International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

## COPY RIGHT

ELSEVIER
SSRN

IJIEMR Transactions, online available on 19th Dec 2022. Link :http://www.ijiemr.org/downloads.php?vol=Volume-11&issue=ISSUE-12

**DOI: 10.48047/IJIEMR/V11/ISSUE 12/33**

Title **TRACKING AND PREDICTING STUDENT PERFORMANCE IN ACADEMICS USING MACHINE LEARNING APPROACH**

Volume 11, Issue 12, Pages: 244-252

Paper Authors
**Nukam Mallesh, Dr.G.Venkata Rami Reddy**

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

# TRACKING AND PREDICTING STUDENT PERFORMANCE IN ACADEMICS USING MACHINE LEARNING APPROACH

**1 Nukam Mallesh ,** M.Tech in Data Sciences SIT JNTUH

**2 Dr.G.Venkata Rami Reddy**, Professor of It, M.Tech, Ph.D School of IT, JNTUH

**ABSTRACT:** Accurately predicting students' future performance based on their ongoing academic records is crucial for effectively carrying out necessary pedagogical interventions to ensure students' on-time and satisfactory graduation. Although there is a rich literature on predicting student performance when solving problems or studying for courses using data-driven approaches, predicting student performance in completing degrees (e.g. college programs) is much less studied and faces new challenges: (1) Students differ tremendously in terms of backgrounds and selected courses; (2) Courses are not equally informative for making accurate predictions; (3) Students' evolving progress needs to be incorporated into the prediction. In this paper, we develop a novel machine learning method for predicting student performance in degree programs that is able to address these key challenges. The proposed method has two major features. First, a bilayered structure comprising of multiple base predictors and a cascade of ensemble predictors is developed for making predictions based on students' evolving performance states. Second, a data-driven approach based on latent factor models and probabilistic matrix factorization is proposed to discover course relevance, which is important for constructing efficient base predictors. Through extensive simulations on an undergraduate student dataset collected over three years at UCLA, we show that the proposed method achieves superior performance to benchmark approaches.

***Keywords:*** *Student performance prediction, data-driven course clustering, personalized education.*

## 1. INTRODUCTION

Making higher education affordable has a significant impact on ensuring the nation's economic prosperity and represents a central focus of the government when making education policies [1]. Yet student loan debt in the United States has blown past the trillion-dollar mark, exceeding Americans' combined credit card and auto loan debts [2]. As the cost in college education (tuitions, fees and living expenses) has skyrocketed over the past few decades, prolonged graduation time has become a crucial contributing factor to the evergrowing student loan debt. In fact, recent studies show that only 50 of the more than 580 public four-year institutions in the United States have on-time graduation rates at or above 50 percent for their full-time students [2]. To make college more affordable, it is thus crucial to ensure that many more students graduate on time through early interventions on students whose performance will be unlikely to meet the graduation criteria of the degree program on time. A critical step towards effective intervention is to build a system that can continuously keep track of students' academic performance and accurately predict their future performance, such as when they are likely to graduate and their estimated final GPAs, given the current progress. Although predicting student performance has been extensively studied in the literature, it was primarily studied in the contexts of solving problems in Intelligent Tutoring Systems (ITSs) [3][4][5][6], or completing courses in classroom settings or in Massive Open Online Courses (MOOC) platforms [7][8]. However, predicting student performance within a degree program (e.g. college program) is significantly different and faces new challenges.

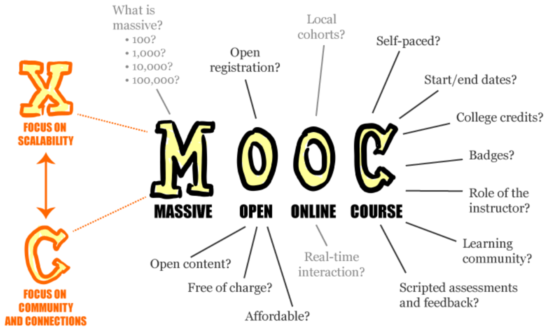


**Fig.1 MOOC model**

First, students can differ tremendously in terms of backgrounds as well as their chosen areas (majors, specializations), resulting in different selected courses as well as course sequences. On the other hand, the same course can be taken by students in different areas. Since predicting student performance in a particular course relies on the student past performance in other courses, a key challenge for training an effective predictor is how to handle heterogeneous student data due to different areas and interests. In contrast, solving problems in ITSs often follow routine steps which are the same for all students [9]. Similarly, predictions of students' performance in courses are often based on in-course assessments which are designed to be the same for all students [7].

