

Intelligent Methods and Models for Assessing Level of Student Adaptation to Online Learning

Vasyl Teslyuk, Anastasiya Doroshenko, Dmytro Savchuk

Lviv Polytechnic National University, S. Bandera str. 12, Lviv, 79013, Ukraine

Abstract

The problem of introducing online learning is becoming more and more popular in our society. Due to COVID-19 and the war in Ukraine, there is an urgent need for the transition of educational institutions to online learning, so this paper will help people not make mistakes in the process and afterward. The paper's primary purpose is to investigate the effectiveness of machine learning tools that can solve the problem of assessing student adaptation to online learning. These tools include intelligent methods and models, such as classification techniques and neural networks. This work uses data from an online survey of students at different levels: school, college, and university. The survey consists of questions such as gender, age, level of education, whether the student is in the city, class duration, quality of Internet connection, government/non-government educational institution, availability of virtual learning environment, whether the student is familiar with IT, financial conditions, type of Internet connection, a device used for studying, etc. To obtain the results on the effectiveness of online education were used the following machine learning algorithms and models: Random Forest (RF), Extra Trees (ET), Extreme, Light, and Simple Gradient Boosting (GB), Decision Trees (DT), K-neighbors (K-mean), Logistic Regression (LR), Support Vector Machine (SVM), Naïve Bayes (NB) classifier and others. An intelligent neural network model (NNM) was built to address the main issue.

Keywords

Prediction, classification, data, dataset, datastore, feature importance, Random Forest, SMV, Logistic Regression, Extra Trees, Gradient Boosting, Neural Network, model.

1. Introduction

Technological progress does not stand. Still, each professional field is gradually developing, and online learning has not been spared either. Therefore, the first appearance of COVID-19 in the world and in Ukraine significantly pushed this direction. Due to this, traditional education has undergone significant changes over the past few years. There was a transition from simple paper and pen sheets to online forms and a keyboard. Now instead of listening to the teacher directly in the classroom, people do it in a Google classroom.

Usually, the transition to online learning is the only possible option. Before COVID-19, students were not sufficiently aware of online education, but after its sudden appearance, they are still trying to adapt. However, the process and its results are not always good, although the majority succeeded, confirming the quality of online learning. The transition can depend on many factors, namely: the level of education of the student, the presence of urbanization around the student, the length of online classes, the presence of a virtual learning environment (LMS), financial conditions, the type of Internet connection or device on which the student is studying. The available dataset used in the work shows that only 8.3% of students can adapt well to online learning, and the remaining 91.7% have a medium or low level of adaptation [1].

COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine
EMAIL: vasyi.m.teslyuk@lpnu.ua (V. Teslyuk); anastasia.doroshenko@gmail.com (A. Doroshenko); savchukd7@gmail.com (D. Savchuk)
ORCID: 0000-0002-5974-9310 (V. Teslyuk); 0000-0002-7214-5108 (A. Doroshenko); 0009-0006-0937-6161 (D. Savchuk);



© 2023 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

Lack of a permanent internet connection, digital skills, electricity problems, poor network connectivity, proper tools, and instructions are the main obstacles to online education during the COVID-19 pandemic or other events that may challenge the very existence of the state and its society.

Therefore, this paper was carried out and researched to predict the level of adaptation to the online education system.

Since the object of the research is data mining processes that solve classification problems, it is necessary to find out what classification is, how it is performed, and by what methods.

Usually, there are only two forms of data analysis with which you can get models that will describe classes or predict the future data trend. These two types are as follows:

- Classification;
- Prediction.

Researchers use classification and prediction to obtain models that predict the future data trend and how it will spread and change.

Thus, classification models predict categorical labels of classes, while predictive models predict continuous functions (numerical values). Sometimes, the number of labels can exceed two (as in our case), which means that we are dealing with non-binary (multi-class) classification. The main essence of classification is to determine the category or class label of a new value (observation).

Therefore, to start building a model, it is necessary to determine whether using the existing data set to create a classification model is appropriate. First, it is essential to establish whether there is enough data, i.e., its size (number of observations), and then whether the description of these observations is sufficient (number of independent characteristics).

Sometimes with a sufficiently small number of observations and, in turn, with an insufficient number of independent characteristics, obtaining good accuracy and quality of the model can be challenging.

The next step is to choose the classification model itself. As a result, the following types were used in work: RF, ET, XGBOOST, LIGHTGBM and GB, DT, K-mean, LR, SVM, NB, and NNM. Finally, the classification results have been compared. This is done by drawing the following types of charts: ROC curve, scatter matrix, distribution of features, their importance, and others - which in turn will help to present the results more clearly.

2. Data Preparing

2.1. Initial Dataset

The first step is to look at the original data set. We need to look at how many observations it contains, what characteristics it has, and its desired variable.

In our case, the dataset was collected after a survey of students. It consisted of various questions, such as gender, age, level of education, whether they live in the city, duration of the lecture, quality of the Internet connection, type of educational institution, availability of LMS, whether the student is familiar with IT, financial conditions, type of Internet or device they use, and their level of adaptation. The total number of students surveyed is 1205, and the number of observations in the dataset also equals this number [1].

