AI-BASED SYSTEM FOR ANALYZING, PREDICTING AND VISUALIZING STUDENT GRADES

Vasil Kostadinov, Todorka Glushkova, Evgeni Valchev

Abstract. *In the context of digital transformation of education, the analysis of academic results is gaining increasing importance for improving the quality of teaching and early detection of anomalies. This paper presents an approach to smoothing a system for automated analysis of student grades, integrating methods from statistics, data analysis, and data visualization. The developed system processes Excel files, extracts and standardizes the data, then applies linear regression to predict final grades and K-Means clustering to detect groups with similar results. An anomaly detection mechanism based on the difference between intermediate and final grades is employed, along with a rule-based module that generates personalized recommendations for each student. The results are presented visually and exported in an enriched Excel format with conditional coloring, which facilitates teachers in analyzing performance and detecting problem cases. The system demonstrates potential for application in academic analytics, offering a basis for future expansion through classification models and a web-based interface.*

Key words: AI, Academic Analytics, Linear Regression, Clustering, K-Means.

Introduction

Assessment of students' knowledge and skills is one of the basic components of modern education, which aims not only to measure achievements, but also to make informed pedagogical decisions. In the conditions of digital transformation, traditional approaches to analyzing results are increasingly proving insufficient for timely detection of learning difficulties and planning personalized interventions. This increases the importance of methods based on statistical data processing, machine learning and the AIEd (Artificial Intelligence in Education) concept, aimed at applying AI in educational practice. The use of these approaches allows for more objective assessment, detection of hidden dependencies in data, identification of at-risk learners and implementation of adaptive learning strategies. Machine learning provides significant advantages – from predicting final grades through regression models to detecting groups of students with similar characteristics through clustering and detecting anomalies associated with atypical results.

In this context, this study presents a developed AI-based system for automated analysis of student grades, which combines elements of statistical analysis, linear regression, K-Means clustering and an anomaly detection mechanism. The obtained results are visualized and exported in an advanced Excel format with conditional coloring, which facilitates teachers in making decisions related to improving the quality of education.

Assessing students' knowledge. Results and discussion

Student assessment is a fundamental part of the educational process, not only measuring achievement but also providing a basis for planning personalized learning interventions and improving teaching strategies [1]. In recent years, there has been a growing interest in using mathematical, statistical, and especially AI methods to automate and objectify this process. The combination of statistical analysis and machine learning allows for a deeper understanding of students' academic performance and behavior patterns [2]. The traditional statistical approach involves calculating key indicators such as the mean, median, mode, and standard deviation, which provide information about the distribution of grades and the differences between groups of students. These indicators allow teachers to identify some deviations and make adjustments to the teaching process. For example, a histogram of final grades can visualize the data and show the concentration of results around the average values.

On the other hand, linear regression can model the relationship between students' intermediate and final grades [3]. A regression function of the type:

$$Final\ grade = \alpha + \beta_1.Grade1 + \beta_2.Grade2 + \ \ldots \ + \beta_4.Grade4$$

allows for the prediction of final success based on previous achievements [4]. A similar model is also used in the system presented in the article for automated prediction and verification of results.

In addition to classical statistical methods, machine learning provides opportunities for deeper analysis and classification. Three key approaches have been applied in the project under consideration:

- Linear Regression – used to predict final grades based on intermediate results. The algorithm extracts variable dependencies and assists the teacher in discovering patterns, aiming to identify a linear relationship between the input data and the target. The coefficients in the above formula reflect the importance of each intermediate grade.

- K-Means Clustering – the algorithm groups students into clusters according to the similarity of their grades. Group centers are calculated, and students are assigned based on their proximity. This allows for the identification of groups such as “high achievers”, “average”, and “needing support” [5]. The resulting groups can be used for personalized educational strategies. This method is used to visualize patterns of behavior or similarities.

- Anomaly detection – by comparing intermediate and final results, cases of unusually high or low scores are detected. Such a technique can be improved by algorithms such as Isolation Forest [6] or Local Outlier Factor [7], which automatically identify deviations without the need for predefined thresholds.

