

7. Zhilenkov A.A., Cherny S.G. Use of artificial intelligence technologies in the design and operation of phased array systems for GIS. *Oboronnyj kompleks – nauchno-tekhnicheskomu progressu Rossii*, 2020, no. 2 (146), pp. 57–63. (In Russ.)
8. Ivanov Yu.S. Algorithms for recognizing moving objects for intelligent video surveillance systems: diss. ... cand. tech. science. Khabarovsk, 2015. 155 p. (In Russ.)
9. Vasiliev S.A. et al. Study of complex images of screw propulsors Regional means with a mechatronic profilograph. *Obrabotka metallov (tekhnologiya, oborudovanie, instrumenty)*, 2021, vol. 23, no. 4, pp. 65–78. (In Russ.)
10. Kuzmin A.N. et al. Innovative technology of acoustic emission inspection of linear-type facilities of main pipelines. *V mire nerazrushayushchego kontrolya*, 2020, vol. 23, no. 1, pp. 46–53. (In Russ.)
11. Magamedova D.M. OpenCV is a computer vision tool. *Tendencii razvitiya nauki i obrazovaniya*, 2020, no. 63-3, pp. 42–48. (In Russ.)
12. Svirsky S.N. Detection algorithms of simple geometric shapes. *Innovacionnye tekhnologii: teoriya, instrumenty, praktika*, 2014, vol. 2, pp. 311–315. (In Russ.)
13. Solovyov N.V., Sergeev A.M. *Improving the quality of raster images: Textbook*. Saint Petersburg: SPbGU ITMO, 2010, 158 p. (In Russ.)
14. Yusupov R.M., Yusupov Yu.V. Informatics state of the art and development prospective. *Trudy SPIIRAN*, 2007, no. 5, pp. 10–46. (In Russ.)
15. Sevbitov A.V. et al. Analysis of electronic microscopy results based on combining the infiltration method with different restoration technologies and in vitro investigation of enamel focal demineralization treatment at the defect stage. *Periodico Tche Quimica*, 2019, vol. 16, no. 33, pp. 53–59.

Received 30.11.2023

УДК 004.89

КОМБИНИРОВАННЫЙ МЕТОД НАСТРОЙКИ ГИПЕРПАРАМЕТРОВ МАТЕМАТИЧЕСКОЙ МОДЕЛИ

Пальмов С.В., Диязитдинова А.А.

Поволжский государственный университет телекоммуникаций и информатики, Самара, РФ

E-mail: psvzo@yandex.ru, a.diyazitdinova@psuti.ru

Автоматизация процессов обработки данных является важным направлением в области информационных технологий. Основное внимание исследователей, как правило, сосредоточено на обучении интеллектуальных систем. Одним из ключевых аспектов упомянутого процесса является подбор гиперпараметров моделей. В работе рассмотрен комбинированный метод настройки гиперпараметров классификационной математической модели. Он совмещает в себе возможности двух известных подходов: полного и ограниченного перебора. Сначала для поиска предварительной оценки максимума значения метрики качества модели применяется первый из них. Далее посредством второго создается окончательная оценка достижимого качества и формируется список сочетаний значений гиперпараметров, максимизирующих эффективность классификатора. Проверка метода осуществлена посредством авторского программного обеспечения, созданного с применением алгоритма стохастического градиентного спуска. Полученные результаты свидетельствуют об эффективности предложенного метода.

Ключевые слова: гиперпараметр, математическое моделирование, grid search, randomized search, градиентный спуск, машинное обучение

Введение

Построение математических моделей в настоящее время является распространенной задачей, с которой сталкиваются исследователи в различных областях. Такая ситуация вызвана значительным интересом к методам машинного обучения. Создание искусственного разума пока не представляется возможным, однако системы, в той или иной мере имитирующие «разумное» поведение, получили широкое распространение, темпы

которого продолжают возрастать; требуется решать все более трудные задачи. Последний факт значит, что сложность математических моделей, лежащих в основе создаваемых «интеллектуальных» продуктов, также увеличивается. Одна из характеристик сложности модели – число ее гиперпараметров [10]. «Ручной» подбор оптимальной конфигурации малоэффективен [9]. Существует две группы методов для автоматизации указанного процесса: полный и ограниченный

перебор. Каждый из них обладает определенными достоинствами и недостатками.

