https://doi.org/10.31891/2219-9365-2025-81-58
UDC 004.081.345

ANTONENKO Artem
National University of Life and Environmental Sciences of Ukraine
https://orcid.org/0000-0001-9397-1209
SOLSKYI Danyil
State University of Information and Communication Technologies
https://orcid.org/0009-0005-0351-5987
SOLOBAIEV Serhii
State University of Information and Communication Technologies
https://orcid.org/0009-0008-6298-4777
CHECHYK Serhii
State University of Information and Communication Technologies
https://orcid.org/0009-0009-9293-5156
CHEREVYK Oleksii
State University of Information and Communication Technologies
https://orcid.org/0009-0006-0347-7607

USING THE SCIKIT-LEARN LIBRARY IN MACHINE LEARNING CLASSIFICATION METHODS

The article analyzes the selection of an appropriate machine learning algorithm, which depends on several factors, such as the amount of data, its quality and diversity, as well as the awareness of the business goals that need to be achieved from this data. Machine learning in scikit-learn is all about importing the right modules and running the model fitting method. The object of the study is the application of various classification algorithms for grouping the results of machine learning models in cases of binary and multi-class classification. Therefore, it is necessary to test different algorithms, evaluating their effectiveness on test data sets, and choose the best one. In this regard, it is important to select algorithms that are most suitable for the given task. The authors of the paper focused on accuracy, training time, and data characteristics. Therefore, the choice of the optimal algorithm is a combination of business requirements, technical specifications, experimental activities and taking into account the available time. The implementation of machine learning methods in various fields was also investigated. The machine learning process is described, which includes the following stages: data preparation, training set creation, classifier development, classifier training, forecasting, classifier performance evaluation, and parameter setting. An analysis of the application of various classification algorithms was carried out using the Scikit-learn library with Python. An analysis of the method of model selection, calculation, formatting and data preparation was also carried out, as well as the selection of optimal input values and models. Several approaches to classifier evaluation are evaluated. The main goal of the work is to study the library to assess its practical effectiveness. Classification methods in machine learning using Scikit-Learn are described. A comparison of different classification methods was made using the Scikit-learn library for machine learning models.

Keywords: optimization, forecasting, machine learning, classification, data analysis.

АНТОНЕНКО Артем
Національний університет біоресурсів і природокористування України
СОЛЬСЬКИЙ Даниїл, СОЛОБАЄВ Сергій, ЧЕЧИК Сергій, ЧЕРЕВИК Олексій
Державний університет інформаційно-комунікаційних технологій

ВИКОРИСТАННЯ БІБЛІОТЕКИ SCIKIT-LEARN У МЕТОДАХ КЛАСИФІКАЦІЇ МАШИННОГО НАВЧАННЯ

У статті аналізується вибір відповідного алгоритму машинного навчання, який залежить від кількох факторів, таких як кількість даних, їх якість і різноманітність, а також усвідомлення бізнес-цілей, яких потрібно досягти на основі цих даних. Машинне навчання в scikit-learn полягає в імпорті правильних модулів і виконанні методу підгонки моделі. Об'єктом дослідження є застосування різноманітних алгоритмів класифікації для групування результатів моделей машинного навчання у випадках бінарної та багатокласової класифікації. Тому необхідно протестувати різні алгоритми, оцінивши їх ефективність на тестових наборах даних, і вибрати найкращий. У зв'язку з цим важливо вибрати алгоритми, які найбільше підходять для поставленої задачі. Автори статті зосередилися на точності, часу навчання та характеристиках даних. Тому вибір оптимального алгоритму – це поєднання вимог бізнесу, технічних характеристик, експериментальної діяльності та врахування наявного часу. Також було досліджено реалізацію методів машинного навчання в різних сферах. Описано процес машинного навчання, який включає наступні етапи: підготовка даних, створення навчальної множини, розробка класифікатора, навчання класифікатора, прогнозування, оцінка продуктивності класифікатора та налаштування параметрів. Аналіз застосування різних алгоритмів класифікації проводився за допомогою бібліотеки Scikit-learn з Python. Також проведено аналіз методу вибору моделі, розрахунку, форматування та підготовки даних, а також вибір оптимальних вхідних значень і моделей. Оцінено кілька підходів до оцінки класифікатора. Основною метою роботи є дослідження бібліотеки для оцінки її практичної ефективності. Описано методи класифікації в машинному навчанні за допомогою Scikit-Learn. Порівняння різних методів класифікації було зроблено за допомогою бібліотеки Scikit-learn для моделей машинного навчання.