Second, students may take many courses but not all courses are equally informative for predicting students' future performance. Utilizing the student's past performance in all courses that he/she has completed not only increases complexity but also introduces noise in the prediction, thereby degrading the prediction performance. For instance, while it makes sense to consider a student's grade in the course "Linear Algebra" for predicting his/her grade in the course "Linear Optimization", the student's grade in the course "Chemistry Lab" may have much weaker predictive power. However, the course correlation is not always as obvious as in this case. Therefore, discovering the underlying correlation among courses is of great importance for making accurate performance predictions.

Third, predicting student performance in a degree program is not a one-time task; rather, it requires continuous tracking and updating as the student finishes new courses over time. An important consideration in this regard is that the prediction needs to be made based on not only the most recent snapshot of the student accomplishments but also the evolution of the student progress, which may contain valuable information for making more accurate predictions. However, the complexity can easily explode since even mathematically representing the evolution of student progress itself can be a daunting task. However, treating the past progress equally as the current performance when predicting the future may not be a wise choice either since intuition tells us that old information tends to be outdated.

## 2. LITERATURE REVIEW

**Learning factors analysis–a general method for cognitive model evaluation and improvement**

A cognitive model is a set of production rules or skills encoded in intelligent tutors to model how students solve problems. It is usually generated by brainstorming and iterative refinement between subject experts, cognitive scientists and programmers. In this paper we propose a semi-automated method for improving a cognitive model called Learning Factors Analysis that combines a statistical model, human expertise and a combinatorial search. We use this method to evaluate an existing cognitive model and to generate and evaluate alternative models. We present improved cognitive models and make suggestions for improving the intelligent tutor based on those models.

**Addressing the assessment challenge with an online system that tutors as it assesses**

Secondary teachers across the United States are being asked to use formative assessment data (Black & Wiliam, 1998a, 1998b; Roediger & Karpicke, 2006) to inform their classroom instruction. At the same time, critics of US government's No Child Left Behind legislation are calling the bill "No Child Left Untested". Among other things, critics point out that every hour spent assessing students is an hour lost from instruction. But, does it have to be? What if we better integrated assessment into classroom instruction and allowed students to learn during the test? We developed an approach that provides immediate tutoring on practice assessment items that students cannot solve on their own. Our hypothesis is that we can achieve more accurate assessment by not only using data on whether students get test items right or wrong, but by also using data on the effort required for students to solve a test item with instructional assistance. We have integrated assistance and assessment in the ASSISTment system. The system helps teachers make better use of their time by offering instruction to students while providing a more detailed evaluation of student abilities to the teachers, which is impossible under current approaches. Our approach for assessing student math proficiency is to use data that our system collects through its interactions with students to estimate their performance on an end-of-year high stakes state test. Our results show that we can do a reliably better job predicting student end-of-year exam scores by leveraging the interaction data, and the model based on only the interaction information makes better predictions than the traditional assessment model that uses only information about correctness on the test items.

### 2.3 Personalized grade prediction: A data mining approach

To increase efficacy in traditional classroom coursesas well as in Massive Open Online Courses (MOOCs), automatedsystems supporting the instructor are needed. One importantproblem is to automatically detect students that are going todo poorly in a course early enough to be able to take remedialactions. This paper proposes an algorithm that predicts the finalgrade of each student in a class. It issues a prediction for eachstudent individually, when the expected accuracy of the predic-tion is sufficient. The algorithm learns online what is the optimalprediction and time to issue a prediction based on past historyof students' performance in a course. We derive demonstrate theperformance of our algorithm on a dataset obtained based onthe performance of approximately 700 undergraduate studentswho have taken an introductory digital signal processing overthe past 7 years. Using data obtained from a pilot course, ourmethodology suggests that it is effective to perform early in-classassessments such as quizzes, which result in timely performanceprediction for each student, thereby enabling timely interventionsby the instructor (at the student or class level) when necessary.Index Terms—Forecasting algorithms, online learning, gradeprediction, data mining, digital signal processing education.