Существуют реализации как методов первой группы (например, «Grid Search», GS) [2], так и второй (например, «Randomized Search», RS) [3]. Кроме этого, разработаны альтернативные решения (например, «Helena4»).

Алгоритм «Grid Search» получил свое название из-за способа, которым выполняется обход пространства поиска. Создается «сетка» (grid) доступных значений для каждого гиперпараметра и затем перебираются их комбинации. Таким образом, исследуются все участки сетки в поисках оптимального сочетания.

GS гарантирует обнаружение такого сочетания значений, которое в заданных условиях будет максимизировать качество математической модели. К недостаткам относится низкая скорость работы.

RS, в отличие от GS, выбирает случайные комбинации на каждой итерации, что позволяет заметно снизить время, затрачиваемое на поиск решения, но ценой возможности появления сильной корреляции между качеством получаемых результатов и числом итераций.

Основная идея алгоритма «Helena4» состоит в поэтапном уменьшении объема пространства, в котором осуществляется поиск максимума функции. При этом уменьшение объема пространства осуществляется не за счет уменьшения количества измерений, а посредством сокращения диапазона, в котором ищется максимум. Этот диапазон сокращается по каждому измерению. С течением времени объем пространства, на котором алгоритм ищет максимум, сводится к небольшой величине, в котором и находится максимум функции [1].

Достоинства: достаточно высокое качество получаемых результатов (со слов автора). Недостатки: сложность алгоритма.

Предлагаемый метод

Общий недостаток перечисленных алгоритмов заключается в том, что они осуществляют поиск одного сочетания значений гиперпараметров, максимизирующего качество модели, но таковых может быть более одного.

Определенные значения некоторого гиперпараметра могут по-разному влиять на скорость построения модели (при сохранении максимального качества последней). Таким образом, появится возможность выбрать вариант, удовлетворяющий паре критериев: $q|t \rightarrow \max$, $t \rightarrow \min$, а не только первому ($q|t$ – качество модели, t – время ее формирования). Например, гиперпараметр h_1 может принимать два значения: «1» и «2». Пусть

максимальные значения метрики качества (вероятность правильного распознавания объекта), формируемые на основе h_1 моделей, будут равными: $m_{h_1=1} = m_{h_1=2}$, а время, затрачиваемое на обучение, – нет: $t_{h_1=1} = t_{h_1=2}$. Вывод очевиден: необходимо использовать значение «2».

Предлагаемый метод позволяет достаточно быстро найти часть подобных сочетаний, поскольку использует сочетание GS и RS.

Также присутствует возможность задания числа повторных прогонов для одного набора значений гиперпараметров, следствием чего является повышение вероятности формирования более качественной оценки создаваемых моделей. Для некоторых алгоритмов доступна рандомизация начальных значений генератора случайных чисел. То есть в этом случае каждый новый прогон для фиксированного набора параметров будет порождать отличные модели. Указанное решение призвано снизить вероятность переобучения.

Обобщенно работу метода можно описать следующим образом:

1. Выбрать гиперпараметр (h^*) категориального типа, который имеет наибольшую вариативность значений, то есть: $h^* = \arg \max(h \in H) \cdot V(h)$, где H – множество гиперпараметров категориального типа, $V(h)$ – функция, возвращающая вариативность значений гиперпараметра h .

2. Выбрать одно из значений h^* и зафиксировать его. На остальные h такие ограничения не накладываются.

3. Выполнить GS в указанных в п. 2 рамках. При необходимости, если получено низкое значение m_{GS} (это решение принимает пользователь), возможно повторение прогона для другого значения h^* . Число повторных прогонов желательно установить равным единице или несколько превышающим данный порог.

4. Выполнить RS для всей сетки (без ограничений п. 1). Число повторных прогонов желательно установить превышающим единицу (задается пользователем, исходя из доступных времени / вычислительных ресурсов). Число итераций RS выбирается много меньшим числа итераций из п. 1. Возможно потребуются более одного выполнения п. 4.

