ИНФОРМАТИКА

УДК 004.891.3, 004.855.5, 004.9 DOI: https://doi.org/10.54341/20778708_2023_4_57_94
EDN: ZCJEJY

ИССЛЕДОВАНИЕ ПРОИЗВОДИТЕЛЬНОСТИ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ В ЗАДАЧАХ КЛАССИФИКАЦИИ ДАННЫХ

Е.В. Тимощенко[1], **А.Ф. Ражков**[2]

[1]*Могилевский государственный университет имени А.А. Кулешова*
[2]*Объединенный институт проблем информатики Национальной академии наук Беларуси*, *Минск*

RESEARCH OF THE PERFORMANCE OF MACHINE LEARNING ALGORITHMS IN DATA CLASSIFICATION PROBLEMS

E.V. Timoschenko[1], **A.F. Razhkov**[2]

[1]*Mogilev State A. Kuleshov University*
[2]*United Institute of Informatics Problems of the National Academy of Sciences of Belarus*, *Minsk*

Аннотация. Предложен подход к решению задачи построения моделей машинного обучения в решении задач классификации данных. На примере анализа наборов биомедицинских данных проведено сравнение производительности алгоритмов машинного обучения, настроенных с помощью предварительно оптимизированных гиперпараметров. Найдены наилучшие значения гиперпараметров, обеспечивающие эффективное прогнозирование, для самых распространенных алгоритмов машинного обучения.

Ключевые слова: *машинное обучение*, *классификация данных*, *оптимизация гиперпараметров*, *обработка больших данных*, *прогнозирование заболеваний.*

Для цитирования: *Тимощенко*, *Е.В.* Исследование производительности алгоритмов машинного обучения в задачах классификации данных / Е.В. Тимощенко, А.Ф. Ражков // Проблемы физики, математики и техники. – 2023. – № 4 (57). – С. 94–102. – DOI: https://doi.org/10.54341/20778708_2023_4_57_94. – EDN: ZCJEJY

Abstract. An approach to solving the problem of constructing machine learning models in solving data classification problems is proposed. Using the example of analyzing biomedical data sets, the performance of machine learning algorithms tuned using pre-optimized hyperparameters is compared. The best values of hyperparameters that provide effective prediction were found for the most common machine learning algorithms.

Keywords: *machine learning*, *data classification*, *hyperparameter optimization*, *big data processing*, *disease prediction.*

For citation: *Timoschenko*, *E.V.* Research of the performance of machine learning algorithms in data classification problems / E.V. Timoschenko, A.F. Razhkov // Problems of Physics, Mathematics and Technics. – 2023. – № 4 (57). – P. 94–102. – DOI: https://doi.org/10.54341/20778708_2023_4_57_94 (in Russian). – EDN: ZCJEJY

Введение

Машинное обучение находит широкое применение в медицине и здравоохранении. Оно помогает врачам в диагностике заболеваний, прогнозировании их развития и выборе оптимального лечения [1]. Алгоритмы машинного обучения анализируют большие массивы медицинских данных – результаты анализов, жалобы пациентов, данные мониторинга и т. д. На основе этих данных строятся прогностические модели, которые затем используются для конкретных пациентов. Использование моделей машинного обучения позволяет сравнивать характеристики нового пациента с имеющимися данными и получать вероятность наличия того или иного заболевания. Такой подход позволяет диагностировать заболевания на ранних стадиях и своевременно назначать лечение. Кроме того, врачи могут опираться на рекомендации системы при выборе схемы лечения и прогнозировании его эффективности для конкретного пациента [2]. Таким образом, машинное обучение делает медицинскую помощь более персонализированной и результативной.

Прогностические модели могут быть также полезны на этапе обучения будущих специалистов. Некоторые промежуточные результаты исследования, приведенного в данной статье (определение вероятности наличия заболевания у пациента по перечню биомедицинских данных, а также прогнозирование заболевания по симптомам пациента) нашли применение в сфере образования. Они были апробированы, положены в основу программного модуля виртуального практикума для студентов медико-биологического профиля [3] и успешно внедрены в учебный процесс МГУ имени А.А. Кулешова [4].