### 2.4 Data mining for adaptive learning in a tesl-based e-learning system

This study proposes an Adaptive Learning in Teaching English as a Second Language (TESL) for e-learning system (AL-TESL-e-learning system) that considers various student characteristics. This study explores the learning performance of various students using a data mining technique, an artificial neural network (ANN), as the core of AL-TESL-e-learning system. Three different levels of teaching content for vocabulary, grammar, and reading were set for adaptive learning in the AL-TESL-e-learning system. Finally, this study explores the feasibility of the proposed AL-TESL-e-learning system by comparing the results of the regular online course control group with the AL-TESL-e-learning system adaptive learning experiment group. Statistical results show that the experiment group had better learning performance than the control group; that is, the AL-TESL-e-learning system was better than a regular online course in improving student learning performance.

## 3. IMPLEMENTATION

In fact, recent studies show that only 50 of the more than 580 public four-year institutions in the United States have on-time graduation rates at or above 50 percent for their full-time students . To make college more affordable, it is thus crucial to ensure that many more students graduate on time through early interventions on students whose performance will be unlikely to meet the graduation criteria of the degree program on time. A critical step towards effective intervention is to build a system that can continuously keep track of students' academic performance and accurately predict their future performance, such as when they are likely to graduate and their estimated final GPAs, given the current progress. Although predicting student performance has been extensively studied in the literature, it was primarily studied in the contexts of solving problems in Intelligent Tutoring Systems (ITSs).

### Disadvantages:

- However, predicting student performance within a degree program (e.g. college program) is significantly different and faces new challenges.

We consider a degree program in which students must complete a set of courses to graduate in T academic terms. Courses have prerequisite dependencies, namely a course can be taken only when certain prerequisite courses have been taken and passed. In general, the prerequisite dependency can be described as a directed acyclic graph (DAG) . There can be multiple specialized areas in a program which require different subsets of courses to be completed for students to graduate. We will focus on the prediction problem for one area in this department. Nevertheless, data from other areas will still be utilized for our prediction tasks. The reason is that data from a single area is often limited while different areas still share many common courses.

**Advantages:**

- It is important for constructing efficient base predictors.
- System that can continuously keep track of students' academic performance and accurately predict their future performance

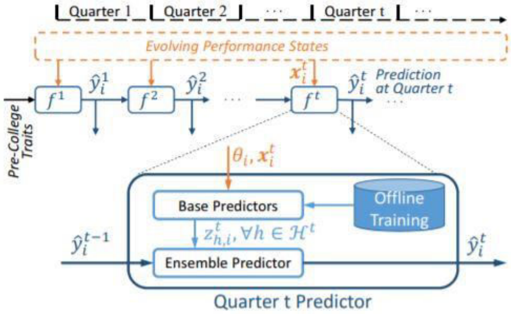


Fig.2: System architecture

**MODULES:**

In this project author is proposing following modules

1. Upload UCLA Students Dataset: Using this module we will load dataset into the application.
2. Matrix Factorization: Using this module we will build feature vector from dataset
3. Run SVM Algorithm: Using this module to train SVM classifier and to get it accuracy and MSE value
4. Run Random Forest Algorithm: Using this module to generate training model using Random Forest and to get it accuracy and MSE
5. Run Logistic Regression Algorithm: Using this module to generate training model using Logistic Regression and to get it accuracy and MSE
6. Propose Ensemble-based Progressive Prediction (EPP) Algorithm: Using this module to generate model using propose EPP algorithm and to get it accuracy and MSE
7. Predict Performance: Using this module to upload student on going test marks and to predict GPA for future course
8. Mean Square Error Graph: Using this module to get graph, through this we can conclude that propose EPP is better in prediction compare to other algorithms.

To implement this paper author using base classifiers such as Random Forest, SVM, Logistic Regression or KNN. The prediction results of base classifier will be pass to ensemble classifier to predict better results for ongoing courses and future courses. To implement this project author using student performance dataset from UCLA University and this dataset saved inside dataset folder.

## 4. ALGORITHMS

### SUPPORT VECTOR MACHINE (SVM):

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot). The SVM algorithm is implemented in practice using a kernel. The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra, which is out of the scope of this introduction to SVM.

