DOI 10.31891/2307-5732-2024-335-3-47
УДК 004.8

ПИЛИПЕНКО ВЛАДИСЛАВ
Київський національний університет технологій та дизайну
https://orcid.org/0000-0002-2761-4817
e-mail: pylypenko.vi@knutd.edu.ua
СТАЦЕНКО ВОЛОДИМИР
Київський національний університет технологій та дизайну
https://orcid.org/0000-0002-3932-792X
e-mail: statsenko.v@knutd.edu.ua

ДОСЛІДЖЕННЯ ТОЧНОСТІ МЕТОДІВ МАШИННОГО НАВЧАННЯ ПРИ ПРОГНОЗУВАННІ УСПІШНОСТІ СТУДЕНТІВ

В роботі проведено дослідження точності трьох методів машинного навчання у задачі прогнозування успішності студентів на основі даних про їх роботу з навчальними матеріалами. Досліджувались наступні методи машинного навчання: Logistic Regression, SVM, Random Forest. Визначено параметри, що характеризують точність класифікації, а саме: чутливість (Sensitivity), специфічність (Specificity) та збалансовану точність (Balanced Accuracy). За отриманими результатами побудовано графіки ROC-кривої для оцінки здатності класифікатора правильно розпізнавати позитивні класи і відхиляти негативні класи при зміні порогового значення.

Для виконання прогнозування із бази даних електронної системи управління навчанням Moodle були експортовані дані оцінок та відвідуваності користувачів за перший та другий семестр 2021-2022 навчального року. Загальна кількість записів з оцінками та відвідуваністю студентів, експортованих у файли склала 1,308,262. Для обробки даних з файлів було написано додаток у середовищі розробки Microsoft Visual Studio на мові C#. Для обробки колонок таблиць використовувалася бібліотека CsvHelper. Створення моделей для прогнозування виконано у середовищі розробки PyCharm на мові Python із використанням бібліотеки Scikit-learn. Після проведення розрахунків визначено, що метод випадкового лісу найкраще виконує прогнозування успішності на основі наявних вхідних даних і має більшу точність. Отримане значення точності 80% та AUC 73% свідчить про якість моделі класифікації, та гарну дискримінаційну силу моделі. По графіку ROC-кривої для методу випадкового лісу видно, що він має чітко виражену ділянку під кривою яка більше вигнута вгору і вліво, що підтверджує ефективність моделі. Отримані результати можуть бути використані в якості основи для подальших досліджень.

Ключові слова: Logistic Regression; SVM; Random Forest; Machine Learning; Python; Scikit-learn.

PYLYPENKO VLADYSLAV, STATSENKO VOLODYMYR
Kyiv National University of Technologies and Design

ASSESSMENT OF THE EFFICIENCY OF THE SUCCESS PREDICTION MODEL USING MACHINE LEARNING METHODS

The paper compares the accuracy of predicting success based on attendance among the following machine learning methods: Logistic Regression, SVM, Random Forest. The following values characterizing classification accuracy were determined: sensitivity, accuracy, specificity, and balanced accuracy. Based on the obtained results, ROC curve graphs were constructed to assess the classifier's ability to correctly recognize positive classes and reject negative classes when the threshold value changes. In order to perform forecasting, data on user ratings and attendance for the first and second semesters of the 2021-2022 academic year were exported from the Moodle database. The total number of records of subject grades and student attendance exported to files was: 1,308,262. To process data from files, an application was written in the Microsoft Visual Studio development environment in the C# language. The CsvHelper library was used to process table columns. The creation of predictive models was performed in the PyCharm development environment in Python using the Scikit-learn library.

After the calculations, it was determined that the random forest method performs the best prediction of success on the input data and has a higher accuracy. Random forest is an ensemble method that usually works better in cases where the relationships between features and the original classes are more complex, non-linear, or when there are many features. It can automatically consider feature importance and make better predictions than linear models on complex data.

The obtained accuracy value of 80% and AUC of 73% indicates the quality of the classification model and good discriminating power of the model. The graph of the ROC curve for the random forest method shows that it has a clearly defined area under the curve that is more curved up and to the left, which shows the effectiveness of the model. This result is a good enough starting point, but in the process of further research, additional refinement may be needed to improve these indicators. It is also important to increase the amount of data, as well as add new features to obtain a more accurate result. Obtaining additional features is possible by creating additional plugins for Moodle.

Keywords: Logistic Regression; SVM; Random Forest; Machine Learning; Python; Scikit-learn.