Detailed description of the labels of the target class is given below in Table 1:

Table 1
Labels of the Target Class

<i>N_o</i>	<i>Details</i>	<i>Feature</i>	<i>Value</i>
Y ₁	Low Adaptation		0
Y ₂	Moderate Adaptation		1
Y ₃	Hight Adaptation		2

Now let's look at the independent characteristics of our dataset in more detail Table 2:

Table 2
Initial Dataset

<i>N_i</i>	<i>Details</i>	<i>Feature</i>	<i>Value</i>
X ₁	Gender		Girl, Boy
X ₂	Age of the student		From 1 to 5, 6 – 10, 11 – 15, 16 – 20, 21 – 25, 26 – 30, 30+
X ₃	Institution level		School, Collage, University
X ₄	Institution type		Non-Government, Government
X ₅	IT student		No, Yes
X ₆	Location in City		No, Yes
X ₇	Load shedding		Low, Hight
X ₈	Student's financial conditions		Poor, Mid, Rich
X ₉	Internet type		2G, 3G, 4G
X ₁₀	Device type		Tab, Mobile, Computer
X ₁₁	Network connection type		Mobile Data, Wifi
X ₁₂	Class duration (daily)		0, from 1 to 3, from 3 to 6
X ₁₃	Learning management system availability		No, Yes

2.2. Data Processing

The next step is to create graphs that show how variables are distributed in the dataset (Fig. 1 and Fig. 2). This can be achieved by building a histogram that shows the number of unique characteristics for each independent variable.

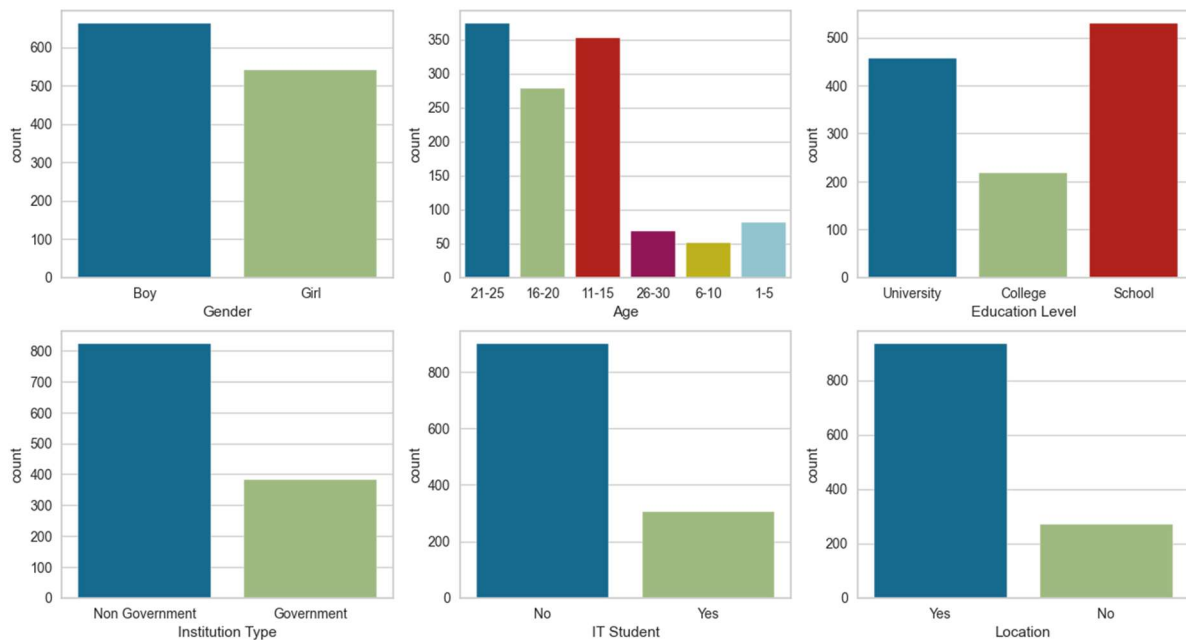


Figure 1: Features' distribution of the dataset

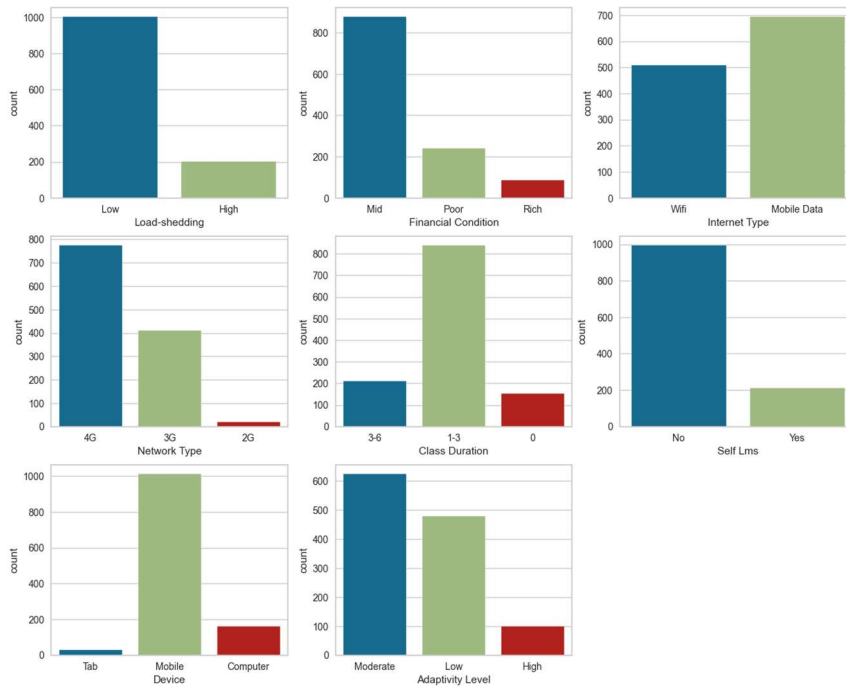


Figure 2: Features' distribution of the dataset (continuation)

Now it is advisable to plot the distribution of characteristics depending on the target class. Therefore, below from Fig. 3. and Fig. 4., we can see the distribution of characteristics values depending on the level of students' adaptation (low, medium, high). These dependencies are built for such characteristics as age, gender, level of education, type of institute, IT studying, student's location, load shedding, economic conditions, type of Internet and network, length of classes, availability of LMS, and type of device the student is studying on.