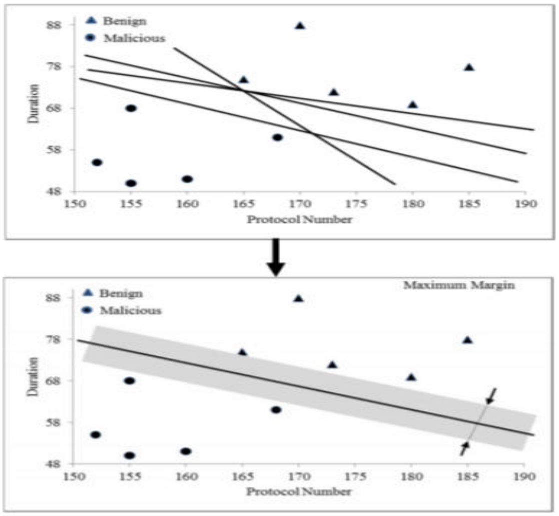


Fig.3: SVM model

**Random Forest Algorithm:**

Random Forest algorithm is a supervised classification algorithm. We can see it from its name, which is to create a forest by some way and make it random. There is a direct relationship between the number of trees in the forest and the results it can get: the larger the number of trees, the more accurate the result. But one thing to note is that creating the forest is not the same as constructing the decision with information gain or gain index approach.

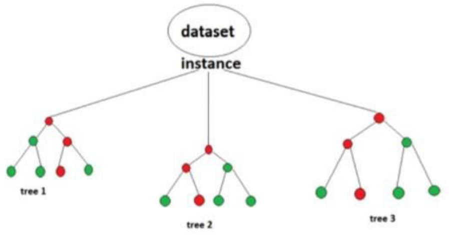


Fig.4: Random forest model

**Logistic Regression:**

The logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

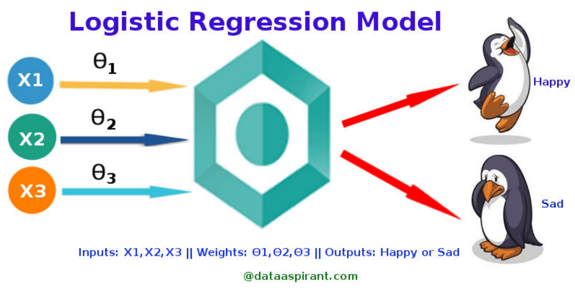

Logistic Regression Model
X1
X2
X3
θ1
θ2
θ3
Happy
Sad
Inputs: X1,X2,X3 || Weights: Θ1,Θ2,Θ3 || Outputs: Happy or Sad
@dataaspirant.com

Fig.5: Logistic regression model

Logistic regression becomes a classification technique only when a decision threshold is brought into the picture. The setting of the threshold value is a very important aspect of Logistic regression and is dependent on the classification problem itself. The decision for the value of the threshold value is majorly affected by the values of precision and recall. Ideally, we want both precision and recall to be 1, but this seldom is the case.

## 5. EXPERIMENTAL RESULTS

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

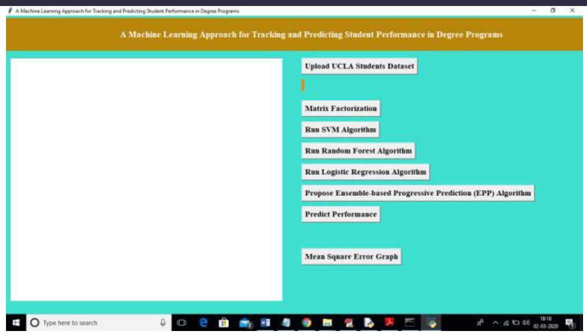
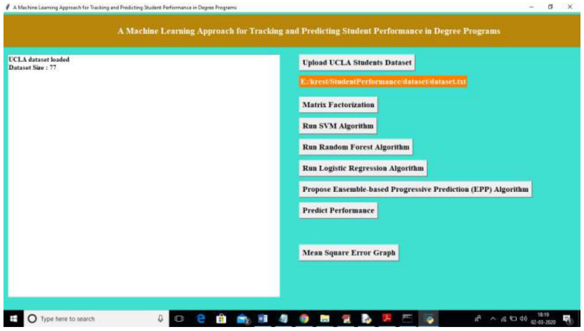
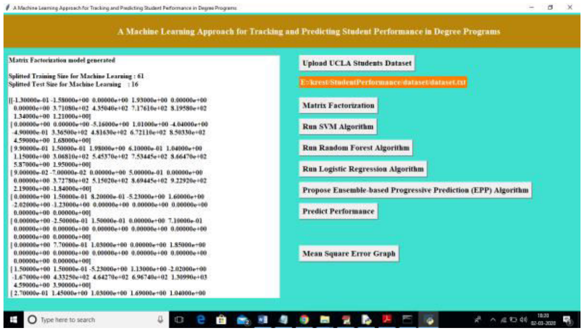