5. Критерий останова сформулирован так:

$$\exists N_A : \forall N \geq N_A, m_{RS} \geq m_{GS},$$

где N_A – достаточное число сочетаний (определяется пользователем), N – общее число сочетаний, m_{GS} и m_{RS} – максимальные значения метрик качества для моделей, полученных при помощи GS и RS.

Программная реализация предложенного метода

Для демонстрации возможностей предложенного метода было создано программное обеспечение (ПО) на Python. Графический интерфейс представлен на рисунке 1. Использовались готовые реализации GS и RS (библиотека scikit-learn) [6]. Интерфейс описан в таблице 1.

Таблица 1. Элементы интерфейса ПО

Элемент	Описание
Параметры для GS и RS	
Число циклов проверок	Число повторных прогонов для одного набора значений гиперпараметров
Степень логирования	Уровень детализации сообщений, выводимых по ходу процесса подбора параметров
Число блоков	Число блоков (folds) перекрестной проверки
Предобработка	Размер кэша заданий для процессора
Число сочетаний	Число итераций для RS
Основное меню	
Загрузить данные (с обработкой / без обработки)	Входные данные могут быть загружены «как есть» или с дополнительной обработкой, заключающейся в кодировке строковых значений переменных
Загрузить настройки	Загрузка диапазонов значений гиперпараметров из конфигурационного файла
Автонастройка	Применение GS или RS (r)

В работе для формирования математических моделей используется алгоритм стохастического градиентного спуска [4]. Конфигурационный файл содержит соответствующие гиперпараметры [5] и диапазоны их значений. Пример указанной информации представлен в таблице 2.

Также предусмотрена возможность настройки размера очереди заданий для процессора, что способствует более полному использованию ресурсов компьютера.

Кроме этого, в ПО реализована предобработка входных данных (нормализация к интервалу [0, 1]) [8].

Алгоритм работы с созданным ПО:

1. Отредактировать конфигурационный файл (или создать его в случае отсутствия).
2. Запустить ПО.
3. Задать требуемые настройки GS.
4. Загрузить данные (одним из двух доступных способов).

5. Загрузить настройки.
6. Выполнить поиск посредством GS.
7. Выполнить поиск посредством RS.

Результаты выполнения п. 6 выводятся в консоль.

Таблица 2. Содержимое конфигурационного файла (пример)

Гиперпараметр	Значение
loss	hinge, log_loss, modified_huber, squared_hinge, perceptron, squared_error, huber, epsilon_insensitive, squared_epsilon_insensitive
penalty	l2, l1, elasticnet, None
alpha	0.0001
l1_ratio	0.15
fit_intercept	True, False
max_iter	1000
tol	0.001, None
shuffle	True, False
verbose	0
epsilon	0.1
n_jobs	-1
random_state	10
learning_rate	optimal, constant, invscaling, adaptive
eta0	0.00001
power_t	0.5
early_stopping	False, True
validation_fraction	0.1
n_iter_no_change	5
class_weight	balanced, None
warm_start	False, True
average	False, True

После каждого выполнения п. 7 автоматически формируется файл, в строках которого содержатся значения параметров каждой итерации, а также лучшее значение метрики качества для каждого сочетания. Изучив данный массив информации, пользователь получит перечень наборов значений гиперпараметров, способных породить модели максимального качества (или близким к нему).

Исследование возможностей разработанного ПО

Была проведена серия экспериментов, направленная на демонстрацию возможностей предлагаемого метода. В ее ходе производилось тестирование ПО при различных настройках. Ниже приведено краткое описание каждого экспери-