Постановка проблеми

Одним із основних факторів, що впливає на майбутні можливості та кар'єрний розвиток студентів є результати освітнього процеси. Високий рівень успішності в навчанні дозволяє в подальшому отримати кваліфікованих спеціалістів та забезпечити економічне зростання країни. Прогнозування успішності є актуальним завданням, оскільки швидке реагування на можливі проблеми дозволяє збільшити шанси студента на успішну здачу сесії, а також виявити недоліки, що можна виправити. Оцінювання та прогнозування успішності, як ризику, дає змогу відповідальним сторонам краще розуміти, які фактори впливають на результат та яка результативність засобів контролю. Ефективним інструментом для прогнозування майбутніх значень на основі наявних даних є методи машинного навчання. Алгоритми класифікації, такі як: логістична регресія (Logistic Regression), метод опорних векторів (SVM), випадковий ліс (Random Forest) широко використовуються для прогнозування ймовірностей настання певних подій [1]. Зазвичай ці моделі навчаються на вхідних даних, які містять як ознаки (фактори), так і відповідні цільові значення. Після навчання модель може використовуватися для прогнозування ймовірностей для нових даних. Методи машинного навчання

особливо корисні при роботі з великим обсягом даних або складними взаємозв'язками між ознаками. Вони можуть автоматично виявляти складні взаємозв'язки у даних, що дозволяє робити точні прогнози ймовірностей. Проте кожен алгоритм має свої переваги та недоліки, що в впливають на точність прогнозування. Для порівняння точності прогнозування успішності студентів на основі інформації про їх роботу з освітніми матеріалами за допомогою різних методів машинного навчання необхідно побудувати відповідні моделі на однаковому набору даних оцінок та відвідуваності користувачів платформи Moodle.

Аналіз останніх джерел

Аналіз літературних джерел показав, що прогнозування успішності є важливим напрямом, який постійно удосконалюються за рахунок використання комп'ютерних засобів та програмних рішень. А рівень активності користувачів в LMS дає можливість відслідковувати ступінь залучення студента до освітнього процесу, що в свою чергу також впливає на успішність.

Згідно [2] прогнозування можна виконувати використовуючи метод випадкового лісу "Random Forest" для задач класифікації. Він добре підходить для прогнозування категорії або класу нового зразка на основі його характеристик. Розрахунок точності моделі складає 83% і показує, що прогнозування результатів можна провести досить точно.

Згідно [3] проведене оцінювання моделі прогнозування успішності на базі методів машинного навчання показало, що дані моделі можуть мати високу ефективність, і можуть бути використані в прикладних задачах. У дослідженні загальна ефективність моделі склала - 89%, а точність прогнозування успішності склала - 84%.

Згідно [4] для оцінки ефективності та якості моделі було запропоновано наступні ключові показники: точність (Accuracy), чутливість (Sensitivity), специфічність (Specificity), збалансована точність (Balanced Accuracy). А також побудовано ROC-криву для відображення здатності класифікатора правильно розпізнавати позитивні класи і відхиляти негативні класи при зміні порогового значення та визначено AUC (Area Under Curve).

Згідно [5] проведено порівняння точності класифікації алгоритмів: Random Forest, SVM, Logistic Regression, Naive Bayes, точність яких становила 74,6%, 73,5%, 71,7%, та 71,3%. Алгоритм Random Forest зміг досягти найбільшої точності, у результаті 74,6% зразків були класифіковані правильно. Результати показали, що запропонована модель досягла точності класифікації 70 – 75%.

Згідно [6] проведено порівняння прогнозування успішності студента за допомогою шести алгоритмів машинного навчання: Decision Tree (C5.0), Naive Bayes, Random Forest, Support Vector Machine, K-Nearest Neighbor та Deep neural network. Отримана точність склала: 69%, 73%, 79%, 69%, 75% та 84%. Результати показали, що алгоритм Deep neural network має найвищу точність 84% прогнозування успішності.

Метою роботи є: проведення оцінки точності методів машинного навчання у задачі прогнозування успішності студентів, що є користувачами LMS Moodle.

Виклад основного матеріалу

Дослідження проводилось на основі інформації з бази даних Moodle. Були експортовані у csv формат оцінки та відвідування користувачів за перший та другий семестр 2021-2022 навчального року. Загальна кількість записів по оцінкам з дисциплін та відвідуваності студентів, експортованих у файли склала: 1,308,262. Структура таблиці з оцінками по дисциплінам представлено у табл. 1, а структура таблиці з даними відвідуваності представлено у табл. 2.

