**Title: Understanding the Factors that Shape Academic Success: A Study on the User knowledge modeling dataset**

**Student ID: s4000592**

**Student Name and email (contact info): Name: Sankalp Dangwal**

**Affiliations: RMIT University.**

**Date of Report: 24-05-2023**

**I certify that this is all my own original work. If I took any parts from elsewhere, then they were non-essential parts of the assignment, and they are clearly attributed in my submission. I will show I agree to this honor code by typing "Yes": Yes.**

**Table of Contents**

[ABSTRACT 3](#_TOC_250006)

[INTRODUCTION 3](#_TOC_250005)

[METHODOLOGY 4](#_TOC_250004)

[RESULTS 7](#_TOC_250003)

[DISCUSSION 8](#_TOC_250002)

[CONCLUSION 10](#_TOC_250001)

[REFERENCES 10](#_TOC_250000)

# ABSTRACT

This study seeks to determine what distinguishes students who are categorized as high-level, middle-level, low-level, and extremely low-level knowledge users by analyzing the behavior of various students. Students' information about study time, repetition, and exam performance was collected, and materials for related objects. We employ the "K nearest neighbors’ classifier with parameter tuning and hill climbing" in this study, which examines several attributes and only chooses those that produce higher predicted results. To analyze user data and more accurately anticipate which user belongs to which group, "decision tree" categorization is also utilized. Overall, it was clear that a student's "exam performance of goal" determined which category of knowledge consumers they would belong to. With the help of the models mentioned above, educators can identify students who need extra attention and support, as well as provide insights into the factors that contribute to academic success. We are trying to find what makes a user high-level and how can those at the lower level.

improve their performance. The models discussed above can assist instructors identify students who require additional attention and support as well as give them insights into the elements that contribute to academic performance. What makes a high-level user? How can users at lower levels do better? These are the questions we are attempting to answer.

# INTRODUCTION

Before the introduction of online educational platforms, it was not clear how various users would interact with their study material. The introduction of online educational platforms enabled educators to observe and track the study patterns, time durations, repetition of goals, and related objects of various users. The goal of a user modelling system is to provide enough or suitable knowledge for students/users. In this process, user models are used for content personalization, accessing the data easily and fast, and streamlining the applications more effectively and efficiently. The user models have been mostly created in web-based applications, especially for online learning environments. Students’ models are composed of static and dynamic data in a web-based adaptive learning environment. Static data represents information about the students such as username, password, and age. Dynamic data represents the knowledge of students about domain-dependent data. Thus, the dynamic data in the user model might be also called the user knowledge model user modelling system (UMS) is used to obtain the behaviour or data of students from their interaction with web applications. This report dwells on understanding patterns in the behaviour of students belonging to different knowledge levels.

# METHODOLOGY

The user knowledge was first collected by H. T. Kahraman and Sagiroglu, S., Colak to develop  
intuitive knowledge classifiers and modelling of users' domain-dependent data in web  
Knowledge-Based Systems Our research uses KNN and Decision tree classifications to  
understand and evaluate the classification of students in different knowledge categories. The user data was obtained from the UCI machine learning repository which contained two datasets about user knowledge classification. The first contained training data composed of 258 user records and the second contained testing data composed of 145 user records. The dataset comprised the following columns:

* STG (The degree of study time for goal object materials)
* SCG (The degree of repetition number of a user for goal object materials)
* STR (The degree of study time of a user for related objects with goal object)
* LPR (The exam performance of a user for related objects with goal object)
* PEG (The exam performance of a user for goal objects)
* UNS (The knowledge level of a user)

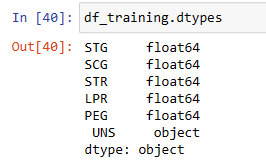
For the **cleaning process**, each column in the dataset went through the following checks.

* **Null value attributes:** I checked the data for possible null value attributes, but none were found.