Ключові слова: оптимізація, прогнозування, машинне навчання, класифікація, аналіз даних.

STATEMENT OF THE PROBLEM IN A GENERAL FORM AND ITS CONNECTION WITH IMPORTANT SCIENTIFIC OR PRACTICAL TASKS

Machine learning technology based on data analysis dates back to 1950, when the first drafts programs were developed. Over the past decades, the general principle has not changed. However, thanks to the explosive growth of computing power of computers, the patterns and forecasts created by them have become many times more complicated, and the range of problems and tasks that can be solved using machine learning has expanded. In recent years, a large number of studies have been conducted demonstrating the implementation of machine learning methods in various fields [2].

ANALYSIS OF RECENT SOURCES

Machine learning is mostly used to solve problems that are too complex and require adaptation. That is, this is a class of tasks that cannot be solved by a certain clear algorithm, it is necessary to take into account the already obtained results. The analysis of literary sources on the application of these methods is limited to a small amount of information.

The purpose of the article is to analyze the use of various classification algorithms, the use of the method of model selection, calculation, formatting and data preparation with the help of the Python library, Scikit-learn; choose optimal input values and models; library research for the effectiveness of its practical application.

FORMULATION OF THE PROBLEM

There are many classification methods that use different mathematical apparatus and different approaches during implementation [1]. However, the effectiveness of these methods depends on the specific task to be solved. Despite the fact that commercial companies have been dealing with the problem of improving the quality of machine learning for the last decade, there are currently no methods that could unambiguously and effectively solve the task of classification. Therefore, it is necessary to analyze the application of different classification algorithms using the Scikit-Learn library.

PRESENTING MAIN MATERIAL

Thanks to machine learning, the programmer does not have to write instructions that take into account all possible problems and contain all solutions. Instead, a computer (or a separate program) embeds an algorithm to independently find solutions through the complex use of statistical data, from which patterns are derived and on the basis of which forecasts are made. Currently, machine learning allows computers to learn to recognize not only people in photos and drawings, but also landscapes, objects, text and numbers. When it comes to text, machine learning is also indispensable here: the grammar check function is now available in any text editor and even in phones. Moreover, not only the writing of words is taken into account, but also the context, shades of meaning and other subtle linguistic aspects. Moreover, there is already software capable of writing news articles (on the topic of economics or, for example, sports) without human intervention. Types of machine learning tasks. All tasks that are solved with the help of machine learning belong to this category (Table 1).

Table 1

Task categories

Task	Appointment
Regressions	Forecasting based on a sample of objects with different characteristics. The output should be a real number (2, 35, 76,454, etc.). For example, the price of an apartment, the value of a security after six months, the expected income of the store for the next month, the quality of wine in blind testing.
Classifications	Obtaining a categorical answer based on a set of features. Has a finite number of answers (usually in yes or no format): is there a cat in the photo, is the image a human face, is the patient sick with cancer. It is used in marketing when assessing the creditworthiness of borrowers, determining customer loyalty, pattern recognition, medical diagnostics and in many other areas.
Clustering	Distribution of data into groups: division of all customers of the mobile operator according to the level of payment capacity, inclusion of space objects in one or another category (planet, star, black hole, etc.).
Dimension reduction	Reducing a large number of features to a smaller number (usually 2-3) for the convenience of their further visualization (for example, data compression).
Detection of anomalies	Detection of anomalies from standard cases. In practice, such a task is, for example, detection of fraudulent actions with bank cards.