Після цього дані по відвідуваності та оцінкам були розділені по семестрам. Для кожного студента по унікальному ідентифікатору «id_student» було виконано прив'язку оцінок та відвідуваності по кожному предмету. Результуюча оцінка за кожен предмет вираховувалася шляхом сумування усіх отриманих балів з колонки «value» в табл.1. А також додано загальне значення відсотку відвідуваності, за кожен вид занять, яке вираховувалося шляхом сумування з колонки «value» в табл. 2.

Таблиця 1

Таблиця з оцінками по дисциплінам експортована з БД Moodle

id_student	date	id_sem	value	name	who_write	student_name	discipline_name	group_name
…	…	…	…	…	…	…	…	…
23163	22.02.2022	2	5	Звіт	522	#####	Алгоритми даних	#####
23171	23.02.2022	2	5	Звіт	522	#####	Алгоритми даних	#####
…	…	…	…	…	…	…	…	…

де id_student – унікальний ідентифікатор користувача;

date – дата запису даних;

id_sem – позначення семестру (1 або 2);

value – значення оцінки отриманої студентом за виконаний вид роботи;

name – назва виду роботи, яку виконував студент;

who_write – ідентифікатор викладача;

student_name – ім'я та прізвище користувача;

discipline_name – назва дисципліни;

group_name – назва групи.

Таблиця 2

Таблиця з даними про відвідуваність експортована з БД Moodle

id_student	date_write	id_sem	value	who_write	student_name	discipline_name	group_name	para_name
…	…	…	…	…	…	…	…	…
23163	22.02.2022	2	0	582	#####	Алгоритми даних	#####	Практ. заняття
23171	23.02.2022	2	1	582	#####	Алгоритми даних	#####	Практ. заняття
…	…	…	…	…	…	…	…	…

де id_student – унікальний ідентифікатор користувача;
date_write – дата запису даних;
id_sem – позначення семестру (1 або 2);
value – значення присутності (1-так, 0-ні);
who_write – ідентифікатор викладача;
student_name – ім'я та прізвище користувача;
discipline_name – назва дисципліни;
group_name – назва групи;
para_name – назва виду заняття (лекція, лабораторна, семінар).

Для обробки даних з файлів було написано додаток з UI-інтерфейсом на WindowsForms, загальний вигляд представлено на рис. 1.

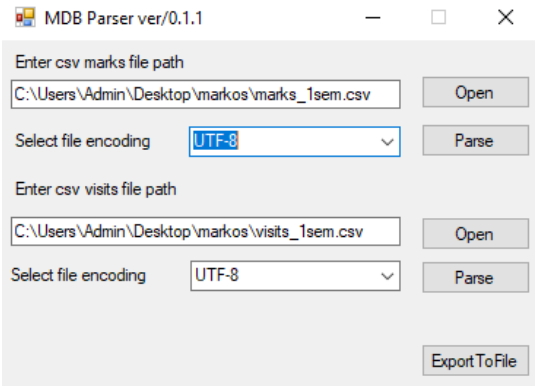


Рис. 1. Додаток для обробки даних із таблиць

Додаток написаний на мові C# у середовищі розробки Microsoft Visual Studio. Для обробки колонок таблиць використовувалася бібліотека CsvHelper. Перелік отриманих значень наведено у табл. 3.

Таблиця 3

Вихідні дані з оцінками та відсотком відвідуваності

group	id_student	discipline_name	mark	lection_visits	practice_visits	lab_visits	total_visits
#####	20833	WEB-технології	86	83	-1	83	83
#####	20844	WEB-технології	51	25	-1	33	29

де group – назва групи;
id_student – унікальний ідентифікатор користувача;
discipline_name – назва дисципліни;
mark – оцінка за предмет;
lection_visits – відсоток відвідування лекцій;
practice_visits – відсоток відвідування практичних занять;
lab_visits – відсоток відвідування лабораторних занять;
total_visits – загальний відсоток відвідування.

Якщо одного, з наявних в таблиці, виду занять у студента по предмету не було там вказується -1.

Для виконання прогнозування успішності відносно відвідуваності було використано наступні методи машинного навчання для задачі класифікації: логістична регресія [7], метод опорних векторів [8] та випадковий ліс [9]. Створення моделей для прогнозування виконано на мові Python із використанням бібліотеки Scikit-learn.