A screenshot of a computer

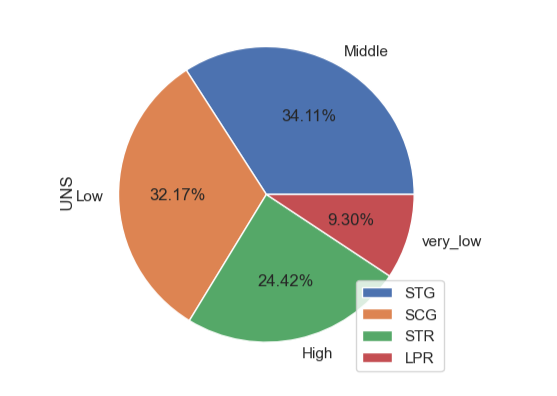
Description automatically generated with medium confidence

* **Data type:** I ensured that all the columns had the same datatype except for one that is UNS.



* **Invalid values:** I checked the data for invalid values using the describe () method to ensure the maximum and minimum values for each column were well within the specified range. Also stripped the textual columns of any leading or trailing white spaces, for dealing whith white space that is in column UNS I renamed the column instead of removing the space and checked if any invalid values are present in the target column.
* **Check for duplicate values:** I have checked duplicate values and there were none.

I now tried to find any patterns in the data given to us. Patterns that will help us classify the students. From the given graph below, it is quite visible that PEG(The exam performance of a user for goal objects) is a clear indicator of a student's knowledge level as it separates the users distinctively. But we know that exam performance is not always a good indicator of a student's knowledge level.



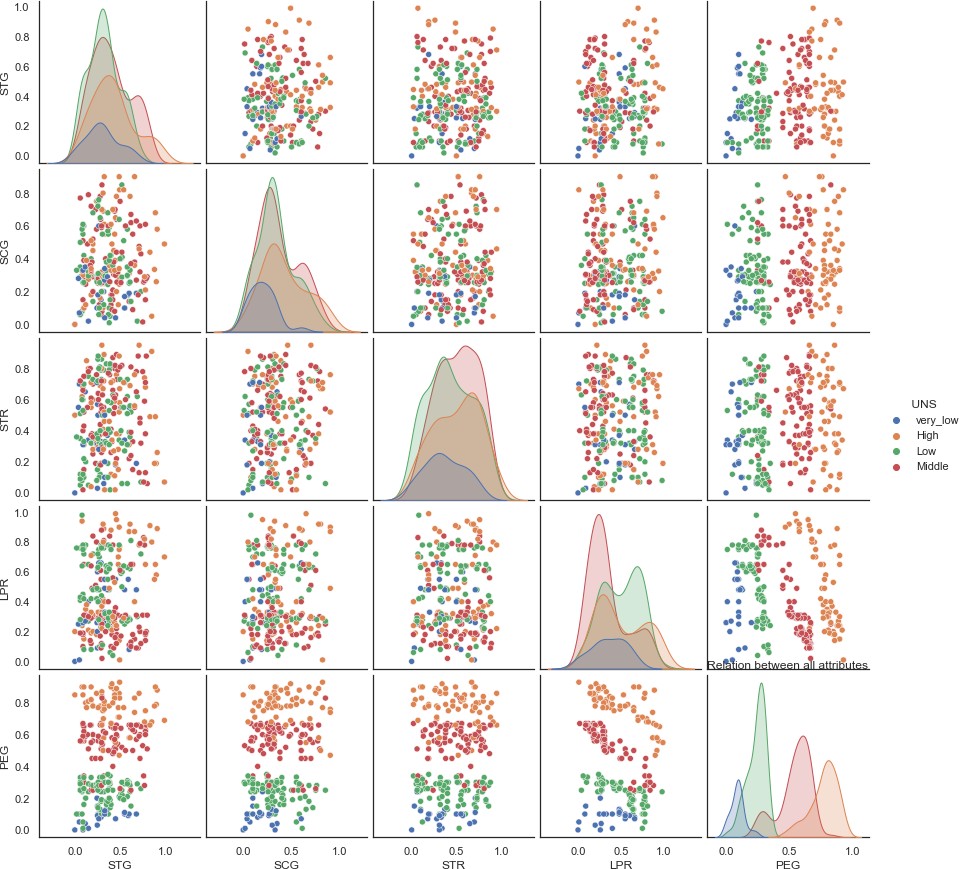


Fig 2.2

To see the disparity or similarity in the users we tried checking using box plots.