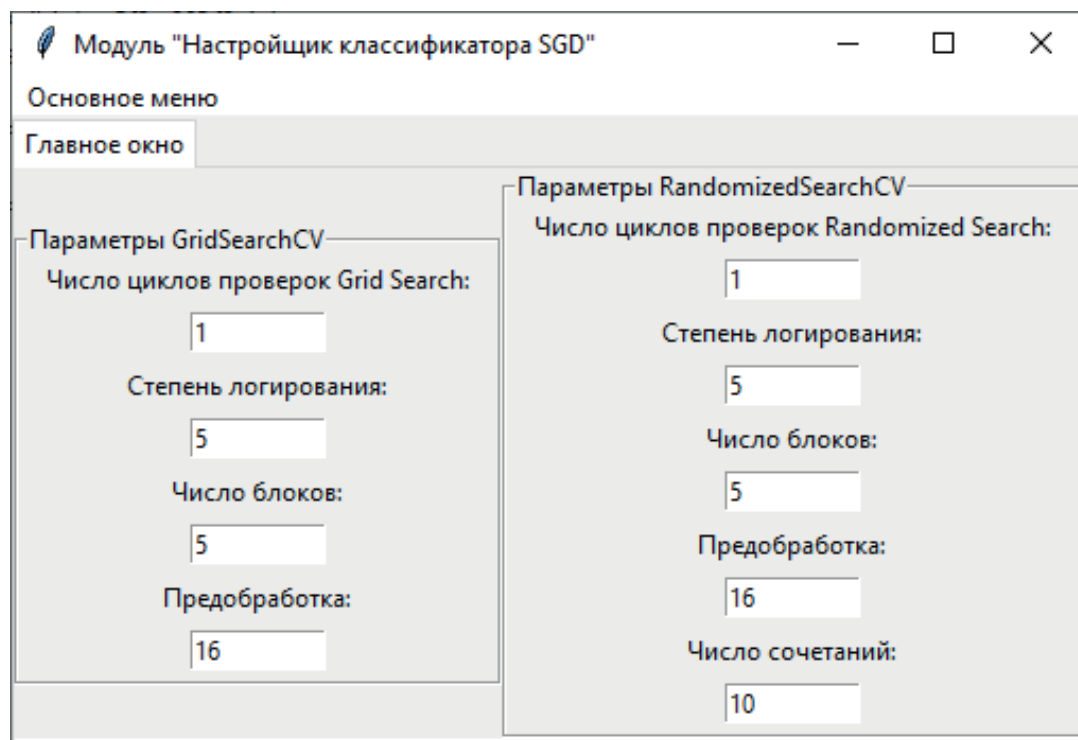


Рисунок 1. Главное окно ПО

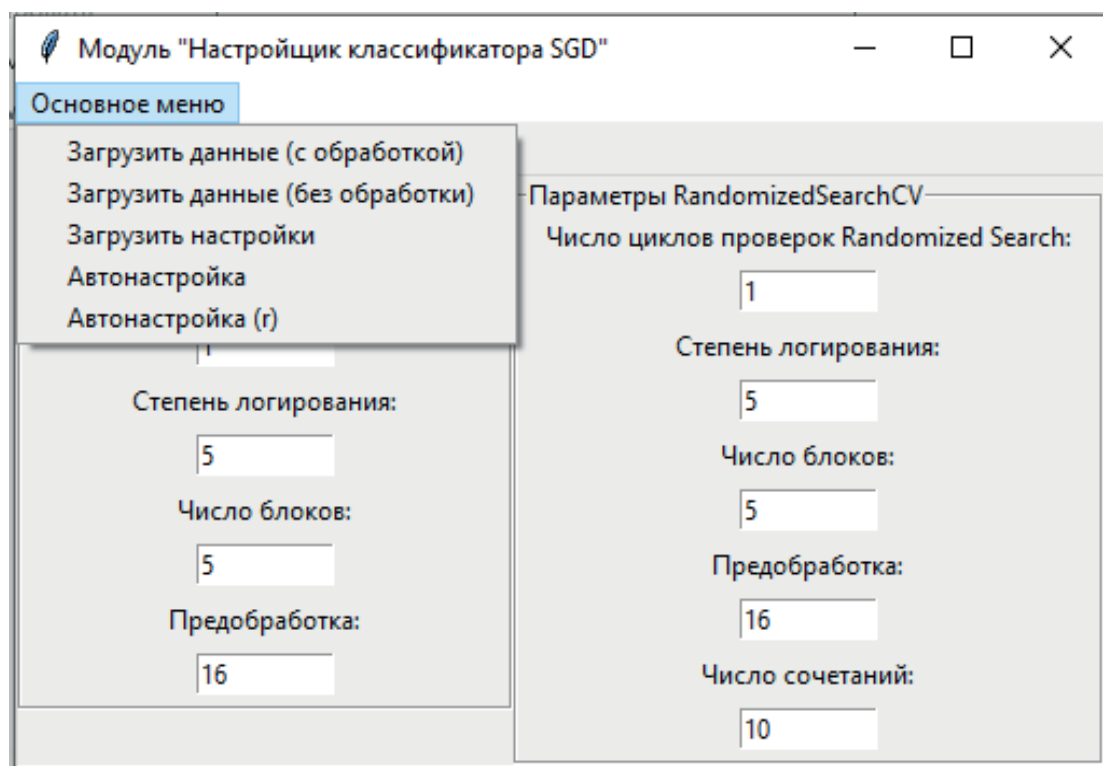


Рисунок 2. Основное меню ПО

мента и его входные данные. Результаты, полученные в ходе экспериментов, представлены в таблице 3.