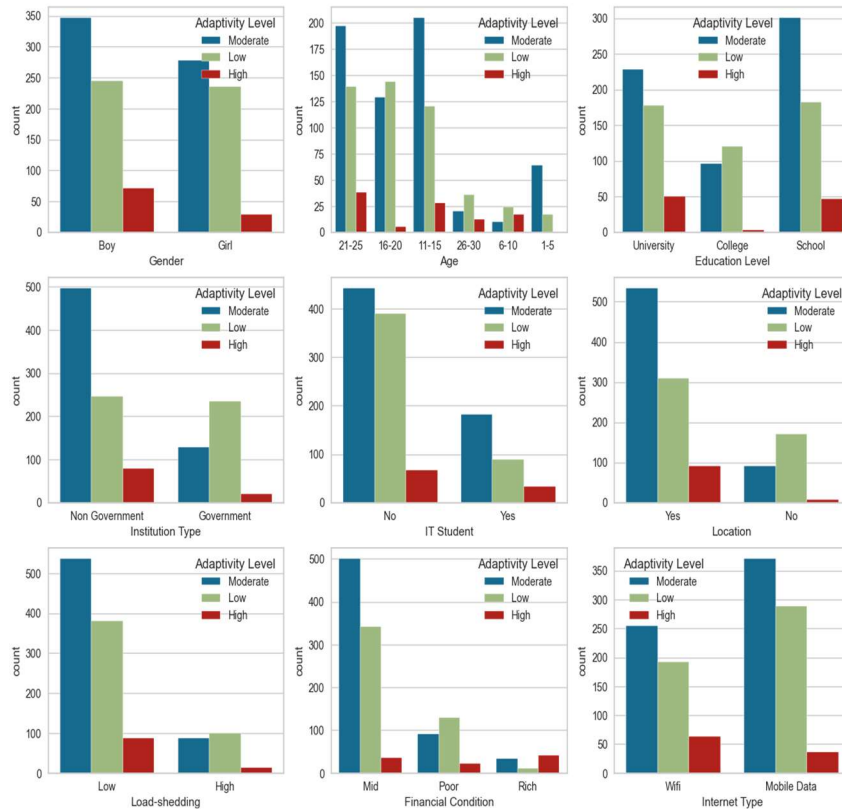


Figure 3: Features' distribution depending on the level of adaptation

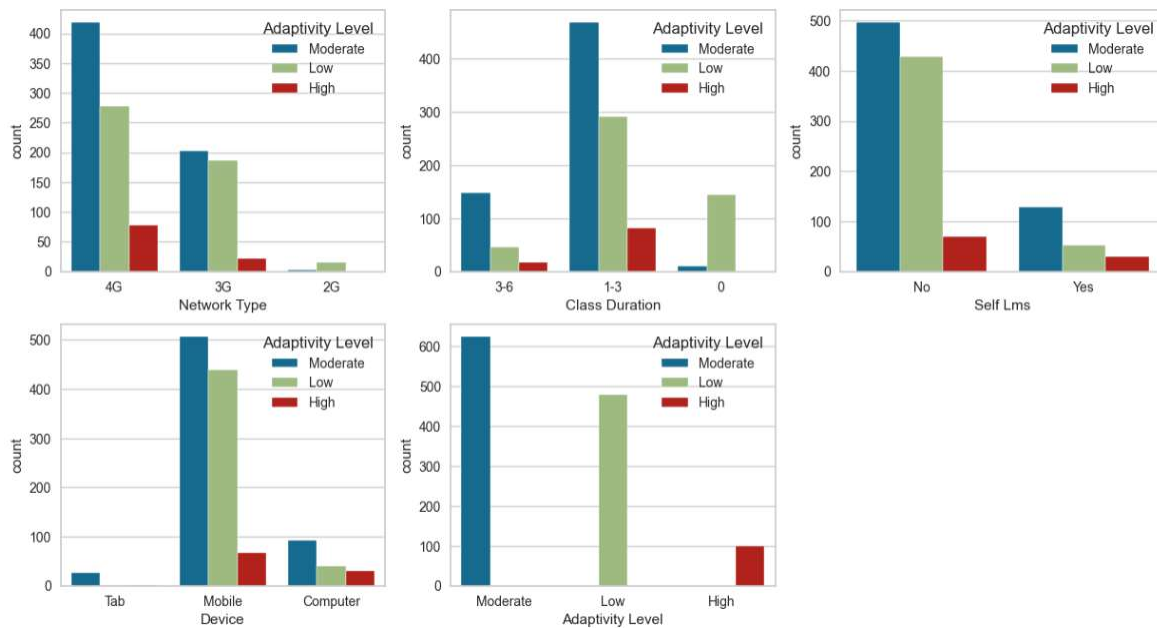


Figure 4: Features' distribution depending on the level of adaptation (continuation)

The following conclusions can be drawn from these diagrams (Fig. 3 and Fig. 4):

- The level of students' adaptation does not depend on their gender, as the distribution is almost equal for women and men.
- Students aged 21-25, 11-15, and 1-5 are the best adapted, and the worst are those aged 26-30, 16-20, and 6-10.
- Students of universities and schools are better adapted to online education, especially when they are non-government-type.
- Studying students in IT does not increase the effectiveness of online learning.
- A student's stay in the city improves the quality of online learning.
- A high load-shedding negatively affects students' adaptation.
- Wealthier students generally adapt better than other students.
- The type of Internet connection and network have a minor impact on student adjustment.
- The best performers are students with 1 to 3 hours of daily classes.
- Other characteristics have almost no impact on the quality of online education.

2.3. Target Class Research

After all the data manipulations have been performed, you can also analyze the labels of the target class. As mentioned earlier, the level of student adaptation to online learning can be of three types:

- Low;
- Moderate;
- High.