UNS VS STG

A picture containing diagram, rectangle, square, screenshot

Description automatically generated

UNS VS STR

A picture containing diagram, rectangle, square, screenshot

Description automatically generated

UNS VS LPR

A picture containing diagram, rectangle, square

Description automatically generated

UNS VS SCG

A picture containing diagram, rectangle, square, screenshot

Description automatically generated

Fig 2.3

* First thing that we notice in all these graphs is that most of the High-level users are on the higher end of the spectrum in terms of study time, repetition, and the other two aspects.
* We then used two classification models, K nearest neighbors and decision tree, and the first thing we notice in all of these graphs is that most High-level users are on the higher end of the spectrum in terms of study time, repetition, and the other two aspects. To get the best value of K for the given analysis, I implemented a basic for loop for values of K and accessed the accuracy of the model for each value of K. The K value for the model that provided the h score was the model that required the most study time. The model that was generated gave an accuracy score of **0.86**. To better understand the impact of each feature, I used the parameter tuning and hill climbing technique for K-nearest neighbors. The pseudo-code for the technique that I have used is given below. When conducting this analysis, the data was already divided into testing data and training data. The model was trained on the testing data and checked against the training data.

1. Following actions must be taken for each iteration of the test-train split.

2. Make the initial state the current state by evaluating the initial first characteristic.

3. Continue to improve the current state indefinitely until no more features can be used to do so. Choose a feature that hasn't yet been applied to the current state and apply it to create a new state.

b) Carry out these actions to assess the new state.

i. If the current state is a goal state, then stop and return to success.

ii. If it is an improvement over the situation as it is, make it the situation and go on.

Continue in the loop until a solution is found if step three is not an improvement over the existing situation.

4. Print the model accuracy, the test-train split percentage, and the optimum feature selection.

5. Exit.

The target value set is not evenly distributed, so using 'Entropy' as the classifier criterion for the decision tree algorithm will be efficient. The minimum leaf node sample size has been set to 4, and the Max depth has been set to 7, to prevent overfitting the data. Analysing the generated decision tree has the potential to provide insights into what feature differentiates the users.

**RESULTS**

On training the KNN model based solely on the different values of K to get the best number of neighbors, we attained the value of **K = 1 and an accuracy score of 0.**95 over already split data. To further validate and analyze the accuracy of the model over a large enough dataset, I combined the training and testing data and then split the entire dataset into random training and testing datasets 20 times and gauged the accuracy of the model for the same value of K that was previously generated. The accuracy of the model for different train test splits of the data set was as follows:

A screenshot of a computer

Description automatically generated with medium confidence

As we can see, the score of the model varied for each train test split, and upon taking the average of those scores, we got an average accuracy of the model for the value of **K = 1 as 0.95**.

The next thing to analyze was which feature along with PEG had a significant impact on the accuracy of the model. To understand this using KNN, we used the method of parameter tuning and hill climbing function. Now the question comes as to what size of data should be used to train the model and what size should be used to test it. To find the optimal size, the model was run on multiple different sizes of train test splits. PEG and LPR were the two features that were selected in almost all of the test instances indicating that exam results are the prime indicator and classifier when it comes to user knowledge groups. One other feature that appears in many test instances is STG. Indicating the degree of study time is also an important classifier.

* Model prediction for KNN with parameter tuning

The next classifier we use is the decision tree classifier. For this classifier, we gave the entire feature set with a minimum leaf node size of 3. The training and testing data were combined to create one whole user data on which the model was executed. The data was split into training and testing data like the hill climbing technique for KNN. The accuracy of the model for each testing size was noted. The accuracy is given below:

A screenshot of a computer

Description automatically generated with medium confidence

* We can see that for the test size 0.3, we get the maximum accuracy of 0.90. The general split of testing and training data lies around 20-30%. Which is, 70-80% of the data will be used to train the model and the remaining 20-30% will be used to test the model.