Перед виконанням навчання моделі вихідні дані були розділені на тренувальну та тестову вибірки для того, щоб перевірити, наскільки добре модель, навчена на тренувальній вибірці, може передбачати класи нових даних. Обсяг даних взятих для обробки складав 68670 вибірок користувачів, які були розподілені у відношенні 10225/58445. З яких тренувальна вибірка містила – 58445, а тестова – 10225. Ділення даних на тренувальну та

тестову вибірки допомагає уникнути перенавчання (overfitting) моделі [10].Оцінка та перевірка якості моделей здійснювалась на основі тестової вибірки. Моделі були побудовані за наступними ознаками (features): загальна відвідуваність, відвідуваність на лекціях, лабораторних та практичних заняттях по кожній дисципліні. У лістингу 1 представлено розділ даних на навчальний та тестовий набори, ініціалізацію та тренування для випадкового лісу (RandomForest) з використанням параметрів по замовчуванню.

Лістинг 1. Програмний код для поділу даних, ініціалізацію та тренування

```
# Розділ даних на навчальний та тестувальний набори
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Ініціалізація та навчання моделі випадкового лісу
model = RandomForestClassifier()
model.fit(X_train, y_train)
# Проведення прогнозування на тестувальному наборі
y_pred = model.predict(X_test)
```

При побудові моделей для кожного алгориту використовувалися параметри по замовчуванню. Нижче представлено повний список параметрів для RandomForestClassifier [11] у лістингу 2, для LogisticRegression [12] у лістингу 3 та для
SVM [13] у лістингу 4.

Лістинг 2. Повний список параметрів для RandomForestClassifier

class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='sqrt', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None, monotonic_cst=None)

Параметр *n_estimators* - задає кількість дерев у лісі, *criterion* - визначає критерій (параметр залежить від дерева), *max_depth* - задає максимальну глибина дерева, *min_samples_split* - мінімальна кількість зразків, необхідних для розбиття внутрішнього вузла, *min_samples_leaf* - визначає мінімальну суму ваги для кожного листка, *min_weight_fraction_leaf* - мінімальна зважена частка загальної суми ваг (усіх вхідних вибірок), *max_features* - визначає кількість ознак, *max_leaf_nodes* - кількість листових вузлів, *min_impurity_decrease* – розщеплення вузла, *bootstrap* – визначає вибір вибірки, *oob_score* - оцінка набору навчальних даних, *n_jobs* - кількість завдань, *random_state* - контролює випадковість початкового завантаження зразків, verbose - відображає вихідну інформацію, *warm_start* – визначає, чи буде використовуватися попередньо навчена модель для ініціалізації та навчання нової моделі, *class_weight* - вирівнювання ваги класів, *ccp_alpha* - параметр обрізки, *max_samples* - максимальна кількість зразків, *monotonic_cst* - обмеження монотонності.

Лістинг 3. Повний список параметрів для LogisticRegression

class sklearn.linear_model.LogisticRegression(penalty='l2', *, dual=False, tol=0.0001, C=1.0, fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='lbfgs', max_iter=100, multi_class='auto', verbose=0, warm_start=False, n_jobs=None, l1_ratio=None)