A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

Fig.6: Home screen


A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

```
UCLA dataset loaded
Dataset Size : 77
```

Fig.7: Dataset loading

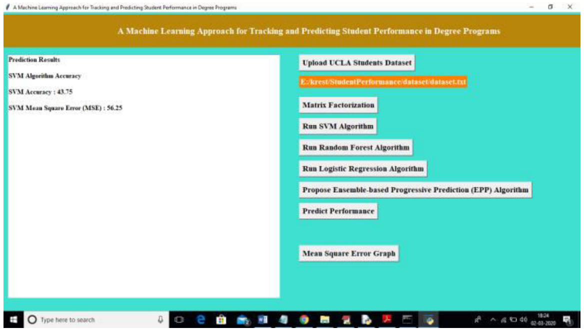

A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

```
Matrix Factorization model generated

Splitted Training Size for Machine Learning : 61
Splitted Test Size for Machine Learning    : 16

[[-1.30000e-01 -1.58000e+00  0.00000e+00  1.93000e+00  0.00000e+00
   0.00000e+00  3.71080e+02  4.35040e+02  7.17610e+02  8.19580e+02
   1.34000e+00  1.21000e+00]
 [ 0.00000e+00  0.00000e+00 -5.16000e+00  1.01000e+00 -4.04000e+00
  -4.90000e-01  3.36500e+02  4.81630e+02  6.72110e+02  8.50330e+02
   4.59000e+00  1.68000e+00]
 [ 9.90000e-01  1.50000e-01  1.98000e+00  6.10000e-01  1.04000e+00
   1.15000e+00  3.06810e+02  5.45370e+02  7.53445e+02  8.66470e+02
   5.87000e+00  1.95000e+00]
 [ 9.00000e-02 -7.00000e-02  0.00000e+00  5.00000e-01  0.00000e+00
   0.00000e+00  3.72780e+02  5.15020e+02  8.69445e+02  9.22920e+02
   2.19000e+00 -1.84000e+00]
 [ 0.00000e+00  1.50000e-01  8.20000e-01 -5.23000e+00  1.60000e+00
  -2.02000e+00 -1.23000e+00  0.00000e+00  0.00000e+00  0.00000e+00
   0.00000e+00  0.00000e+00]
 [ 0.00000e+00 -2.50000e-01  1.50000e-01  0.00000e+00  7.10000e-01
   0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00
   0.00000e+00  0.00000e+00]
 [ 0.00000e+00  7.70000e-01  1.03000e+00  0.00000e+00  1.85000e+00
   0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00
   0.00000e+00  0.00000e+00]
 [ 1.50000e+00  1.50000e-01 -5.23000e+00  1.13000e+00 -2.02000e+00
  -1.67000e+00  4.33250e+02  4.64270e+02  6.96740e+02  1.30990e+03
   4.59000e+00  3.90000e+00]
 [ 2.70000e-01  1.45000e+00  1.03000e+00  1.69000e+00  1.04000e+00
```

Fig.8: Matrix factorization


A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

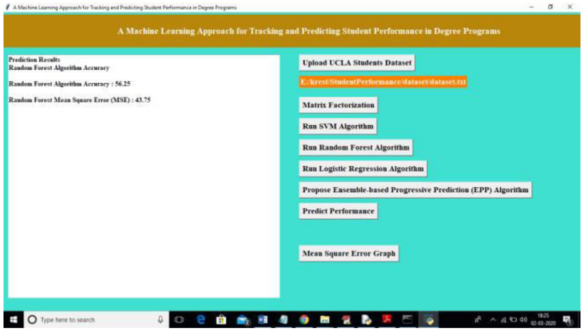
```
Prediction Results

SVM Algorithm Accuracy

SVM Accuracy : 43.75

SVM Mean Square Error (MSE) : 56.25
```