From Fig. 5. we can understand that the distribution of students' adaptability to online learning is as follows:

- Low adaptation is about 39.8% of all observations in the dataset, which is 408 values.
- Moderate adaptation equals 51.9%, which is $1025 * 0.519 = 532$ observations.
- High adaptation is present in 8.3% of students. In turn, it is only 85 students out of the total number.

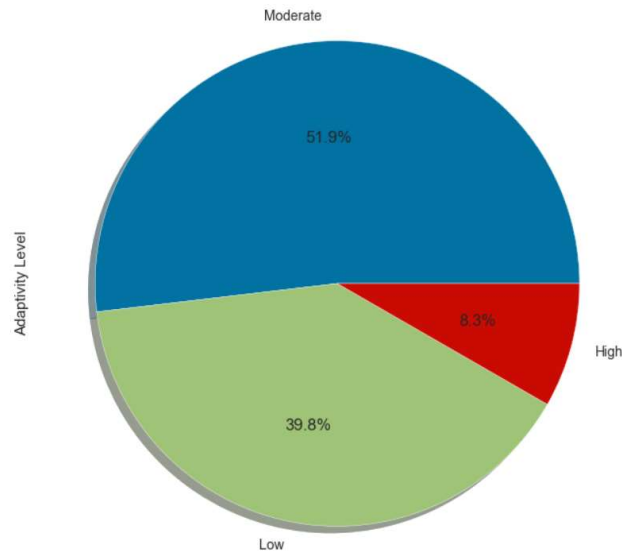


Figure 5: Distribution of the level of adaptation in the dataset

2.4. Correlation Matrix

It is also necessary to build a correlation matrix to understand how the characteristics depend on each other. The result of building this type of diagram is shown below in Fig. 6.

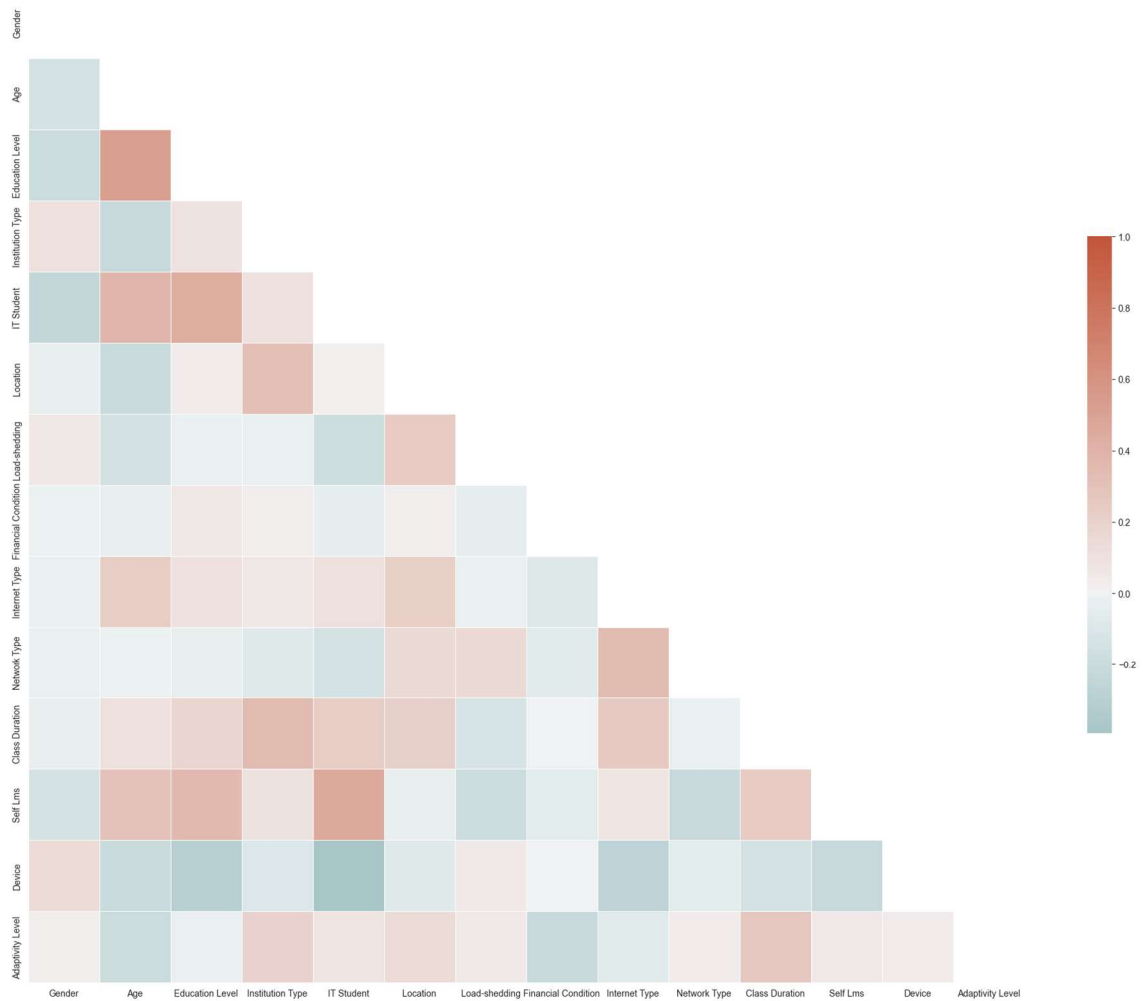


Figure 6: Correlation Matrix

From this chart, we can see that the most dependent characteristics are the **age** and **financial conditions** of the student. However, for others, this relationship is relatively tiny.

This may indicate that no further operations on the dataset's characteristics need to be performed. Such operations may include principal component analysis (PCI) and different types of normalization or regularization (L1 or L2).