Построение эффективной модели машинного обучения является сложным и трудоемким процессом, который включает в себя нахождение

подходящего алгоритма обучения и получение оптимальной архитектуры модели путем настройки ее гиперпараметров [5] – параметров, которые настраиваются непосредственно перед обучением модели, а не в процессе машинного обучения.

1 Методика построения модели машинного обучения

Анализ доступных методов и схем построения модели машинного обучения [6]–[8], позволил выбрать надежную и оптимальную схему, наиболее подходящую для решения поставленной задачи (рисунок 1.1).

Реализация такой схемы проходит в несколько этапов.

1 *этап*. Сбор данных – это первый шаг при решении любой проблемы машинного обучения. Для данной задачи используются наборы биомедицинских данных из репозитория UCI [9], которые предназначены для обучения, проверки и тестирования. В выбранных нами наборах данных присутствуют признаки заболеваний, а также информация о наличии заболевания.

Каждый экземпляр данных, используемый в обучении модели машинного обучения, имеет как входные, так и выходную переменные. Например, данные, которые анализируются на наличие определенного заболевания, имеют в качестве входных переменных набор признаков, описывающих это заболевание, и в качестве выходной – атрибут, указывающий, имеется ли заболевание у пациента или нет.

2 *этап*. Подготовка данных – самый важный шаг проекта в сфере машинного обучения, который должен проводиться до того момента, как набор данных будет использоваться для обучения модели. Использование необработанных данных при моделировании может приводить к неверным результатам.

