter values : 2 layers with 9 Neurons with tanh activation function.
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5194	275
Actual positive	600	875

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(875/1475) + 0.3*(5194/5469)
  = 70%

Gradient Boosting :

Parameters :

- Learning rates: 1, 1.5, 2
- Estimators : 20, 40, 60 ,80
- Max depth : 3, 4, 5, 6, 7, 8, 9, 10

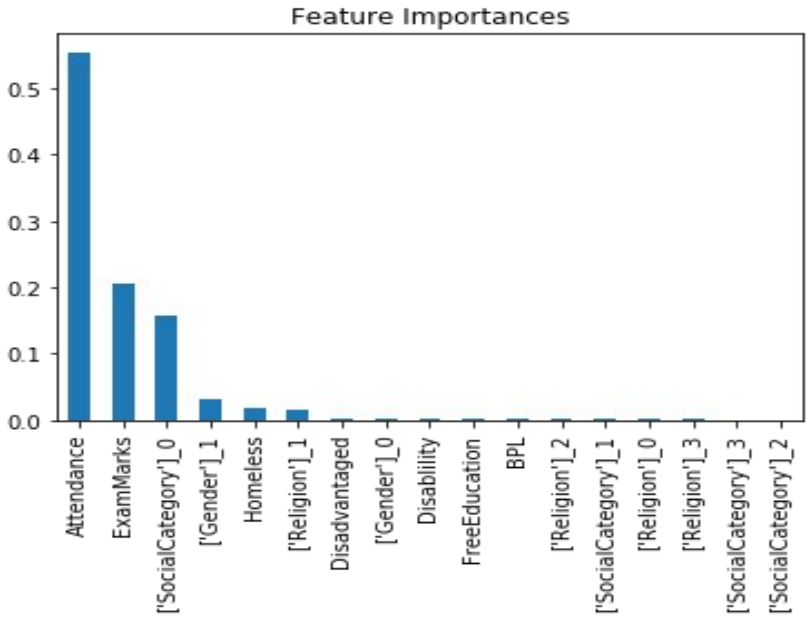
Results :

- Optimal parameter values : Learning rate = 1, max depth = 3, estimators = 20
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5167	302
Actual positive	571	904

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(904/1475) + 0.3*(5167/5469)
  = 71%

Feature comparision:



Comparison :

Model	Optimal parameters given in paper	Weighted accuracy (%)	Optimal parameters obtained	Weighted accuracy (%)
Knn	K = 7	67	K = 9	67.7
Logistic regression	-	63	C = 10 Regularization = L1	65.3
SVM	kernel = linear	63	C = 100 kernel = rbf	64.3
Neural networks	Number of hidden layers = 2 Number of neurons in each layer = 9 Activation function = sigmoid	70	Number of hidden layers = 2 Number of neurons in each layer = 9 Activation function = tanh	70
Gradient boosting	Learning rate = 1 Estimators = 50	78	Learning rate = 1 Max depth = 3 Estimators = 20	71

For dataset-2:

Data analysis:

Total number of students: 395

1. Number of features: 31
2. Number of students who were not left: 265
3. Number of students who left: 130

Attribute names : School, sex, age, address, famsize, Pstatus, Medu, Fedu, Mjob, Fjob, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, goout, Dalc, Walc, health, absences.

Train-test split :

1. Training data = 316 (80%)
2. Testing data = 79 (20%)
3. Model is evaluated and optimal parameters are found using 10 fold cross validation.

Models :

KNN :

Parameters :

- K values : 1 to 50

Results :

- Optimal parameter values : k = 35
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	1	23
Actual positive	0	55

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(55/55) + 0.3*(1/23)
  = 71.3%

Logistic regression :
Parameters:

- C values : 0.01, 0.1, 1, 10, 100
- Regularization : L1, L2

Results:

- Optimal parameter values : C = 0.001
- Regularization: L2
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	0	24
Actual positive	0	55

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(55/55) + 0.3*(0/24)
  = 70%

SVM :

Parameters :

- C values : 0.01, 0.1, 1, 10,100
- Kernels : linear, rbf, poly

Results :

- Optimal parameter values : C = 1, Kernel = rbf
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	5	19
Actual positive	2	53

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)
  = 0.7*(53/55) + 0.3*(5/24)
  = 73.7%

Neural Networks :

Parameters :

- Hidden layers : 1, 2, 3
- Neurons : 3, 6, 9
- Activation : tanh, relu, sigmoid

Results :

- Optimal parameter values : 2 layers with 6 Neurons with tanh activation function.
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	3	21
Actual positive	1	54

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)

  = 0.7*(54/55) + 0.3*(3/24)

  = 70%

Gradient Boosting :

Parameters :

- Learning rates: 0.01, 0.1, 1
- Estimators : 20, 40, 60 ,80
- Max depth : 3, 4, 5, 6, 7, 8, 9, 10

Results :

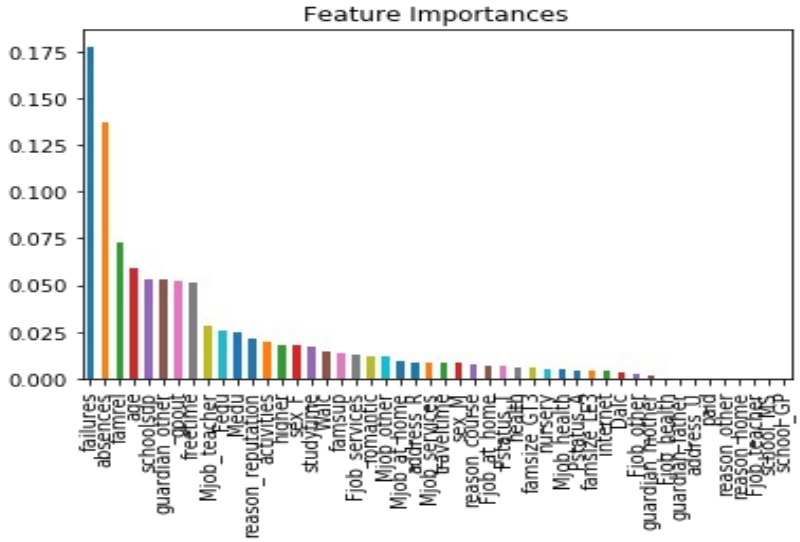
- Optimal parameter values : Learning rate = 0.1, estimators = 28
- Confusion matrix :

	Predicted negative	Predicted positive
Actual negative	10	14
Actual positive	7	48

- Weighted accuracy formula : 0.7*(TP/P) + 0.3*(TN/N)

= 0.7*(904/1475) + 0.3*(5167/5469)
= 71%

Feature comaprision :



Comaprison:

Model	Optimal parameters	Accuracy (%)
Knn	K = 35	71.2
Logistic regression	C = 0.001 Regularization = L2	70
SVM	C = 1 kernel = rbf	73.7
Neural networks	Number of hidden layers = 2 Number of neurons in each layer = 6 Activation function = relu	72
Gradient boosting	Learning rate = 0.1 Estimators = 28	74