Использовался модифицированный датасет (было исключено поле «time») из [7].

Эксперимент № 1. Заключался в выполнении пп. 1–3 обобщенного описания работы метода.

$h^* = \langle \text{loss} \rangle$, $h^*(\text{loss}) = \langle \text{hinge} \rangle$. Остальные настройки соответствуют данным в таблице 2 и рисунок 1 (секция «Параметры Grid Search»).

Эксперимент № 2. Заключался в выполнении пп. 4–5 обобщенного описания работы метода.

Число циклов проверок RS, шт.: 50. Остальные настройки соответствуют данным в таблица

2 и рисунок 1 (секция «Параметры Randomized Search»).

Во втором эксперименте были получены высокие результаты. Имея такой «запас прочности», попытаемся оценить, как скажется на качестве моделей ограничение диапазонов значений гиперпараметров.

Эксперимент № 3. Проверим, как повлияет на качество снижение значения гиперпараметра «maxiter» (максимальное количество итераций) с «1000» до «100»? Все остальные настройки эквивалентны № 2.

Эксперимент № 4. Проверим, как повлияет на качество снижение значения гиперпараметра «maxiter» с «1000» до «100» и выключение «фиксированной» рандомизации (random_state = «None»)? Все остальные настройки эквивалентны № 2.

Эксперимент № 5. Проверим, как повлияет на качество снижение значения гиперпараметра «maxiter» с «1000» до «100», выключение «фиксированной» рандомизации (random_state = «None») и уменьшение числа сочетаний до семи?

Эксперимент № 6. Проверим, как повлияет на качество снижение значения гиперпараметра «maxiter» с «1000» до «100», выключение «фиксированной» рандомизации (random_state = «None») и уменьшение числа сочетаний до пяти?

Таблица 3. Результаты экспериментов

Номер эксперимента	Результат		
	Метрика качества (МК)	Значение МК	Число сочетаний с определенной МК, шт.
1	MK_{GS}	0,938	1
2	MK_{RS}	0,938	49
		0,685	1
3	MK_{RS}	0,938	49
		0,763	1
4	MK_{RS}	0,938	50
5	MK_{RS}	0,938	48
		0,880	1
		0,826	
6	MK_{RS}	0,938	47
		0,702	1
		0,616	
		0,588	

Вывод

Из полученных результатов следует, что метод продемонстрировал достаточную высокую эффективность: доля обнаруженных сочетаний с максимальным значением метрики качества коле-

блется около единицы, даже при принудительном снижении требований к качеству получаемых моделей. Следовательно, представляется оправданным сделать вывод о возможности практического применения предложенного авторами метода при решении реальных задач, связанных с разработкой интеллектуальных систем.