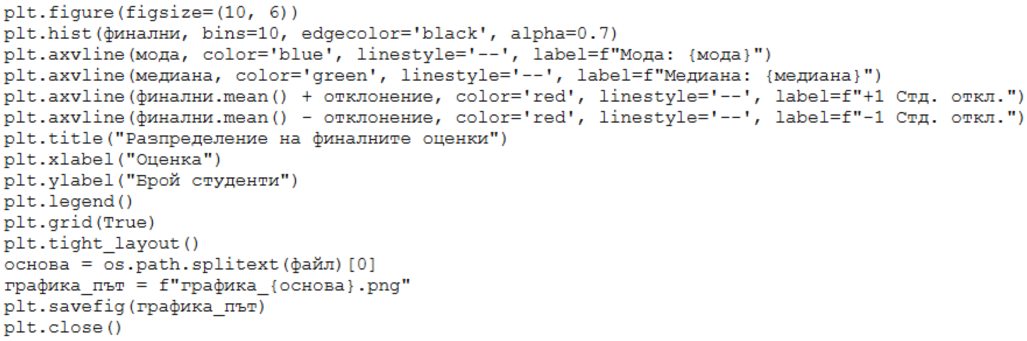
Isolation Forest (IF) is an algorithm based on the principle of isolation. Instead of modeling normal observations, as classical probabilistic approaches do, IF isolates anomalies by randomly splitting the data. At each split (“branch”), the algorithm chooses a random feature and a random split value. Since anomalies are usually rare and differ significantly, they are isolated more quickly – i.e., at a shallower depth in the tree. The average isolation depth is used to calculate the so-called “anomalous index” (score). The smaller this value, the more likely the observation is to be an anomaly. The advantages of Isolation Forest are: high efficiency on large data sets (works with linear complexity $O(n)$); independence from the dimensionality of the data as it does not require normalization; easy interpretation as the result can be easily visualized and integrated into educational systems to detect unexpected grades.

On the other hand, Local Outlier Factor (LOF) uses a density approach that assesses the degree to which a point is rarer in its local environment compared to neighboring points. A local density is calculated and compared to the densities of the k nearest neighbors. If a point has a significantly lower local density, it is classified as anomalous. This algorithm is useful for groups of students with different academic profiles and can show that a student has a grade that is “anomalous” only compared to his group, and not to the entire course, making it a valuable tool for context-aware evaluation. The combined use of Isolation Forest and LOF allows AI-based educational systems to detect both global and local anomalies, increasing the reliability and objectivity of the assessment process.

The main goal of the study is to conduct a quantitative analysis of the success rate of students in the discipline “Discrete Mathematics”, using basic methods of descriptive statistics and AI algorithms. Through statistical data processing, the aim is to establish the general level of mastery of the material; to compare the results of different forms of assessment (homework, control, test); to assess the uniformity and variation of the results; and to derive a generalized assessment for the group. By applying selected AI algorithms to assess anomalies and perform a deeper analysis and classification. The object of the study is the students of the "Informatics" specialty at the University of Plovdiv, studying the discipline "Discrete Mathematics". The subject of the study, with the result of the current control of their knowledge, includes three homework assignments and one control test. The study used data from real student results recorded in a spreadsheet (Grades.xlsx). The processing was done by Microsoft Excel and Python (libraries Pandas, NumPy, SciPy and Matplotlib). The research was implemented using Python scripts in two steps:

- Statistical processing and analysis of the results
- Application of AI algorithms to assess anomalies and perform a deeper analysis and classification.

The following statistical characteristics were calculated for each variable and for the total score: *Mean* ($\bar{x}$) and *Median* (Me), which characterize the average values; *Mode* (Mo) – the most common score; *Standard deviation* (s) and *mean square deviation* (σ), which assess the degree of dispersion of the scores around the mean; *Coefficient of variation* (V) to represent the relative variability of the scores; *Skewness* (Sk) and *Kurtosis* (K) to assess the shape of the distribution (normal, right/left skewed). Part of the Python statistical analysis implementation is visualized in Figure 1.