3. Model Creation and Classification

3.1. Model Types

The following machine learning algorithms and models were used to obtain results on the effectiveness of online education implementation [3]: Decision Tree, Random Forest, Extra Tree, extreme, light, and normal Gradient Boosting, K-Neighbors, Logistic Regression, Ada Boosting, Support Vector Machine, Naïve Bayesian Classifier, Ridge, Dummy, Linear and Quadratic Discriminants, and finally a Sequential Neural Network—more details in Table 3.

Table 3
List of Used Models

№	Abbreviation	Classifiers	Reference
		Name	
1	DT	Decision Tree	[4]
2	RF	Random Forest	[5]
3	ET	Extra Trees	[6]
4	GBC	Gradient Boosting	[7]
5	KNN	K-Neighbors	[8]
6	LR	Logistic Regression	[9]
7	Ada	Ada Boosting	[10]
8	SVM	Support Vector Machine	[11]
9	NB	Naïve Bayes	[12]
10	Ridge	Ridge Classifier	[13]
11	Dummy	Dummy Classifier	[14]
12	LD	Linear Discriminant	[15]
13	QD	Quadratic Discriminant	[16]
14	NNM	Neural Network Model	[17]

3.2. Model Creation

Before starting to build models, it is necessary to divide the data set into two parts: data for modeling and data that can be used for prediction. As mentioned before, the total number of students surveyed is 1205, so the final split is nine to one, which equals 1084 observations for modeling and 121 for prediction, as shown below in Fig. 7.

```

1 model_data = class_df.sample(frac=0.9, random_state=786)
2 data_unseen = class_df.drop(model_data.index)
3
4 model_data.reset_index(drop=True, inplace=True)
5 data_unseen.reset_index(drop=True, inplace=True)
6
7 print('Data for Modeling: ' + str(model_data.shape))
8 print('Data For Predictions: ' + str(data_unseen.shape))

```

```

Data for Modeling: (1084, 14)
Data For Predictions: (121, 14)

```

Figure 7: Splitting a Dataset into parts

The next step is to split the modeling data into training and test data. This is done in two ways: for the classification and neural network models.

Thus, 70% of all observations were used for training data and only 30% for test data.

3.3. Model with Classifiers

Fig. 8. shows a typical model creation and training process using standard classifiers. As we can see, there were ten runs of the classifier. The average accuracy after these runs is 87%.

```
1 model_xg = create_model('rf')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.8289	0.9449	0.7648	0.8368	0.8316	0.6899	0.6915
1	0.8684	0.9709	0.7922	0.8683	0.8673	0.7556	0.7564
2	0.8947	0.9666	0.7496	0.8894	0.8892	0.8012	0.8035
3	0.8816	0.9749	0.7381	0.8777	0.8760	0.7754	0.7790
4	0.9211	0.9661	0.8413	0.9293	0.9222	0.8562	0.8598
5	0.8816	0.9804	0.8175	0.8822	0.8804	0.7813	0.7833
6	0.9342	0.9559	0.9087	0.9362	0.9341	0.8807	0.8815
7	0.8816	0.9725	0.7302	0.8882	0.8710	0.7766	0.7808
8	0.8800	0.9658	0.7913	0.8867	0.8774	0.7701	0.7759
9	0.8133	0.9256	0.6929	0.8094	0.8090	0.6491	0.6499
Mean	0.8785	0.9624	0.7827	0.8804	0.8758	0.7736	0.7762
Std	0.0347	0.0155	0.0591	0.0358	0.0350	0.0648	0.0651

Figure 8: Building a RandomForest Classifier Model

The main metrics of this classification report include the following:

- Fold;
- Accuracy;
- ROC curve (AUC);
- Recall;
- Precision;
- F1 - score;
- Cohen's Kappa [18];
- Matthew's correlation (MCC) [19];

It should also be noted that a confusion matrix was created for typical classification models [20].

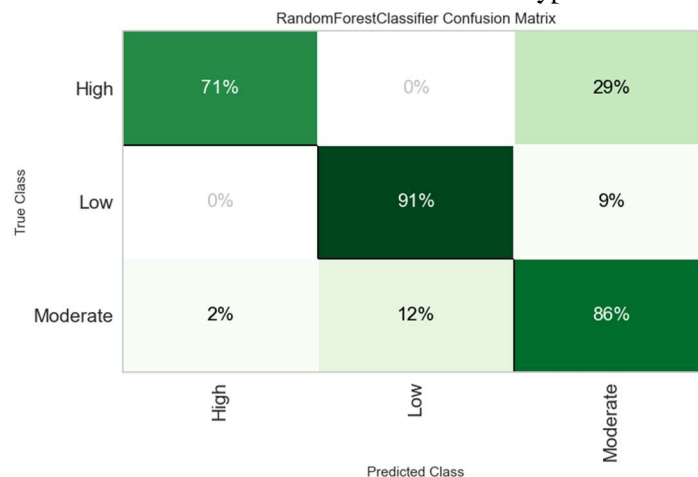


Figure 9: Confusion Matrix of a RandomForest Classifier

From this matrix (Fig. 9.), we can understand the following:

- 71% of the predictions that were labeled "High" were predicted as "High" and the rest as "Medium".
- 91% of the predictions classified as "Low" were classified correctly, and the remaining 9% as "Medium".
- 12% of the "Medium" classes were incorrectly predicted as "Low".

3.4. Neural Network Model

The neural network model used in this paper is shown below in Fig. 10.

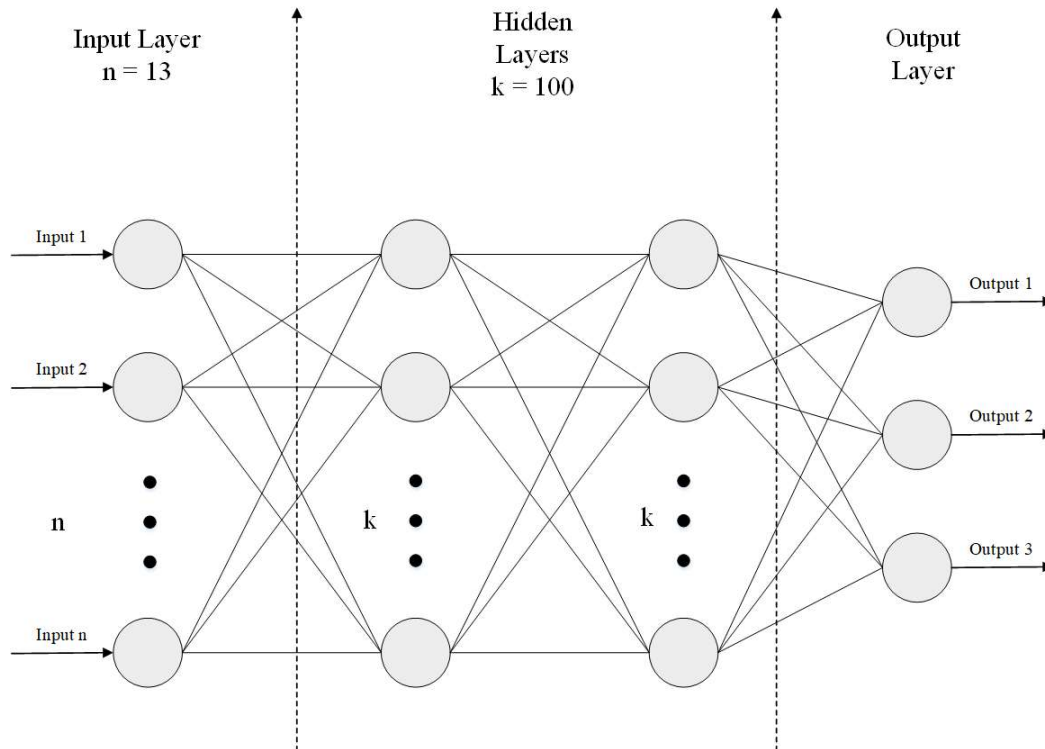


Figure 10: Neural Network Model

From Fig. 10. we can see that the neural network is sequential and has four layers: one input layer with thirteen neurons (for each independent feature), two hidden layers with one hundred neurons each, and one output layer with three neurons, each of which is responsible for the label of the desired class. In particular, it should be noted that training takes place in 50 epochs.

The classification results using a neural network are shown below.