Upon getting the accuracies of both the models, KNN with parameter tuning and Decision tree over different test train splits. We plot a line graph to compare their accuracies.

Throughout all different sets, KNN seems to be having better accuracy when compared to decision trees when classifying the data. But for the recommended test train split size (0.3), they seem to have the same accuracy level of around 0.90.

* For both models, we first found the parameters that provided the local best result. Then, for these said parameters we used K-fold cross-validation to validate the accuracies of the models.

This confirms the statement that they both have the same accuracy of around 0.90 while predicting the knowledge level of the students.

# DISCUSSION

The following findings are based on the analysis of the generated graphs and data models:

1. We can see that there are many outliers when it comes to high-level users, indicating that some members of the high-performance group study for much longer durations than the average high-performance user to be. When we look at the STG against UNS box plot, it appears that all students in all groups study for approximately the same amount of time.
2. Although a single outlier might be an exception, we will not be taking that into account during our analysis. It is quite evident when we look at the box plot of SCG against UNS that most of the users that belong to the very-low level don't seem to be repeating what they study. This suggests that perhaps increasing the number of times a user studies their given study material might result in them improving their scores and promoting themselves to a higher group.
3. When we look at the box plot of STR against UNS, and the box plot of LPR against UNS, we can observe that all user groups put in the same amount of effort to study related objects with the goal object. But there is not much difference in the exam performance of all the users belonging to the different groups. This says that the exam performance of users for objects related to the goal object does not define the knowledge level of the user by itself.
4. Since the low group has a relatively high average LPR, those in the low group may be promoted to the medium or even High intelligence group if they put in more effort. If we compare the box plots of all the features for those who belong to the low and middle knowledge groups, it's obvious that those in the low group on average, put less effort in terms of study time and repetition of study material when compared to those belonging to the middle group.
5. When we compare the box plots for all the features for the groups with moderate and high knowledge, we can see that they are different. The group with moderate knowledge puts up a same amount of effort as the group with high knowledge. The fact that the middle group performs poorly on the exam, however, suggests that certain users may exert less effort than other users to perform better.

The graph that the decision tree generates with a test train split of 0.3, is given below



The next classifier that the decision tree uses is SCG, indicating that it considers the degree of repetition of study material as an important classifier. This demonstrates that for a given value of PEG and LPR, if your SCG is lower than a given threshold, the decision tree gets a good idea as to whether you will perform well on the exam for goal objects and related goal objects.This is contrary to the results obtained from using KNN where STG(degree of study) is the next best feature to be considered when classifying the user data.

# CONCLUSION

* The purpose of this study was to determine what characteristics characterize high-level users and how users at lower levels can perform more effectively. The two training models we employed—KNN and decision tree—showed us that, in addition to a user's exam performance, the length of time spent studying and the number of times he repeats the materials for the goal object also affect the user's knowledge level. Even if they spend a lot of time reviewing the information, some users are not High-level users. These students have a desire and need to comprehend an idea. Giving them guidance and special attention should result in these students performing better in exams and promoting themselves to high-level users. Some users have a high enough STG and yet do not have that high value of LPR and PEG. This could be caused due to factors that are out of the scope of this research. Having these trained models, and live data of the current set of students such as how long they spend time on the online learning platform, how many times they repeat the given materials and their exam results, the teachers can classify the students into the knowledge groups and then based on that, focus particularly more on those students that need more attention. These students are those that have low and middle levels of understanding. The users that belong to the very low level need to increase their study duration and number of repetitions at first to improve their level of knowledge.

# REFERENCES

1. H. T. Kahraman, Sagiroglu, S., Colak, I.(2013), Developing intuitive knowledge classifier and modeling of users' domain dependent data in web, Knowledge Based Systems, vol. 37, pp. 283-295,
2. Dr. Yongli Ren,(2023) Practical Data Science: Feature Selection (Hill Climbing) Computer Science & IT School of Science, week6-featureSelection.pdf