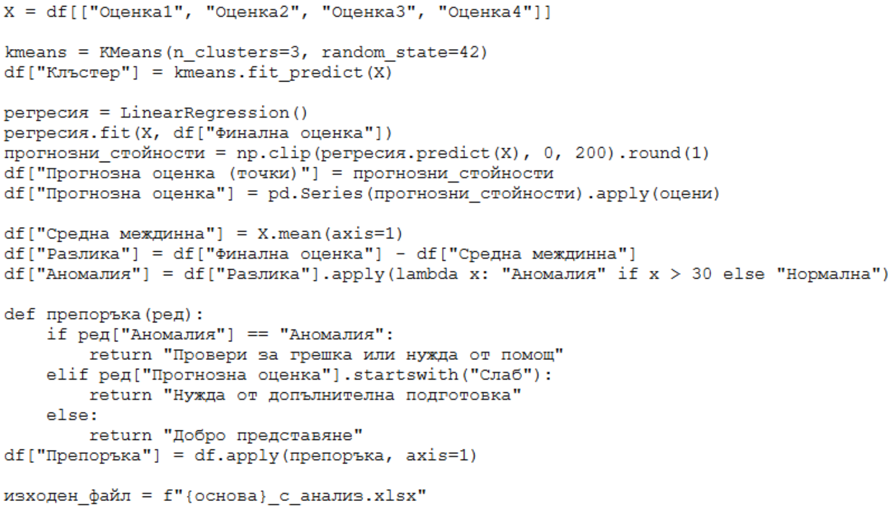
```
plt.figure(figsize=(10, 6))
plt.hist(финални, bins=10, edgecolor='black', alpha=0.7)
plt.axvline(мода, color='blue', linestyle='--', label=f"Мода: {мода}")
plt.axvline(медиана, color='green', linestyle='--', label=f"Медиана: {медиана}")
plt.axvline(финални.mean() + отклонение, color='red', linestyle='--', label=f"+1 Стд. откл.")
plt.axvline(финални.mean() - отклонение, color='red', linestyle='--', label=f"-1 Стд. откл.")
plt.title("Разпределение на финалните оценки")
plt.xlabel("Оценка")
plt.ylabel("Брой студенти")
plt.legend()
plt.grid(True)
plt.tight_layout()
основа = os.path.splitext(файл)[0]
графика_път = f"графика_{основа}.png"
plt.savefig(графика_път)
plt.close()
```

Figure 1. Statistical analysis in Python

The results obtained allow for the assessment of students' results, but do not enable the identification of anomalies or discrepancies for each student. To perform such an analysis, we use the AI algorithms Linear Regression, K-Means Clustering, and Anomaly detection, through which we can identify these anomalies, classify the students into separate groups, and make predictions about the final results of the studies. The graphical presentation of the distribution of the results allows us to clearly observe the trends in the success of the students.

The visualization shows that the grades are not evenly distributed, forming zones of higher concentration around the average. This suggests the presence of a relatively homogeneous group of students, which performs stably, but at the same time, individual values are also noticeable, which deviate significantly from the center of the distribution. Such deviations have important analytical significance, as they can be an indication of gaps in preparation, inaccuracies in assessment, or individual difficulties.