```
1 a, score = model.evaluate(X_test, y_test)
2
3 print("Loss: " + str(a))
4
5 print("Accuracy: " + str(score))
```

✓ 11/11 [=====] - 0s 2ms/step - loss: 0.2856 - accuracy: 0.9080
Loss: 0.2855681777000427
Accuracy: 0.907975435256958

Figure 11: Classification using a neural network

As we can see from the figure above, the average classification accuracy using this neural network is 91%, and the loss amount is about 29%.

4. Model Comparison and Result Analysis

From Fig. 12., we can see that the best classification models were: Random Forest with 87.85% accuracy, Extra Trees with 87.72% accuracy, and Extreme Gradient Boosting with 87.59% accuracy. And the worst models for the given task are Naïve Bayes - 26.91% prediction efficiency, Quadratic Discriminant - 53% accuracy, and Dummy classifier - 55.67%.

Other classification quality metrics provided the same assessment as the ROC curve (AUC), Recall, Precision, F1 value, Cohen's kappa, and Matthew's correlation (MCC).

As a result, we can see that the best model is the self-built neural network, which provided a classification accuracy of 91% (Fig. 11).

```
1 compare_models()
```

Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
rf	0.8785	0.9624	0.7827	0.8804	0.8758	0.7736	0.7762	0.2160
et	0.8772	0.9643	0.7823	0.8798	0.8748	0.7715	0.7740	0.2180
xgboost	0.8759	0.9497	0.7811	0.8787	0.8733	0.7687	0.7717	0.2050
lightgbm	0.8680	0.9455	0.7695	0.8699	0.8653	0.7536	0.7562	0.0700
dt	0.8653	0.9272	0.7741	0.8705	0.8640	0.7511	0.7540	0.0080
gbc	0.8179	0.9143	0.7081	0.8207	0.8109	0.6542	0.6612	0.1250
knn	0.7454	0.8763	0.6268	0.7564	0.7412	0.5254	0.5320	0.0580
lr	0.7256	0.7948	0.5567	0.7125	0.7092	0.4692	0.4786	0.9060
ridge	0.7150	0.0000	0.5319	0.6932	0.6949	0.4433	0.4539	0.0080
ada	0.7123	0.6334	0.5650	0.7020	0.7015	0.4547	0.4594	0.0380
lda	0.7017	0.7945	0.6149	0.7081	0.6965	0.4437	0.4507	0.0100
svm	0.6779	0.0000	0.5511	0.6936	0.6680	0.3979	0.4118	0.0290
dummy	0.5567	0.5000	0.3333	0.3100	0.3982	0.0000	0.0000	0.0120
qda	0.5303	0.5200	0.3563	0.4574	0.4658	0.0450	0.0408	0.0110
nb	0.2691	0.6589	0.4860	0.8100	0.3199	0.1330	0.2430	0.0110

Figure 12: Comparison of the Results of Classification Models

4.1. Permutation Feature Importance of Models

In this paper, we investigated two types of feature importance: for a classification model and a neural network.

Feature Permutation Importance is a tool that helps explain models and describes its behavior in machine learning. This diagram summarizes and evaluates the importance of features [21]. This is done by studying their impact on the classification quality of the created model.

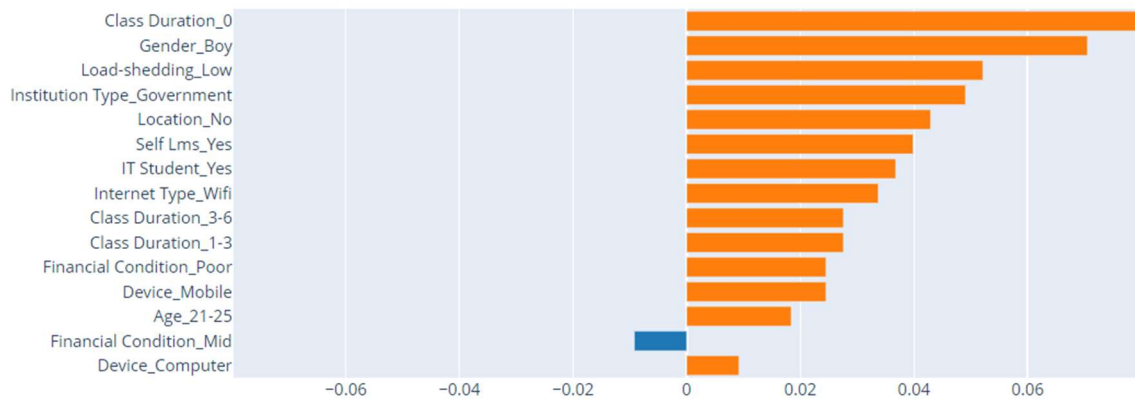


Figure 13: Permutation Feature Importance For Classifiers

The following features of the characteristics can be seen from the bar graph above (Fig. 13).

1. The most important characteristics are the first seven:

- Length of **daytime study** up to one hour.
- **Gender**: Male;
- **Load-shedding**: Low;
- **Type of institution**: Government;
- **Location**: not in the city;
- Presence of **LMS**;
- Belonging to **IT** - yes;

2. Characteristics of lesser importance include:

- Length of **daytime study** up to one hour.
- **Type of internet connection** WiFi;
- Length of **daytime study** from 1 to 6;