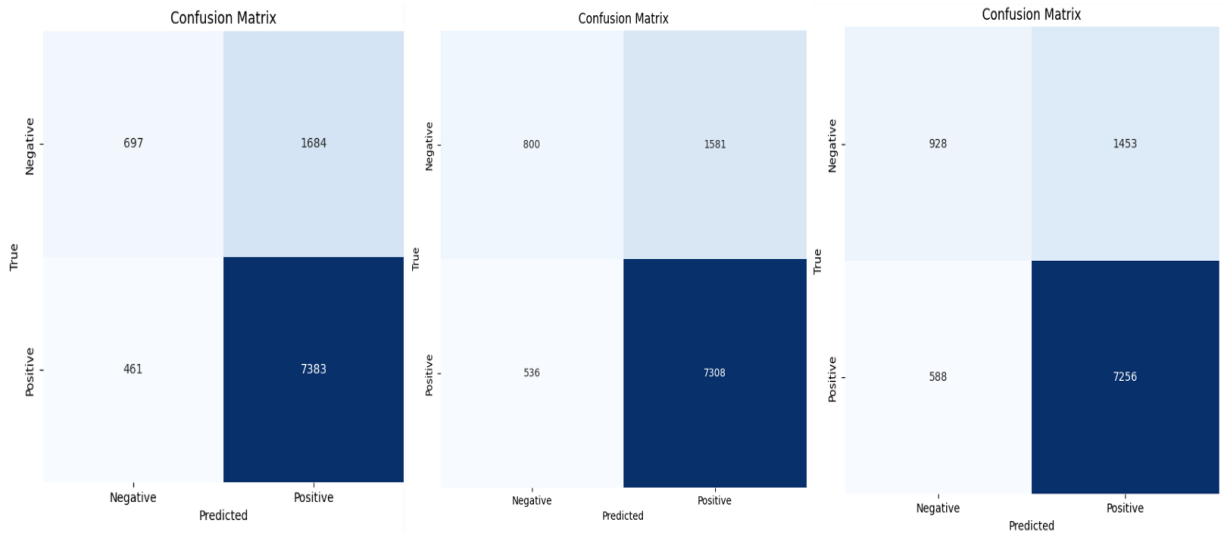
Параметр *penalty* – задає значення штрафу, *dual* - формулювання оптимізаційної задачі, *tol* - порогове значення для зупинки процесу оптимізації, *fit_intercept* - визначає, чи слід додавати константу до функції прийняття рішень, *intercept_scaling* - масштабування константного члена, *class_weight* - збалансування ваги класів, *random_state* - значення для генератора випадкових чисел, *solver* - алгоритм для використання в задачі оптимізації, *max_iter* - максимальна кількість ітерацій, *multi_class* - визначає підхід до обробки з багатьма класами, *verbose* - для розв'язувачів liblinear та lbfgs, *warm_start* - повторне використання рішення, *n_jobs* - кількість ядер ЦП, *l1_ratio* - параметр змішування.

Лістинг 4. Повний список параметрів для SVM

class sklearn.svm.SVC(*, C=1.0, kernel='rbf', degree=3, gamma='scale', coef0=0.0, shrinking=True, probability=False, tol=0.001, cache_size=200, class_weight=None, verbose=False, max_iter=-1, decision_function_shape='ovr', break_ties=False, random_state=None)

Параметр *C* - параметр регуляризації, *kernel* - визначає тип ядра, *degree* - ступінь поліноміальної функції, *gamma* - ядерний коефіцієнт, *coef0* - незалежний термін у функції ядра, *shrinking* – вказує чи використовувати евристику скорочення, *probability* – вмикання оцінки ймовірності, *tol* - критерій допуску до зупинки, *cache_size* - розмір кешу ядра, *class_weight* - множники параметра C для кожного класу, *verbose* - увімкнення докладного виводу, *max_iter* - обмеження на ітерації, *decision_function_shape* – віибір прийняття рішень, *break_ties* - розрив зв'язків, *random_state* - генерація псевдовипадкових чисел.

Основними критеріями ефективності моделі були обрані показники: точність, заблансована точність, чутливість, специфічність, AUC та ROC-крива. Ці показники розраховуються на основі так званої матриці помилок (confusion matrix) [14]. Матриця помилок моделі дозволяє порахувати, для скількох студентів прогнозування було виконано правильно. Отримані матриці помилок, для створених моделей, представлені на рис. 2.



а) Логістична регресія б) Метод опорних векторів в) Випадковий ліс

Рис. 2. Матриці помилок для створених моделей

Виходячи з отриманих матриць проведено розрахунок значень характеризуючих загальну точність класифікації, а саме: чутливості, точності, специфічності, збалансованої точності. Результати розрахунків наведені в табл. 4.

Таблиця 4

Розрахунки значень характеризуючих загальну точність

Метод	**Точність**	**Чутливість**	**Специфічність**	**Збалансована точність**	**Загальна ефективність (AUC)**
Логістична регресія	0.79	0.941	0.292	0.616	0.7
Метод опорних векторів	0.79	0.931	0.335	0.633	0.66
Випадковий ліс	0.80	0.925	0.389	0.657	0.73

Щоб наглядно оцінити здатність моделі до правильної класифікації, враховуючи різні значення порогового значення було побудовано ROC-криву (Receiver Operating Characteristic) [15]. ROC-крива відображає здатність класифікатора правильно розпізнавати позитивні класи та відхиляти негативні класи при зміні порогового значення. Вона дозволяє враховувати компроміс між чутливістю та специфічністю класифікатора та зробити розгляд результатів моделі класифікації більш об'єктивним.