Fig.9: SVM algorithm

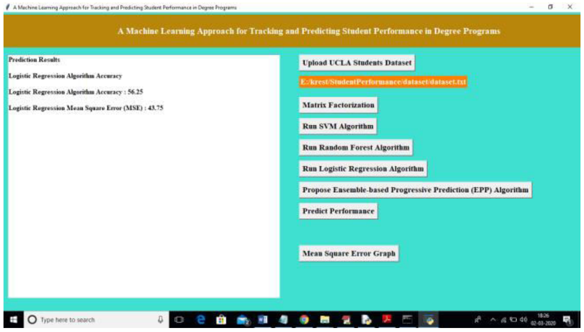

A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

```
Prediction Results
Random Forest Algorithm Accuracy

Random Forest Algorithm Accuracy : 56.25

Random Forest Mean Square Error (MSE) : 43.75
```

Fig.10: Random forest algorithm


A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

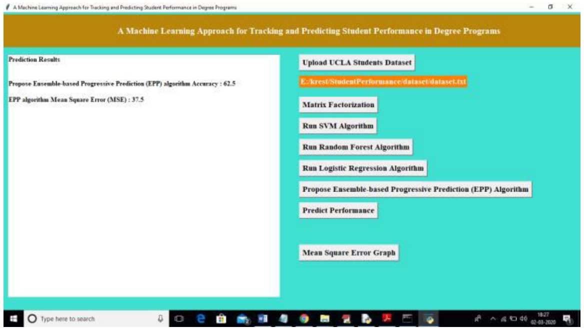
```
Prediction Results

Logistic Regression Algorithm Accuracy

Logistic Regression Algorithm Accuracy : 56.25

Logistic Regression Mean Square Error (MSE) : 43.75
```

Fig.11: Logistic regression algorithm

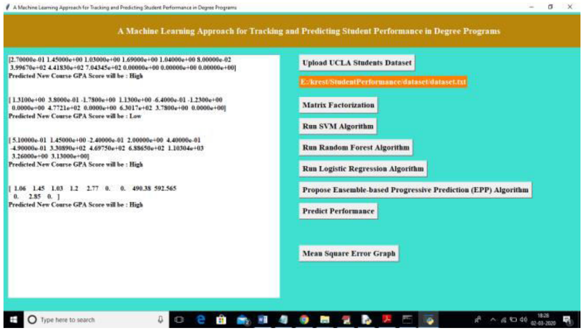

A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search

```
Prediction Results


Propose Ensemble-based Progressive Prediction (EPP) algorithm Accuracy : 62.5

EPP algorithm Mean Square Error (MSE) : 37.5
```

Fig.12: Propose EPP algorithm


A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Upload UCLA Students Dataset
E:/krest/StudentPerformance/dataset/dataset.txt
Matrix Factorization
Run SVM Algorithm
Run Random Forest Algorithm
Run Logistic Regression Algorithm
Propose Ensemble-based Progressive Prediction (EPP) Algorithm
Predict Performance
Mean Square Error Graph
Type here to search
18:28
02-03-2020

```
[2.70000e-01 1.45000e+00 1.03000e+00 1.69000e+00 1.04000e+00 8.00000e-02
 3.99670e+02 4.41830e+02 7.04345e+02 0.00000e+00 0.00000e+00 0.00000e+00]
Predicted New Course GPA Score will be : High


[ 1.3100e+00  3.8000e-01 -1.7800e+00  1.1300e+00 -6.4000e-01 -1.2300e+00
  0.0000e+00  4.7721e+02  0.0000e+00  6.3017e+02  3.7800e+00  0.0000e+00]
Predicted New Course GPA Score will be : Low


[ 5.10000e-01  1.45000e+00 -2.40000e-01  2.00000e+00  4.40000e-01
 -4.90000e-01  3.30890e+02  4.69750e+02  6.88650e+02  1.10304e+03
  3.26000e+00  3.13000e+00]
Predicted New Course GPA Score will be : High


[ 1.06    1.45    1.03    1.2     2.77    0.      0.    490.38  592.565
  0.      2.85    0.  ]
Predicted New Course GPA Score will be : High
```