- Low **financial conditions** of the student.
- Phone as a **device for learning**.
- **Age** from 21 - 25.

The following histogram (Fig. 14.) is slightly different from the one in Fig. 13. because it focuses on showing the importance of the features for each epoch of neural network training. These values allow us to calculate the average feature importance values (Y) and standard deviation (X).

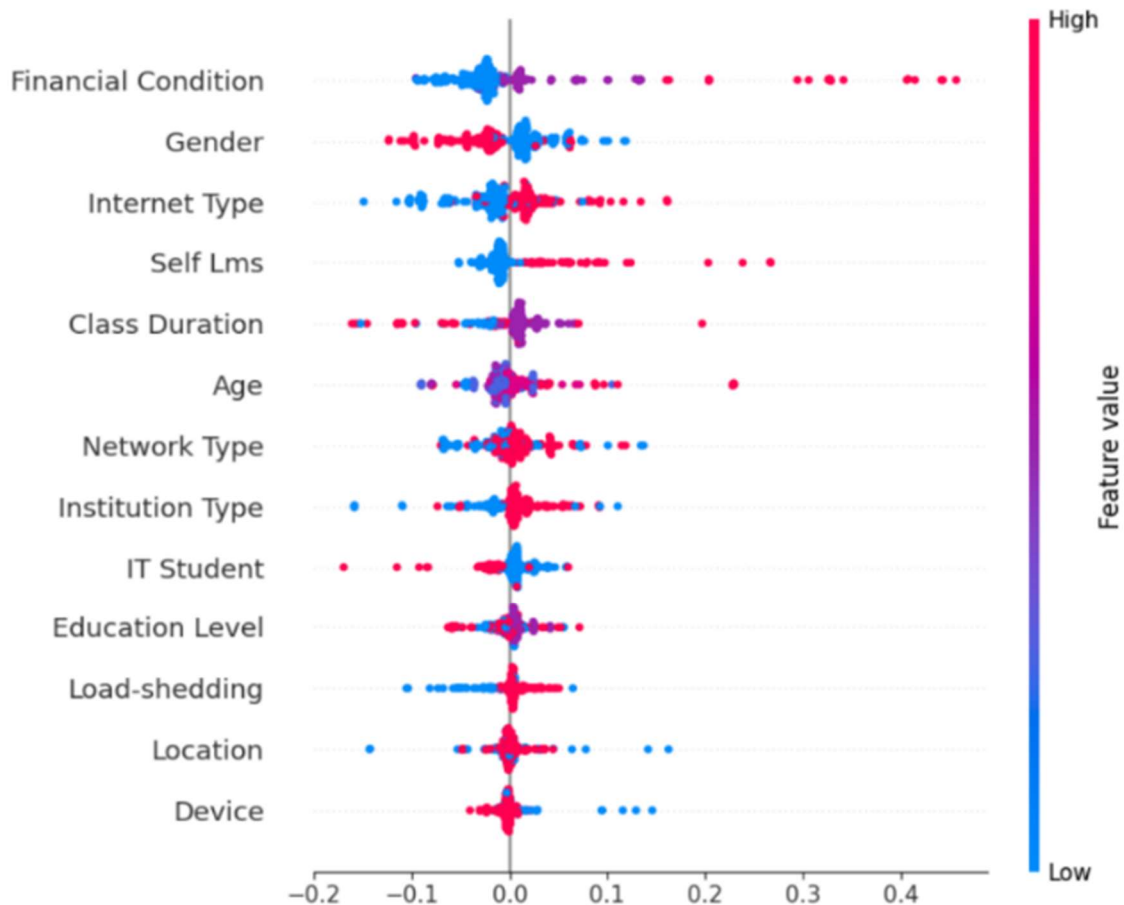


Figure 14: Permutation Feature Importance for Neural Network

Here, the X-axis represents the influence of a variable on the quality of the classification model. And the Y-axis is a set of characteristics ordered by importance. For example, the most important are: **financial conditions, gender, type of internet, LMS, and length of classes**, and the least important are: **load-shedding, stay in the city, and device used by the student**.

5. Conclusions

The paper investigates the effectiveness of machine learning tools that can solve the problem of assessing the level of student adaptation to online learning. These tools include intelligent methods and models, including classification methods and neural networks. The best performers were the Random Forest and the Sequential Neural Network, with results of 88% and 91% of accuracy.

The work can help research and find the most effective intellectual model or method to perform the task, which is to assess the level of adaptation of students to online education. The main practical value of the work is that the developed system allows using a few simple questions with high quality and accuracy to assess the level of adaptation of students to online learning. This, in turn, will help stakeholders in the process of implementing and applying online education to know how effective it will be.

The main perspective for developing the work includes the fact that the system can provide much better results after creating a more extensive set of data, which requires a survey of more students. This will help in the creation of models and their training. In particular, the system is flexible, which allows it to implement more classification models if necessary.

6. Acknowledgements

This work was realized within the framework of the program Erasmus+ Jean Monnet Module «Augmented Reality for Education: implementation of European experience» (101085772 — AR4EDU — ERASMUS-JMO-2022-HEI-TCH-RSCH).

7. References

- [1] M. Hasan Suzan, N. A. Samrin, A. A. Biswas and A. Pramanik, "Students' Adaptability Level Prediction in Online Education using Machine Learning Approaches," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021, pp. 1-7, doi: 10.1109/ICCCNT51525.2021.9579741.
- [2] What is a correlation matrix?. Educative: Interactive Courses for Software Developers. URL: <https://www.educative.io/answers/what-is-a-correlation-matrix> (date of access: 02.02.2023).
- [3] D. Savchuk and A. Doroshenko, "Investigation of machine learning classification methods effectiveness," 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 2021, pp. 33-37, doi: 10.1109/CSIT52700.2021.9648582.
- [4] What is a Decision Tree | IBM. IBM - Deutschland | IBM. URL: <https://www.ibm.com/topics/decision-trees> (date of access: 02.02.2023).