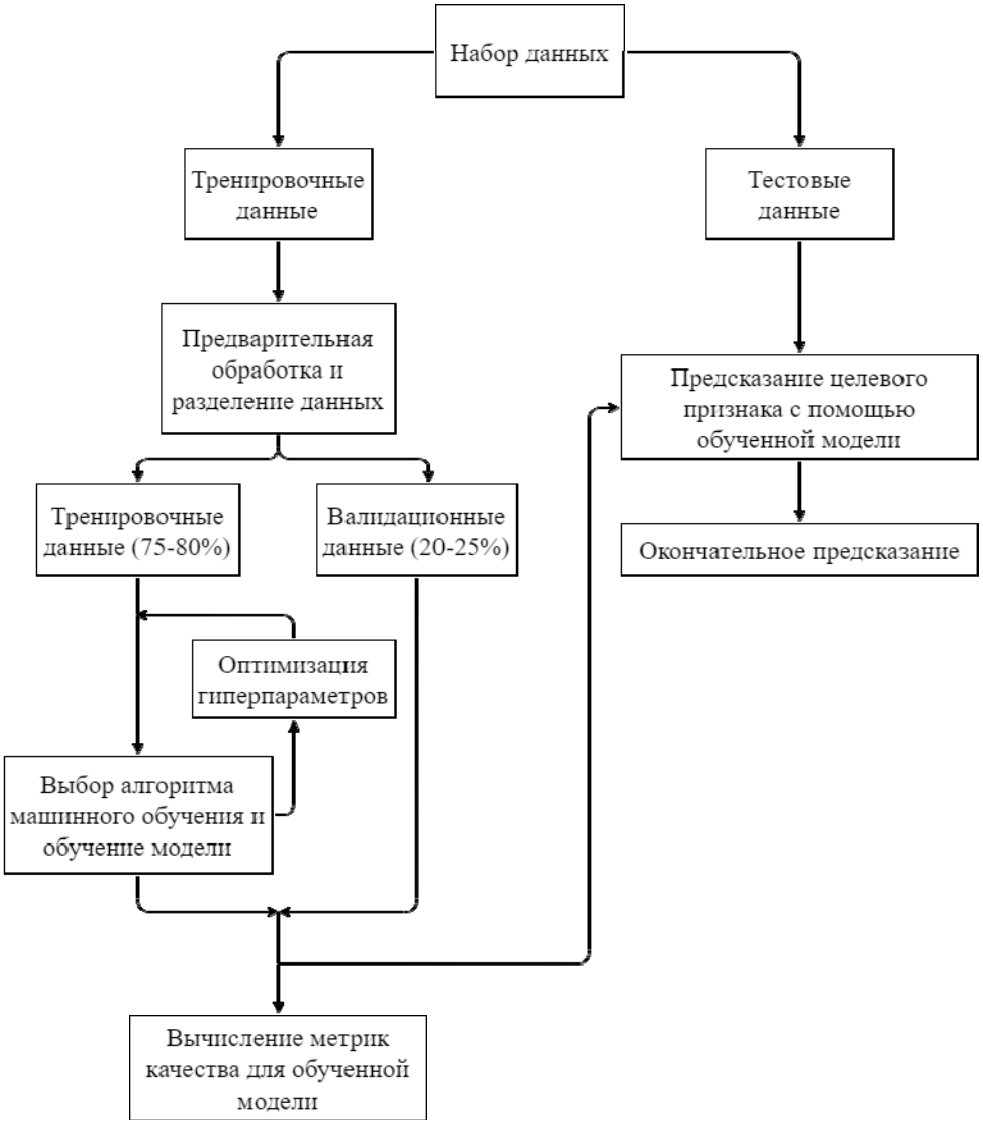

Рисунок 1.1. – Схема построения моделей машинного обучения

Таблица 1.1 – Используемые атрибуты набора данных

Атрибут	Описание	Тип, возможные значения
Age	Возраст	Число
Sex	Пол	0: женский 1: мужской
Cp	Тип болей в груди	0: бессимптомная 1: атипичная стенокардия 2: неангинальная 3: типичная стенокардия
Trestbps	Артериальное давление в состоянии покоя	Число
Chol	Общий холестерин в сыворотке крови	Число
Fbs	Уровень сахара в крови натощак	0: менее 120 мг / дл 1: более 120 мг / дл
Restecg	Результаты электрокардиографии в покое	0: норма 1: наличие аномалии ST-T 2: наличие вероятной или определенной гипертрофии левого желудочка по критериям Эстеса
Thalach	Максимальная достигнутая частота сердечных сокращений	Число
Exang	Стенокардия, вызванная физической нагрузкой	0: Нет 1: Да
Oldpeak	Депрессия ST-сегмента, вызванная упражнениями по сравнению с отдыхом	Число
Slope	Подъем пикового сегмента ST при физической нагрузке	0: косонисходящая 1: платообразная 2: косовосходящая
Ca	Количество крупных сосудов, окрашенных флуороскопией	0, 1, 2, 3
Thal	Талассемия	0: отсутствует 1: исправленный дефект 2: нормальный 3: обратимый дефект
Target	Наличие заболевания	0: отсутствие заболевания 1: наличие заболевания

3 *этап*. Построение модели машинного обучения: после сбора и подготовки данных они могут использоваться для обучения модели машинного обучения. Данный этап включает в себя выбор алгоритма машинного обучения и обучение модели, оптимизация гиперпараметров, вычисление метрик качества для обученной модели и ее проверка на тестовых данных.

В качестве прикладной задачи обучение моделей машинного обучения проводилось на наборах биомедицинских данных. Для примера в таблице 1.1 приведены атрибуты набора данных, которые использовались в обучения модели для прогнозной аналитики сердечно-сосудистых заболеваний.

Исследование производительности алгоритмов машинного обучения в задачах классификации данных проводилось в соответствии со схемой на рисунке 1 для решения прикладной задачи прогнозной аналитики биомедицинских данных. Для этого был проведен анализ наборов данных с использованием следующих алгоритмов машинного обучения [10], [11]–[14], [15]:

– Logistic Regression (Логистическая регрессия),

– Linear Discriminant Analysis (Линейный дискриминантный анализ),

– K-Neighbors Classifier (Метод ближайших соседей),

– Classification and Regression Tree (Метод построения деревьев решений),

– Naive Bayes Classifier (Наивный байесовский алгоритм),

– Linear Support Vector Classification (Линейный метод опорных векторов),

– C-Support Vector Classification (Метод опорных векторов),

– Multilayer Perceptron Classifier (Многослойный персептрон),

– Bagging Classifier (Бутстрэп-агрегирование),

– Random Forest Classifier (Случайный лес),

– Extra Trees Classifier (Классификатор экстремально рандомизированных деревьев),

– AdaBoost Classifier (адаптивный бустинг),

– Gradient Boosting Classifier (Градиентный бустинг),

– Light Gradient Boosting Machine (Градиентный бустинг деревьев решений LightGBM),

– Extreme Gradient Boosting (Градиентный бустинг деревьев решений XGBoost).