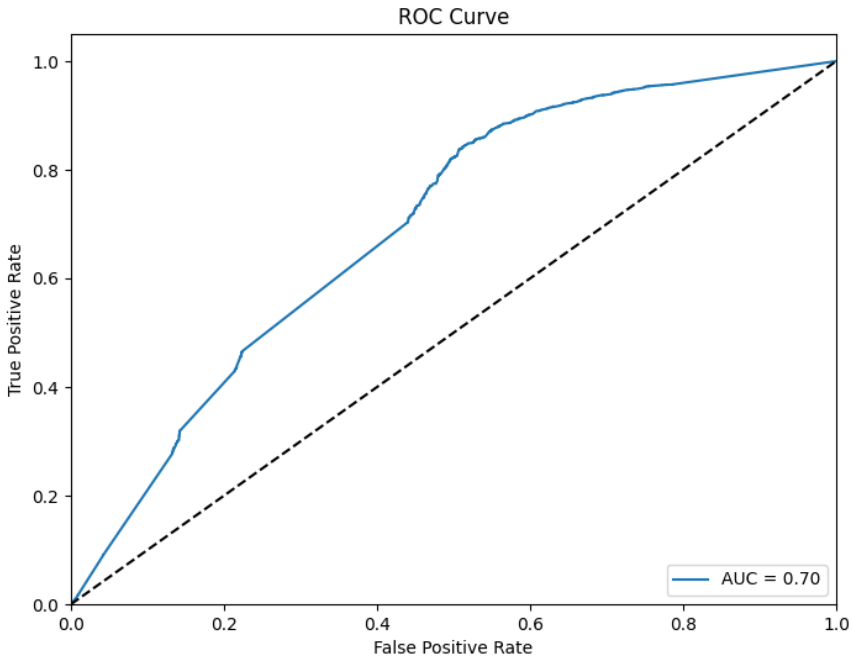


Рис. 3. Графік ROC-кривої логістичної регресії

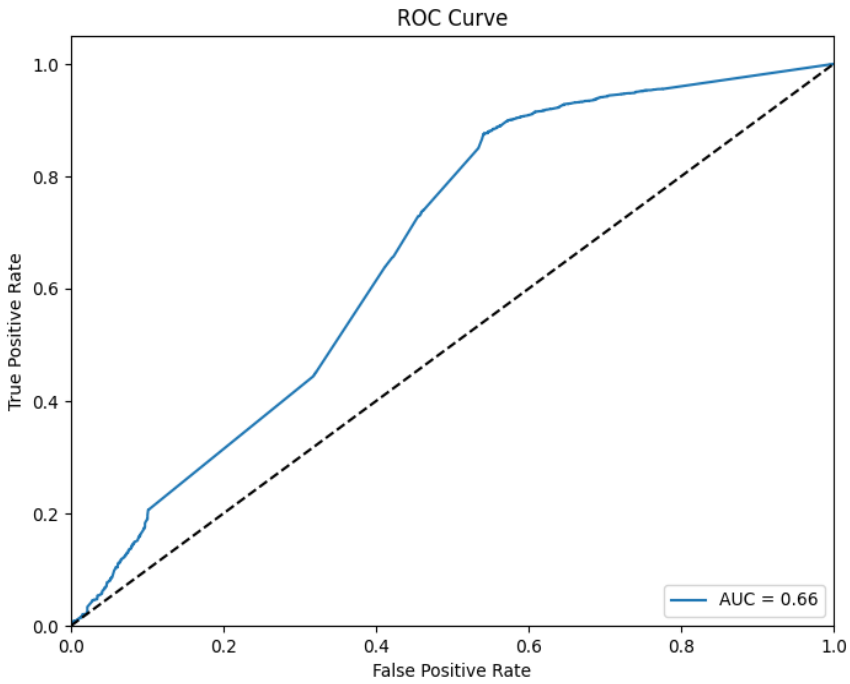


Рис. 4. Графік ROC-кривої методу опорних векторів

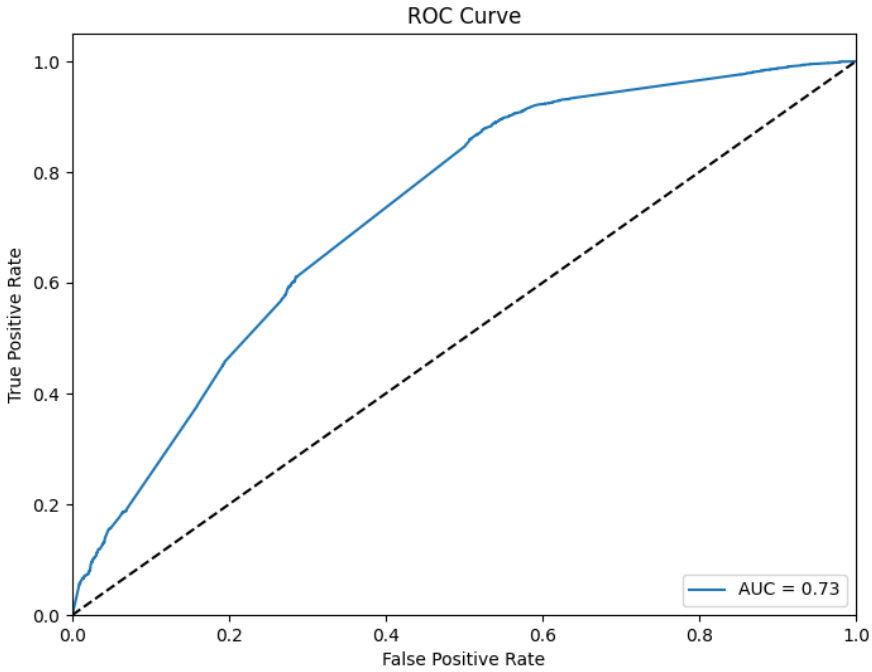


Рис. 5. Графік ROC-кривої випадкового лісу

Після проведення розрахунків визначено, що метод випадкового лісу найкраще виконує прогнозування успішності на вхідних даних і має більшу точність. Отримане значення точності 80% та AUC 73% свідчить про якість моделі класифікації, та гарну дискримінаційну силу моделі. По графіку ROC-кривої для методу випадкового лісу видно, що він має чітко виражену ділянку під кривою яка більше вигнута вгору і вліво, чим показує ефективність моделі. Даний результат є досить доброю початковою точкою, але в процесі подальшого дослідження може знадобитися додаткове вдосконалення для підвищення даних показників. Також важливим є збільшення кількості даних, а також додавання нових ознак для отримання більш точного результату. Отримання додаткових ознак можливе шляхом створення додаткових плагінів та ПЗ для Moodle.

Випадковий ліс є ансамблевим методом, який зазвичай працює краще в тих випадках, коли взаємозв'язки між ознаками та вихідними класами більш складні, нелінійні або коли є багато ознак. Він може автоматично враховувати важливість ознак і робити кращі передбачення, ніж лінійні моделі на складних даних.

Висновки

1. У роботі проведено перевірку точності прогнозування успішності студентів відносно їх відвідуваності з використанням трьох алогоритмів машинного навчання: Logistic Regression, SVM, Random Forest.
2. Побудовано матриці помилок та проведено розрахунок значень, що характеризують точність класифікації, а саме: чутливість, точність, специфічність, збалансовану точність.