This visual stage of the analysis serves as the basis for the next phase, which involves machine learning algorithms – linear regression, clustering, and anomaly detection – to extract deeper and more interpretable information about student performance. We again use Python to implement this step. Figure 2 shows a portion of the script.



```
X = df[["Оценка1", "Оценка2", "Оценка3", "Оценка4"]]

kmeans = KMeans(n_clusters=3, random_state=42)
df["Клъстер"] = kmeans.fit_predict(X)

регресия = LinearRegression()
регресия.fit(X, df["Финална оценка"])
прогнозни_стойности = np.clip(регресия.predict(X), 0, 200).round(1)
df["Прогнозна оценка (точки)"] = прогнозни_стойности
df["Прогнозна оценка"] = pd.Series(прогнозни_стойности).apply(оцени)

df["Средна междинна"] = X.mean(axis=1)
df["Разлика"] = df["Финална оценка"] - df["Средна междинна"]
df["Аномалия"] = df["Разлика"].apply(lambda x: "Аномалия" if x > 30 else "Нормална")

def препоръка(ред):
    if ред["Аномалия"] == "Аномалия":
        return "Провери за грешка или нужда от помощ"
    elif ред["Прогнозна оценка"].startswith("Слаб"):
        return "Нужда от допълнителна подготовка"
    else:
        return "Добро представяне"
df["Препоръка"] = df.apply(препоръка, axis=1)

изходен_файл = f"{основа}_с_анализ.xlsx"
```

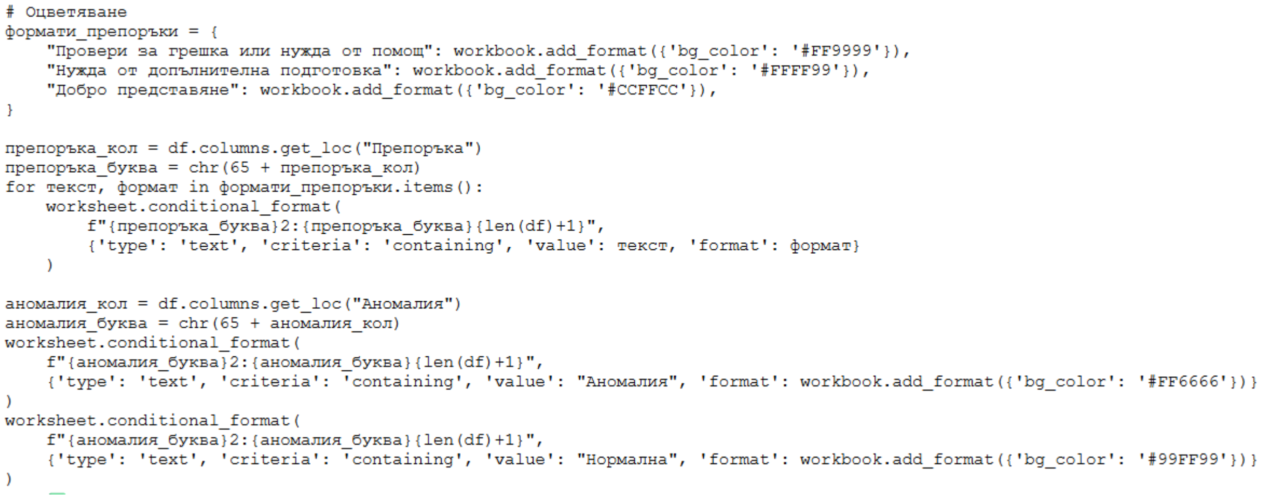
Figure 2. Python script for AI analysis

To improve readability and facilitate analysis of the results, a system for automatic conditional coloring of cells in the source Excel file has been implemented. It is based on the logic embedded in the Python script (Figure 4) and uses the conditional formatting mechanism from the XlsxWriter library. Coloring is divided into two main groups – recommendations and anomalies, with specific color formats defined for each of them.

Recommendation coloring. The Recommendation column applies three different color formats, corresponding to the automatically generated text messages: "Check for error or need help" – is highlighted in red (#FF9999), as it indicates a serious deviation and needs attention; "Needs additional training" – is colored in yellow (#FFFF99), which signals a moderate deviation or omissions; "Good performance" – is visualized in light green (#CCFFCC), emphasizing a positive result. The script automatically determines the range of the column by finding its position in the DataFrame, then adds conditional rules of type text contains, which color the cells according to the text they contain.