Example of use. The task of classification and regression is a task of learning with a teacher. As an example, we will present the task of credit scoring: on the basis of the data accumulated by the credit organization about its customers, it is possible to predict the non-repayment of the loan. Here, for the algorithm, the experience E is the available training sample: a set of objects (people), each characterized by a set of features (such as age, salary, loan type, past defaults, etc.) and a target feature. If the target feature is simply the fact of non-repayment of the loan (1 or 0, i.e. the bank knows about its clients who has returned the loan and who has not), then this is a task of (binary) classification. If it is known how long the client delayed in returning the loan, one would like to predict the same for new clients, then this will be a regression task [2].

A classifier is a system that inputs (as a rule) a vector of discrete and/or continuous functions and outputs one discrete class value.

For example, a spam filter classifies e-mail messages as "spam" or "not spam" and its input can be a vector of boolean values x = (x1, ..., xj ,..., xd), where xj = 1, if the jth word in the dictionary appears in the email, and xj = 0 otherwise. The learner enters a training set of examples (xi ,yi), where xi = (xi ,1, ..., xi ,d) is the observed input, yi is the corresponding output, and outputs the classifier. The student's test is whether this classifier produces the correct output yt for future examples xt (for example, whether a spam filter correctly classifies previously unseen e-mails as spam or not spam). Machine learning is performed using various algorithms, but for all algorithms, 3 components are the most important: – Representation. The classifier must be represented using a formal language that a computer can process. Conversely, choosing a representation for the learner is equivalent to choosing a set of classifiers that he can learn. This set is called the student's hypothesis space. If the classifier is not in the hypothesis space, then it cannot be studied. - Evaluation. An evaluation function (also called objective function) or scoring function) is needed to separate good classifiers from bad ones. The evaluation function used inside the algorithm can be different from the external one we want to optimize for the classifier, for simplicity of optimization. – Optimization. Finally, we need a method to find among the classifiers the one that will classify the fastest and most correctly. The choice of optimization method is a key element of the learner's performance and also helps to determine the classifier of choice if the evaluation function has more than one optimum. For new learners, it is best to start using common optimizers, which are later replaced by specially designed ones. [4]

Classification is a very large part of the field, including statistics and machine learning. As a rule, it can be divided into 2 parts:

1. Binary classification - grouping of the result into one of two groups.
2. Multi-class classification – grouping the result into one of several (more than two) groups.

Classification methods in machine learning using the Scikit-Learn library.

There are many libraries written in Python for machine learning. Let's consider one of the most popular - Scikit-Learn. What is Scikit-Learn? Scikit-Learn is a Python library first developed by David Cournapeau in 2007. This library contains a large number of algorithms for tasks related to classification and machine learning in general. Scikit-Learn is based on the SciPy library, which must be installed before starting. Scikit-Learn simplifies the process of creating a classifier and helps to highlight machine learning concepts more clearly by implementing them with a clear, well-documented and reliable library. It contains a number of techniques covering everything you might need in data analytics: classification and regression algorithms, clustering, validation, and model selection. It can also be used to reduce the dimensionality of data and highlight features (Fig. 1).

Machine learning systems have inputs and outputs. What is submitted to the entrance is called signs. When features are fed to the inputs of a machine learning system, this system tries to find a match, to notice a pattern between the features. The result of this work is generated at the output [3]. This result is usually called a label, because the outputs have some mark that is issued by the system, that is, a prediction of which category the output falls into after classification. Scikit-Learn provides access to various classification algorithms. The main ones are:

– the method of k-nearest neighbors (K-Nearest Neighbors);
– method of support vectors (Support Vector Machines);
– decision tree classifier (Decision Tree Classifier) / random forest (Random Forests);
– naive Bayesian method (Naive Bayes); – linear discriminant analysis (Linear Discriminant Analysis);
– logical regression (Logistic Regression); Examples of classification tasks A classification task is any task where it is necessary to determine the type of an object from two or more existing classes. Such tasks can be different: identifying a cat in the image or a dog, or determining the quality of wine based on its acidity and alcohol content. Depending on the classification task, different types of classifiers can be used. For example, if the classification contains some binary logic, then logistic regression is best suited to it. The process of machine learning The process includes the following steps: data preparation, training set creation, classifier creation, classifier training, forecasting, classifier performance evaluation, and parameter tuning. First, you need to prepare the data set for the classifier: transform the data into a form that is correct for classification and process any anomalies in the data. Missing values in the data, or any other outliers, all need to be handled or they can negatively impact the performance of the classifier. This stage is called data preprocessing. The next step is to divide the data into training and test sets.

ScikitLearn has an excellent traintestsplit function for this. As already mentioned above, the classifier must be created and trained on the training data set. After these steps, the model can already make predictions. By comparing the testimony of the classifier with the actually known data, it is possible to draw a conclusion about the accuracy of the classifier. Most likely, it will be necessary to "adjust" the parameters of the classifier until the desired accuracy is achieved, since it is unlikely that the classifier will meet all the requirements from the first run [4].

Evaluation of the classifier: there are several evaluation options: – Classification accuracy. Classification accuracy is the easiest to measure, and therefore this parameter is most often used. The accuracy value is the number of correct predictions divided by the number of all predictions or, simply put, the ratio of correct predictions to all. Although this metric can quickly give a clear idea of a classifier's performance, it is best used when each class has at least roughly the same number of examples. Since this will rarely happen, it is recommended to use other classification

indicators. – Logarithmic losses. The logarithmic loss value (logloss) shows how confident the classifier is in its prediction. Logloss returns the probability that an object belongs to a particular class, summing them to give an overall idea of the "confidence" of the classifier.

```
In [21]: from sklearn.ensemble import RandomForestClassifier

In [22]: x = known_values[['free', 'super', 'source']]
         y = known_values['phone_type']

In [23]: model = RandomForestClassifier(n_estimators = 100)
         model = model.fit(x, y)

In [25]: sample_user = [1, 6, 0]
         model.predict([sample_user])
```

Fig. 1. Simple classification using the "random forest" model in Scikit-learn

This indicator ranges from 0 to 1 - "not at all sure" and "completely sure", respectively. Logloss drops heavily when the classifier is highly "confident" of an incorrect answer. – ROC-curve area (AUC). This indicator is used only for binary classification. The area under the ROC curve is the ability of the classifier to distinguish between suitable and unsuitable objects for any class. A value of 1.0: the entire area that falls under the curve represents a perfect classifier. Therefore, 0.5 means that the accuracy of the classifier corresponds to chance. The curve is calculated taking into account the accuracy and specificity of the model. You can read more about the calculations here. – Matrix of inaccuracies. The Confusion Matrix is a table or diagram that shows the accuracy of a classifier's prediction for two or more classes. The predictions of the classifier are on the X-axis, and the result (accuracy) is on the Y-axis. As you gain experience, it will become easier to choose the appropriate classifier type. However, it is a good practice to implement several suitable classifiers and choose the most optimal and productive one.

Scikit-learn compiles a wide range of machine learning algorithms, both supervised and unsupervised, using a consistent, task-oriented interface that makes it easy to compare methods for this application. Because it relies on Python's scientific ecosystem, it can be easily integrated into an application outside the traditional scope of statistical data analysis. It is important to note that an algorithm implemented in a high-level language can be used as building blocks for use-specific approaches, such as in medical imaging (Michel et al., 2011). Future work includes online learning, scaling to large datasets.

CONCLUSIONS FROM THIS STUDY AND PROSPECTS FOR FURTHER RESEARCH IN THIS DIRECTION

Machine learning in scikit-learn is all about importing the right modules and running the model fitting method. It is more difficult to clean, format, and prepare the data, as well as to select optimal inputs and models. Therefore, before starting scikitlearn, you need, firstly, to practice Python and Pandas skills in order to learn how to prepare data qualitatively, and secondly, to master the theory and mathematical basis of various prediction and classification models in order to understand what is happening with the data during their use.