3. Визначено, що метод випадкового лісу найкраще виконує прогнозування успішності на вхідних даних і має більшу точність, яка становить 80%.
4. Побудовано графіки ROC-кривої для визначення моделі, що краще справляється з завданням класифікації. Визначено, що метод випадкового лісу має найвище значення AUC, для поточних даних, яке становить 73%
5. Підвищення точності прогнозування можливе за рахунок розширення вхідних ознак, що потребує створення відповідних додатків (плагінів) для платформи Moodle, та є перспективним напрямом розвитку таких систем.

Література

1. Sen, P. C., Hajra, M., & Ghosh, M. (2020). Supervised classification algorithms in machine learning: A survey and review. In Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018 (pp. 99-111). Springer Singapore.

2. Прогнозування активності користувачів платформи Moodle на базі методів машинного навчання / В. І. Пилипенко, В. В. Стаценко. // Вісник Хмельницького національного університету. – 2023. – №4. – С. 257–261.

3. Оцінювання ефективності моделі прогнозування успішності методами машинного навчання / В. В. Стаценко, В. І. Пилипенко. // Вісник Хмельницького національного університету. – 2024. – №1. – С. 271–276.

4. Стаценко В.В. Оцінка ефективності моделі прогнозування активності користувачів Moodle методами машинного навчання. / Стаценко В.В., Пилипенко В.І., // VII Міжнародна науково-практична конференція «Мехатронні системи: інновації та інжиніринг» – «MSIE – 2023», 23 листопада 2023 рік, КНУТД. с. 28-29.

5. Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. Smart Learning Environments, 9(1), 11.