- [5] Introduction to Random Forest in Machine Learning. Engineering Education (EngEd) Program | Section. URL: <https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/> (date of access: 02.02.2023).
- [6] How to Develop an Extra Trees Ensemble with Python - MachineLearningMastery.com. MachineLearningMastery.com. URL: <https://machinelearningmastery.com/extra-trees-ensemble-with-python/> (date of access: 02.02.2023).
- [7] Gradient Boosting. WallStreetMojo. URL: <https://www.wallstreetmojo.com/gradient-boosting/>.
- [8] Christopher A. K-Nearest Neighbor. Medium. URL: <https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4> (date of access: 02.02.2023).
- [9] Thorn J. Logistic Regression Explained. Medium. URL: <https://towardsdatascience.com/logistic-regression-explained-9ee73cede081> (date of access: 02.02.2023).
- [10] Yazhi Gao, W. Rong, Y. Shen and Z. Xiong, "Convolutional Neural Network based sentiment analysis using Adaboost combination," 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 2016, pp. 1333-1338, doi: 10.1109/IJCNN.2016.7727352.
- [11] José Luis Rojo-Álvarez; Manel Martínez-Ramón; Jordi Muñoz-Marí; Gustau Camps-Valls, "Support Vector Machine and Kernel Classification Algorithms," in *Digital Signal Processing with Kernel Methods*, IEEE, 2018, pp.433-502, doi: 10.1002/9781118705810.ch10.

- [12] Yıldırım S. Naive Bayes Classifier–Explained. Medium. URL: <https://towardsdatascience.com/naive-bayes-classifier-explained-50f9723571ed> (date of access: 02.02.2023).
- [13] H. Luo and Y. Liu, "A prediction method based on improved ridge regression," *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China, 2017, pp. 596-599, doi: 10.1109/ICSESS.2017.8342986.
- [14] Tezcan B. Why Using a Dummy Classifier is a Smart Move. Medium. URL: <https://towardsdatascience.com/why-using-a-dummy-classifier-is-a-smart-move-4a55080e3549> (date of access: 02.02.2023).
- [15] J. Ghosh and S. B. Shuvo, "Improving Classification Model's Performance Using Linear Discriminant Analysis on Linear Data," *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019, pp. 1-5, doi: 10.1109/ICCCNT45670.2019.8944632*.
- [16] E. Pękalska and B. Haasdonk, "Kernel Discriminant Analysis for Positive Definite and Indefinite Kernels," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 1017-1032, June 2009, doi: 10.1109/TPAMI.2008.290.
- [17] A. Doroshenko and R. Tkachenko, "Classification of Imbalanced Classes Using the Committee of Neural Networks," *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT)*, Lviv, Ukraine, 2018, pp. 400-403, doi: 10.1109/STC-CSIT.2018.8526611.
- [18] Performance Measures: Cohen's Kappa statistic. The Data Scientist. URL: <https://thedata scientist.com/performance-measures-cohens-kappa-statistic/> (date of access: 02.02.2023).
- [19] sklearn.metrics.matthews_corrcoef. scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.matthews_corrcoef.html (date of access: 02.02.2023).
- [20] What is a Correlation Matrix? Displayr. URL: <https://www.displayr.com/what-is-a-correlation-matrix/> (date of access: 02.02.2023).
- [21] Feature Permutation Importance Explanations – ADS 1.0.0 documentation. Moved. URL: https://docs.oracle.com/en-us/iaas/tools/ads-sdk/latest/user_guide/mlx/permutation_importance.html (date of access: 02.02.2023).