Модели машинного обучения используют 80% набора данных для обучения, а оставшиеся 20% используются для проверки точности алгоритмов. В качестве основной метрики для сравнения в данном случае используется доля правильных ответов алгоритма (*Accuracy*), определяемая следующим соотношением [16]:

$$Accuracy = (TP + TN) / (TP + FP + TN + FN),$$

где *TP* – классификатор верно утверждает, что объект принадлежит к рассматриваемому классу,

TN – классификатор верно утверждает, что объект **не** принадлежит к рассматриваемому классу,

FP – классификатор неверно утверждает, что объект принадлежит к рассматриваемому классу,

FN – классификатор неверно утверждает, что объект **не** принадлежит к рассматриваемому классу.

Для валидации точности прогнозирования были использованы наборы данных по сердечно-сосудистым заболеваниям, раку молочной железы, диабету, хронической болезни почек, заболеваниям печени [9].

2 Результаты исследования

Для наглядной демонстрации преимуществ предложенного подхода и высокой эффективности обученных моделей машинного обучения на примере анализа биомедицинских данных в таблицах 2.1–2.5 детально представлены результаты проведенного исследования производительности алгоритмов машинного обучения и их гиперпараметров для всех рассмотренных наборов данных.

Таблица 2.1 – Сердечно-сосудистые заболевания

Алгоритм	Точность (%)	Параметры	Значения параметров
(BG) Bagging Classifier	84.75	bootstrap, bootstrap_features, n_estimators, n_jobs, random_state, verbose, warm_start	487.0468, True, True, 50, 1, 0, 0, False
(ET) Extra Trees Classifier	84.75	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, n_estimators, n_jobs, random_state	entropy, 10, 2, 2, 0.0, 30, 6, 0
(LR) Logistic Regression	84.74	C, class_weight, fit_intercept, intercept_scaling, l1_ratio, max_iter, multi_class, n_jobs, penalty, random_state, solver, tol	0.01, balanced, False, 0.01, 0.2, 5000, auto, 1, elasticnet, 0, saga, 1.01
(CART) Classification and Regression Tree	84.35	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, splitter	11.4143, gini, 16, 6, 2, 0.0, random
(KNN) K-Neighbors Classifier	84.35	algorithm, leaf_size, n_jobs, n_neighbors, p, weights	auto, 2, 1, 9, 1, uniform
(LSVC) Linear Support Vector Classification	84.34	C, class_weight, dual, fit_intercept, intercept_scaling, loss, max_iter, multi_class, penalty, random_state, tol, verbose	0.01, , True, True, 0.41, squared_hinge, 5000, ovr, l2, 0, 2.01, 0
(RF) Random Forest Classifier	83.91	bootstrap, criterion, max_depth, max_features, n_estimators, random_state	True, gini, 2, auto, 110, 0
(XGB) Extreme Gradient Boosting	83.91	learning_rate, max_depth, n_estimators, nthread	0.051, 1, 480, 6
(LGBM) Light Gradient Boosting Machine	83.5	boosting_type, learning_rate, max_depth, n_jobs, num_leaves, objective, random_state	dart, 0.379, –1, 6, 2, binary, 0
(NB) Naive Bayes Classifier	83.5	var_smoothing	0.3960000001
(LDA) Linear Discriminant Analysis	83.1	solver, store_covariance, tol	svd, False, 0.8
(SVC) C-Support Vector Classification	83.1	C, degree, gamma, kernel, random_state, tol, verbose	0.61, 1, scale, poly, 0, 0.201, False
(MLP) Multilayer Perceptron Classifier	82.29	activation, alpha, hidden_layer_sizes, random_state, tol, verbose, warm_start	relu, 0.4, 130, 0, 0.001, False, False