6. Vijayalakshmi, V., & Venkatachalapathy, K. (2019). Comparison of predicting student's performance using machine learning algorithms. International Journal of Intelligent Systems and Applications, 11(12), 34.

7. Das, A. (2024). Logistic regression. In Encyclopedia of Quality of Life and Well-Being Research (pp. 3985-3986). Cham: Springer International Publishing.

8. Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In Machine learning (pp. 101-121). Academic Press.

9. Genuer, R., Poggi, J. M., Genuer, R., & Poggi, J. M. (2020). Random forests (pp. 33-55). Springer International Publishing.

10. Model underfitting vs. overfitting [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://scikit-learn.org/stable/auto_examples/model_selection/plot_underfitting_overfitting.html.

11. Random Forest Classifier in scikit-learn [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

12. Logistic Regression in scikit-learn [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

13. SVC in scikit-learn [Електронний ресурс]. – 2024. – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

14. Krstinic, Damir & Braović, Maja & Šerić, Ljiljana & Božić-Štulić, Dunja. (2020). Multi-label Classifier Performance Evaluation with Confusion Matrix. 01-14. 10.5121/csit.2020.100801.

15. Bowers, A. J., & Zhou, X. (2019). Receiver operating characteristic (ROC) area under the curve (AUC): A diagnostic measure for evaluating the accuracy of predictors of education outcomes. Journal of Education for Students Placed at Risk (JESPAR), 24(1), 20-46.

References

1. Sen, P. C., Hajra, M., & Ghosh, M. (2020). Supervised classification algorithms in machine learning: A survey and review. In Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018 (pp. 99-111). Springer Singapore.

2. Prohnozuvannia aktyvnosti korystuvachiv platformy Moodle na bazi metodiv mashynnoho navchannia / V. I. Pylypenko, V. V. Statsenko. // Visnyk Khmelnytskoho natsionalnoho universytetu. – 2023. – №4. – S. 257–261.

3. Otsiniuvannia efektyvnosti modeli prohnozuvannia uspishnosti metodamy mashynnoho navchannia / V. V. Statsenko, V. I. Pylypenko. // Visnyk Khmelnytskoho natsionalnoho universytetu. – 2024. – №1. – S. 271–276.

4. Statsenko V.V. Otsinka efektyvnosti modeli prohnozuvannia aktyvnosti korystuvachiv Moodle metodamy mashynnoho navchannia. / Statsenko V.V., Pylypenko V.I., // VII Mizhnarodna naukovo-praktychna konferentsiia «Mekhatronni systemy: innovatsii ta inzhynirynh» – «MSIE – 2023», 23 lystopada 2023 rik, KNUTD. c. 28-29.

5. Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. Smart Learning Environments, 9(1), 11.

6. Vijayalakshmi, V., & Venkatachalapathy, K. (2019). Comparison of predicting student's performance using machine learning algorithms. International Journal of Intelligent Systems and Applications, 11(12), 34.

7. Das, A. (2024). Logistic regression. In Encyclopedia of Quality of Life and Well-Being Research (pp. 3985-3986). Cham: Springer International Publishing.

8. Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In Machine learning (pp. 101-121). Academic Press.

9. Genuer, R., Poggi, J. M., Genuer, R., & Poggi, J. M. (2020). Random forests (pp. 33-55). Springer International Publishing.

10. Model underfitting vs. overfitting [Elektronnyi resurs]. – 2024. – Rezhym dostupu do resursu: https://scikit-learn.org/stable/auto_examples/model_selection/plot_underfitting_overfitting.html.

11. Random Forest Classifier in scikit-learn [Elektronnyi resurs]. – 2024. – Rezhym dostupu do resursu: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

12. Logistic Regression in scikit-learn [Elektronnyi resurs]. – 2024. – Rezhym dostupu do resursu: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

13. SVC in scikit-learn [Elektronnyi resurs]. – 2024. – Rezhym dostupu do resursu: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

14. Krstinic, Damir & Braović, Maja & Šerić, Ljiljana & Božić-Štulić, Dunja. (2020). Multi-label Classifier Performance Evaluation with Confusion Matrix. 01-14. 10.5121/csit.2020.100801.

15. Bowers, A. J., & Zhou, X. (2019). Receiver operating characteristic (ROC) area under the curve (AUC): A diagnostic measure for evaluating the accuracy of predictors of education outcomes. Journal of Education for Students Placed at Risk (JESPAR), 24(1), 20-46.