Литература

1. Helena.4.0 – новый алгоритм для подбора гиперпараметров. URL: <https://habr.com/ru/companies/rosbank/articles/763026/> (дата обращения: 15.11.2023).
2. Hyperparameter Tuning with GridSearchCV. URL: <https://www.mygreatlearning.com/blog/gridsearchcv/> (дата обращения: 17.11.2023).
3. RandomizedSearchCV to Find OPTIMAL Parameters in Python. URL: <https://www.projectpro.io/recipes/find-optimal-parameters-using-randomizedsearchcv-for-regression> (дата обращения: 17.11.2023).
4. Градиентный спуск – полный, пакетный и стохастический. URL: <https://craftappmobile.com/градиентный-спуск-полный-пакетный-и-с/> (дата обращения: 18.11.2023).
5. Sklearn.linear_model.SGDClassifier. URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html#sklearn.linear_model.SGDClassifier (дата обращения: 18.11.2023).
6. Что такое Scikit Learn – гайд по популярной библиотеке Python для начинающих. URL: <https://datastart.ru/blog/read/chto-takoe-scikit-learn-gayd-po-populyarnoy-biblioteke-python-dlya-nachinayuschih> (дата обращения: 19.11.2023).
7. Пальмов С.В., Диязитдинова А.А., Губарева О.Ю. Исследование телекоммуникационного трафика средствами аналитической системы Orange // Моделирование, оптимизация и информационные технологии. 2019. Т. 7, № 1(24). С. 378–386. DOI: 10.26102/2310-6018/2019.24.1.031
8. Нормально делай – нормально будет: нормализация на практике – методы и средства Data Preparation. URL: <https://bigdataschool.ru/blog/нормализация-методы-средства-data-preparation.html> (дата обращения: 18.11.2023).
9. How to Manually Optimize Machine Learning Model Hyperparameters. URL: <https://machinelearningmastery.com/manually-optimize-hyperparameters/> (дата обращения: 18.11.2023).

10. What is the Difference between a Parameter and a Hyperparameter? URL: <https://machinelearningmastery.com/difference-between-a-parameter-and-a-hyperparameter/> (дата обращения: 18.11.2023).

Получено 27.11.2023

Пальмов Сергей Вадимович, к.т.н., доцент, доцент кафедры информационных систем и технологий (ИСТ) Поволжского государственного университета телекоммуникаций и информатики. 443090, Российская Федерация, г. Самара, Московское шоссе, 77. Тел. +7 927 706-61-21. E-mail: s.palmov@psuti.ru

Диязитдинова Альфия Асхатовна, ст. преподаватель кафедры ИСТ ПГУТИ. 443090, Российская Федерация, г. Самара, Московское шоссе, 77. Тел. +7 937 795-01-95. E-mail: a.diyazitdinova@psuti.ru

COMBINED METHOD FOR TUNING HYPERPARAMETERS OF A MATHEMATICAL MODEL

Palmov S.V., Diyazitdinova A.A.

*Povolzhskiy State University of Telecommunications and Informatics, Samara, Russian Federation
E-mail: s.palmov@psuti.ru, a.diyazitdinova@psuti.ru*

Automation of data processing processes is an important direction in the field of information technology. The main focus of researchers is usually on training intelligent systems. One of the key aspects of this process is the selection of hyperparameters for models. This research paper analyzes a combined method for tuning hyperparameters in a classification mathematical model. The method integrates the functionalities of two well-established approaches: exhaustive search and limited search. Initially, the first approach is employed to discover a preliminary estimation of the model's quality metric's maximum value. Subsequently, the second approach is utilized to generate a final estimation of achievable quality and compile a list of hyperparameter value combinations that optimize the classifier's efficiency. To verify the validity of the method, custom software was developed using the stochastic gradient descent algorithm. The results obtained from the experiment demonstrate the effectiveness of the proposed method.

Keywords: *hyperparameter, mathematical modeling, grid search, randomized search, gradient descent, machine learning*

DOI: 10.18469/ikt.2023.21.3.08

Palmov Sergey Vadimovich, Povolzhskiy State University of Telecommunications and Informatics, 77, Moscovskoye shosse, Samara, 443090, Russian Federation; Associated Professor of Information Systems and Technologies Department, PhD in Technical Science. Tel. +7 927 706-61-21. E-mail: s.palmov@psuti.ru

Diyazitdinova Alfiya Askhatovna, Povolzhskiy State University of Telecommunications and Informatics, 77, Moscovskoye shosse, Samara, 443090, Russian Federation; Senior Teacher of Information Systems and Technologies Department. Tel. +7 937 795-01-95. E-mail: a.diyazitdinova@psuti.ru

References

1. Helena.4.0 – A new algorithm for selecting hyperparameters. URL: <https://habr.com/ru/companies/rosbank/articles/763026/> (accessed: 15.11.2023). (In Russ.)
2. Hyperparameter Tuning with GridSearchCV. URL: <https://www.mygreatlearning.com/blog/grid-searchcv/> (accessed: 17.11.2023).
3. RandomizedSearchCV to Find Optimal Parameters in Python. URL: <https://www.project-pro.io/recipes/find-optimal-parameters-using-randomizedsearchcv-for-regression> (accessed: 17.11.2023).
4. Gradient descent – full, batch and stochastic. URL: <https://craftappmobile.com/градиентный-спуск-полный-пакетный-и-с/> (accessed: 18.11.2023). (In Russ.)