References

1. Rao C., Govindaraju V. (2013). Handbook of Statistics: Machine Learning: Theory and Applications, 552
2. Machine learning with Python and Scikit-Learn. URL: https://habr.com/ru/company/ mlclass/blog/247751/ (Last accessed: 17.10.2024).
3. Classification in Python with Scikit-Learn and Pandas. URL: https://stackabuse.com/classification-in-pythonwith-scikit-learn-and-pandas/ (Last accessed: 17.10.2024).
4. An overview of classification methods in machine learning using Scikit-Learn. URL: https://tproger. ru/translations/scikit-learn-in-python/s://stackabuse.com/classification-in-python-with-scikit-learn-and-pandas/ (Last accessed:: 10.05.2024).
5. Aksak, N.H. (2019). Metody ta modeli rozpodilenoi intelektualnoi obrobky velykykh danykh u spetsializovanykh kompiuternykh systemakh [Methods and models of distributed intelligent processing of large data in specialized computer systems] abstract of the dissertation of the doctor of technical sciences: 05.13.05 “Computer systems and components”. Ministry of Education and Science of Ukraine, Kharkiv. Nat. University of Radio Electronics. Kharkiv, 44 p. [in Ukrainian].
6. Tverdokhlib A.O., Korotin D.S. Efektyvnist funktsionuvannia kompiuternykh system pry vykorystanni tekhnolohii blokchein i baz dannykh. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2022, (6) [in Ukrainian].
7. Tsvyk O.S. Analiz i osoblyvosti prohramnoho zabezpechennia dlia kontroliu trafiku. Visnyk Khmelnytskoho natsionalnoho universytetu. Ceriia: Tekhnichni nauky, 2023, (1) [in Ukrainian].
8. Novichenko Ye.O. Aktualni zasady stvorennia alhorytmiv obrobky informatsii dlia lohistychnykh tsentriv. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2023 (1) [in Ukrainian].
9. Zaitsev Ye.O. Smart zasoby vyznachennia avariinykh staniv u rozpodilnykh elektrychnykh merezhakh mist. Tavriiskyi naukovyi visnyk. Seriia: Tekhnichni nauky, 2022, (5) [in Ukrainian].
10. Cui, D.D., & Liu, F. (2012). The application of BP neural network in Internet of Things. In Advanced Engineering Forum. Vol. 6, pp. 1098–1102. Trans Tech Publications Ltd.

11. Xiao, H., & Li, Y. (2011). A New Thought based on the Service Composition of Automatic Transmission Semantic Grid in Internet of Things. IJACT: International Journal of Advancements in Computing Technology, 3(7), 10–16. DOI: 10.4156/ijact.vol3.issue7.2.

12. Xiang, C., & Zhou, Z. (2011). A new music classification method based on BP neural network. International Journal of Digital Content Technology and its Applications, 5(6), 85–94.

13. Xiao, H., & Li, Y. (2011). A New Thought based on the Service Composition of Automatic Transmission Semantic Grid in Internet of Things. IJACT: International Journal of Advancements in Computing Technology, 3(7), 10–16.

14. Fang, R., Jian-feng, M., & Xuan-wcn, H. (2012). Attribute-based access control scheme for the perceptive layer of the Internet of Things. Journal of Xidian University, 39(2), 66–72.

15. Naveen Dr, Raina Rohini. Machine learning in Internet of Thing. (2018). Retrieved from: https://www.researchgate.net/publication/322209934_MACHINE_LEARNING_IN_INTERNET_OF_THING (Last accessed: 10 January 2021).

16. Xinwu, L. (2011). A new color correction model for based on BP neural network. Advances in Information Sciences and Service Sciences, 3(5), 72–78.