Anomaly Coloring. A similar approach is used for the "Anomaly" column, but with two formats: when the text "Anomaly" is present, the cell is highlighted in dark red (#FF6666), as it indicates a significant or unexpected deviation in the results; when the value is "Normal", the cell is colored green (#99FF99), which indicates the absence of deviations.

Just like with recommendations, the range is dynamically determined based on the column's position in the table, and the conditional rules are applied to all cells until the end of the record. Part of the Python scripts for displaying messages and coloring is visualized in the following Figure 3.



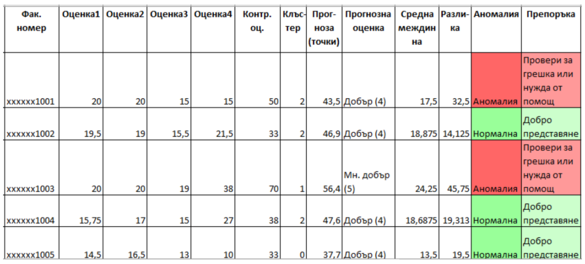
```
# Оцветяване
формати_препоръки = {
    "Провери за грешка или нужда от помощ": workbook.add_format({'bg_color': '#FF9999'}),
    "Нужда от допълнителна подготовка": workbook.add_format({'bg_color': '#FFFF99'}),
    "Добро представяне": workbook.add_format({'bg_color': '#CCFFCC'}),
}

препоръка_кол = df.columns.get_loc("Препоръка")
препоръка_буква = chr(65 + препоръка_кол)
for текст, формат in формати_препоръки.items():
    worksheet.conditional_format(
        f"{препоръка_буква}2:{препоръка_буква}{len(df)+1}",
        {'type': 'text', 'criteria': 'containing', 'value': текст, 'format': формат}
    )

аномалия_кол = df.columns.get_loc("Аномалия")
аномалия_буква = chr(65 + аномалия_кол)
worksheet.conditional_format(
    f"{аномалия_буква}2:{аномалия_буква}{len(df)+1}",
    {'type': 'text', 'criteria': 'containing', 'value': "Аномалия", 'format': workbook.add_format({'bg_color': '#FF6666'})}
)
worksheet.conditional_format(
    f"{аномалия_буква}2:{аномалия_буква}{len(df)+1}",
    {'type': 'text', 'criteria': 'containing', 'value': "Нормална", 'format': workbook.add_format({'bg_color': '#99FF99'})}
)
```

Figure 3. Python script for AI analysis

Figure 4 shows an example fragment of the output Excel file generated by the developed AI system. The table contains integrated results from statistical analysis, machine learning, and automatic rule-based generation of recommendations. Each row corresponds to an individual student and includes: Current control grades (homework assignments and a test); Cluster – grouping using K-Means; Predicted grade and final predicted grade calculated using linear regression; Average intermediate grade; Difference between predicted and intermediate value; Anomaly; Recommendation generated by the system. The last two columns are colored automatically, in accordance with the script in Figure 4, which visually highlights key cases and helps to detect problematic or interesting results more quickly. This integration of AI algorithms and visual indicators allows the instructor to quickly identify students at risk, cases of atypical behavior, as well as students with excellent results. The output presented in Figure 4 demonstrates how data, modeling, and visualization are combined into a single tool to support the academic process.



Фак. номер	Оценка1	Оценка2	Оценка3	Оценка4	Контр. оц.	Клъс-тер	Прог-ноза (точки)	Прогнозна оценка	Средна междин на	Разли-ка	Аномалия	Препоръка
xxxxxx1001	20	20	15	15	50	2	43,5	Добър (4)	17,5	32,5	Аномалия	Провери за грешка или нужда от помощ
xxxxxx1002	19,5	19	15,5	21,5	33	2	46,9	Добър (4)	18,875	14,125	Нормална	Добро представяне
xxxxxx1003	20	20	19	38	70	1	56,4	Мн. добър (5)	24,25	45,75	Аномалия	Провери за грешка или нужда от помощ
xxxxxx1004	15,75	17	15	27	38	2	47,6	Добър (4)	18,6875	19,313	Нормална	Добро представяне
xxxxxx1005	14,5	16,5	13	10	33	0	37,7	Добър (4)	13,5	19,5	Нормална	Добро представяне