5. Sklearn.linear_model.SGDClassifier. URL: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html#sklearn.linear_model.SGDClassifier (accessed: 18.11.2023).
6. What is Scikit Learn – A guide to the popular python library for beginners. URL: <https://datastart.ru/blog/read/chto-takoe-scikit-learn-gayd-po-populyarnoy-biblioteke-python-dlya-nachinayuschih> (accessed: 19.11.2023). (In Russ.)
7. Palmov S.V., Diyazitdinova A.A., Gubareva O.Yu. Telecommunication traffic analysis using ORANGE analytical system. *Modelirovanie, optimizaciya i informacionnye tekhnologii*, 2019, vol. 7, no. 1(24), pp. 378–386. DOI: 10.26102/2310-6018/2019.24.1.031 (In Russ.)
8. Do things right – things will be fine: normalization in practice – methods and means of data preparation. URL: <https://bigdataschool.ru/blog/нормализация-методы-средства-data-preparation.html> (accessed: 18.11.2023). (In Russ.)
9. How to Manually Optimize Machine Learning Model Hyperparameters. URL: <https://machinelearningmastery.com/manually-optimize-hyperparameters/> (accessed: 18.11.2023).
10. What is the Difference between a Parameter and a Hyperparameter? URL: <https://machinelearningmastery.com/difference-between-a-parameter-and-a-hyperparameter/> (accessed: 18.11.2023).

Received 27.11.2023

УДК 004.94

ПРОГРАММНАЯ ОПТИМИЗАЦИЯ ПРОЦЕССА РАСПРЕДЕЛЕНИЯ РЕСУРСОВ В МЕТРОСТРОИТЕЛЬНОЙ ОРГАНИЗАЦИИ: АНАЛИЗ ЭФФЕКТИВНОСТИ СУЩЕСТВУЮЩИХ МЕТОДОВ

Собин А.Е.

Московский финансово-юридический университет, Москва, РФ
E-mail: view7goog757@gmail.com

В настоящей статье рассматриваются цели оптимального распределения ресурсов в строительных организациях метрополитена, включая повышение эффективности работ, сокращение продолжительности проекта и минимизацию затрат. Рассматриваются основные методы оптимизации, такие как математическое моделирование, анализ критического пути, использование сетевых графов и другие. Автором изучается роль программного обеспечения для управления проектами в обеспечении комплексного контроля над планированием, ресурсами и исполнением. Кроме того, в статье подчеркивается важность использования языка программирования Python и его библиотек, таких как Pandas, Scikit-learn и Matplotlib, для анализа и оптимизации распределения ресурсов в метростроении. Приведен конкретный пример программной реализации анализа рисков, связанных с задержками поставок в метростроительстве, с использованием языка программирования Python и его ключевых библиотек. Использовались Pandas для обработки и анализа данных, NumPy для выполнения сложных математических расчетов, а также Matplotlib для наглядной визуализации полученных результатов.

Ключевые слова: ресурсы, оптимизация, методы оптимизации, метростроительные организации, эффективность, затраты

Введение

В современных условиях сложной экономической ситуации, роста стоимости строительных материалов и введения ограничительных мер повышается значимость оптимизации процессов строительства. Оптимизация строительных проектов в организациях играет ключевую роль в достижении баланса между важнейшими аспектами, такими как стоимость, соблюдение графика и обеспечение высокого качества.

Оптимизация строительных проектов в организации направлена на эффективное использование

ресурсов и снижение издержек. Путем анализа и оптимизации всех этапов строительного процесса можно достичь оптимального сочетания стоимости работ, соблюдения установленных сроков и обеспечения высокого уровня качества [5, 14].

Обеспечение материально-техническими ресурсами в области метростроительства является ключевым фактором, напрямую влияющим на успешность внедрения таких сложных инфраструктурных задач. Задержки в поставке необходимых строительных материалов, оборудования и техники могут привести к непредвиденному