Fig.13: Predict performance

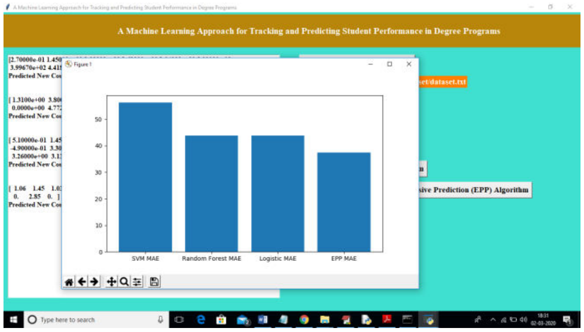

A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
A Machine Learning Approach for Tracking and Predicting Student Performance in Degree Programs
Figure 1
0
10
20
30
40
50
SVM MAE
Random Forest MAE
Logistic MAE
EPP MAE
set/dataset.txt
sive Prediction (EPP) Algorithm
Type here to search
18:31
02-03-2020

Fig.14: Mean square error graph

## 5. CONCLUSION

In this paper, we proposed a novel method for predicting students' future performance in degree programs given their current and past performance. A latent factor model-based course clustering method was developed to discover relevant courses for constructing base predictors. An ensemble-based progressive prediction architecture was developed to incorporate students' evolving performance into the prediction. These data-driven methods can be used in conjunction with other pedagogical methods for evaluating students' performance and provide valuable information for academic advisors to recommend subsequent courses to students and carry out pedagogical intervention measures if necessary. Additionally, this work will also impact curriculum design in degree programs and education policy design in general.

## 7. FUTURE SCOPE

Future work includes extending the performance prediction to elective courses and using the prediction results to recommend courses to students.

## REFERENCES

[1] The White House, "Making college affordable," https:// www.whitehouse.gov/issues/education/higher-education/ making-college-affordable, 2016.

[2] Complete College America, "Four-year myth: Making college more affordable," http://completecollege.org/wp-content/uploads/2014/ 11/4-Year-Myth.pdf, 2014.

[3] H. Cen, K. Koedinger, and B. Junker, "Learning factors analysis–a general method for cognitive model evaluation and improvement," in International Conference on Intelligent Tutoring Systems. Springer, 2006, pp. 164–175.

[4] M. Feng, N. Heffernan, and K. Koedinger, "Addressing the assessment challenge with an online system that tutors as it assesses," User Modeling and User-Adapted Interaction, vol. 19, no. 3, pp. 243–266, 2009.

[5] H.-F. Yu, H.-Y. Lo, H.-P. Hsieh, J.-K. Lou, T. G. McKenzie, J.-W. Chou, P.-H. Chung, C.-H. Ho, C.-F. Chang, Y.-H. Wei et al., "Feature engineering and classifier ensemble for kdd cup 2010," in Proceedings of the KDD Cup 2010 Workshop, 2010, pp. 1–16.

[6] Z. A. Pardos and N. T. Heffernan, "Using hmms and bagged decision trees to leverage rich features of

user and skill from an intelligent tutoring system dataset," Journal of Machine Learning Research W & CP, 2010.

[7] Y. Meier, J. Xu, O. Atan, and M. van der Schaar, "Personalized grade prediction: A data mining approach," in Data Mining (ICDM), 2015 IEEE International Conference on. IEEE, 2015, pp. 907–912.

[8] C. G. Brinton and M. Chiang, "Mooc performance prediction via clickstream data and social learning networks," in 2015 IEEE Conference on Computer Communications (INFOCOM). IEEE, 2015, pp. 2299– 2307.

[9] KDD Cup, "Educational data minding challenge," https://pslcdatashop. web.cmu.edu/KDDCup/, 2010.

[10] Y. Jiang, R. S. Baker, L. Paquette, M. San Pedro, and N. T. Heffernan, "Learning, moment-by-moment and over the long term," in International Conference on Artificial Intelligence in Education. Springer, 2015, pp. 654–657.