Figure 4. The results from AI analysis

Conclusions and future plans

The developed AI-based system for analyzing, predicting, and visualizing student grades demonstrates strong potential for supporting data-driven decision-making in educational environments. By combining classical statistical indicators with predictive analytics models, the system provides a deeper and more precise understanding of student performance, allowing for the early detection of anomalies, the identification of performance clusters, and the generation of personalized recommendations. The integration of automated color-coded feedback and anomaly detection into the exported Excel format presents a practical and accessible interface for educators, enabling quick interpretation of complex data. This confirms that the approach is suitable not only for research purposes but also for real educational practice.

Future development of the system will focus on several major directions. First, the inclusion of additional machine learning methods – such as classification models, ensemble algorithms, or deep learning networks – could improve the accuracy of predictions and enable more nuanced detection of at-risk students. Second, expanding the rule-based recommendation module to incorporate pedagogical strategies and adaptive learning pathways would further enhance its applicability. Third, the inclusion of additional machine learning methods. Finally, developing a web-based platform or dashboard would allow real-time data processing, multi-course integration, and easier access for teachers and administrators.

Ethical considerations regarding the confidentiality and security of the data used should also be taken into account in the design of the system, ensuring compliance with relevant regulations and obtaining informed consent for data collection and processing. Furthermore, to improve scalability and applicability in different educational contexts, the system architecture can be optimized for processing larger data sets and integration into different courses.

Overall, the system provides a solid foundation for building advanced educational analytics tools, supporting both instructors and students through enhanced insights, timely feedback, and more informed pedagogical decisions.

Acknowledgments

The paper is partially supported by project FP25-FMI-010 "Innovative interdisciplinary research in the field of informatics, mathematics, and pedagogy" and project MUPD25-FMI-013 "Innovative Research and Technological Solutions in the Field of ICT" of the Scientific Fund of the Paisii Hilendarski University of Plovdiv, Bulgaria.

References

[1] J. Biggs, C. Tang, G. Kennedy, *Teaching for quality learning at university 5e*, McGraw-hill education (UK), 2022, 440, ISBN: 9780335250820

[2] Q. Nguyen, B. Rienties, L. Toetenel, Unravelling the dynamics of instructional practice: A machine learning approach in learning analytics, *Computers in Human Behavior*, 2020, 107, 106290

[3] O. El Aissaoui, Y. El Alami El Madani, L. Oughdir, A. Dakkak, Y. El Allioui, A multiple linear regression-based approach to predict student performance, *International conference on advanced intelligent systems for sustainable development*, Cham: Springer International Publishing, 2019, July, pp. 9–23

[4] G. James, D. Witten, T. Hastie, R. Tibshirani, *An introduction to statistical learning: with applications in R*, New York: Springer, 2021, Vol. 103, 607, ISBN: 978-1-0716-1417-4

[5] B. Chong, K-means clustering algorithm: a brief review. Academic Journal of Computing & Information Science, 2021, 4 (5), 37–40, ISSN: 2616-5775, 10.25236/AJCIS.2021.040506

[6] D. Xu, Y. Wang, Y. Meng, Z. Zhang, An improved data anomaly detection method based on isolation forest, *2017 10th international symposium on computational intelligence and design (ISCID)*, 2017, Vol. 2, pp. 287–291

[7] M. Breunig, H. Kriegel, R. Ng, J. Sander, LOF: Identifying density-based local outliers, *ACM SIGMOD Record*, 2020, 29 (2), 93–104

Vasil Kostadinov[1], Todorka Glushkova[1], Evgeni Valchev[1]
[1] Paisii Hilendarski University of Plovdiv,
Faculty of Mathematics and Informatics,
236 Bulgaria Blvd., 4027 Plovdiv, Bulgaria
Corresponding author: glushkova@uni-plovdiv.bg