Алгоритм	Точность (%)	Параметры	Значения параметров
(GB) Gradient Boosting Classifier	82.27	criterion, learning_rate, loss, max_depth, n_estimators, random_state, tol	friedman_mse, 0.1, exponential, 1, 400, 0, 0.0001
(AB) AdaBoost Classifier	82.26	algorithm, learning_rate, n_estimators, random_state	525.1748, SAMME, 0.36, 405, 0

Таблица 2.2 – Рак молочной железы

Алгоритм	Точность (%)	Параметры	Значения параметров
(LR) Logistic Regression	98.68	C, class_weight, fit_intercept, intercept_scaling, l1_ratio, max_iter, multi_class, n_jobs, penalty, random_state, solver, tol	0.21, , False, 0.01, 0.2, 5000, multinomial, 1, elasticnet, 0, saga, 0.01
(LSVC) Linear Support Vector Classification	98.68	C, class_weight, dual, fit_intercept, intercept_scaling, loss, max_iter, multi_class, penalty, random_state, tol, verbose	0.21, balanced, True, True, 0.61, hinge, 5000, ovr, l2, 0, 0.51, 0
(SVC) C-Support Vector Classification	98.24	C, degree, gamma, kernel, random_state, tol, verbose	0.41, 1, scale, linear, 0, 0.001, False
(MLP) Multilayer Perceptron Classifier	98.02	activation, alpha, hidden_layer_sizes, random_state, tol, verbose, warm_start	identity, 0.0, 100, 0, 0.001, False, False
(AB) AdaBoost Classifier	97.8	algorithm, learning_rate, n_estimators, random_state	SAMME, 0.4, 400, 0
(LGBM) Light Gradient Boosting Machine	97.8	boosting_type, learning_rate, max_depth, n_jobs, num_leaves, objective, random_state	gbdt, 0.364, –1, 6, 10, binary, 0
(ET) Extra Trees Classifier	97.58	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, n_estimators, n_jobs, random_state	entropy, 12, 1, 3, 0.0, 30, 6, 0
(GB) Gradient Boosting Classifier	97.36	criterion, learning_rate, loss, max_depth, n_estimators, random_state, tol	friedman_mse, 0.4, exponential, 1, 700, 0, 0.0001
(XGB) Extreme Gradient Boosting	97.14	learning_rate, max_depth, n_estimators, nthread	0.201, 1, 520, 6
(KNN) K-Neighbors Classifier	96.92	algorithm, leaf_size, n_jobs, n_neighbors, p, weights	auto, 2, 1, 11, 1, uniform
(RF) Random Forest Classifier	96.48	bootstrap, criterion, max_depth, max_features, n_estimators, random_state	True, entropy, 22, auto, 310, 0
(BG) Bagging Classifier	96.04	bootstrap, bootstrap_features, n_estimators, n_jobs, random_state, verbose, warm_start	True, True, 40, 1, 0, 0, False
(LDA) Linear Discriminant Analysis	96.04	solver, store_covariance, tol	svd, False, 0.2
(CART) Classification and Regression Tree	94.95	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, splitter	gini, 26, 6, 2, 0.0, random
(NB) Naive Bayes Classifier	94.73	var_smoothing	1.00E-10

Таблица 2.3 – Диабет

Алгоритм	Точность (%)	Параметры	Значения параметров
(CART) Classification and Regression Tree	81.17	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, splitter	entropy, 41, 11, 2, 0.0, random

Алгоритм	Точность (%)	Параметры	Значения параметров
(LSVC) Linear Support Vector Classification	81.16	C, class_weight, dual, fit_intercept, intercept_scaling, loss, max_iter, multi_class, penalty, random_state, tol, verbose	0.81, , True, True, 0.41, squared_hinge, 5000, ovr, l2, 0, 4.01, 0
(SVC) C-Support Vector Classification	79.55	C, degree, gamma, kernel, random_state, tol, verbose	0.21, 1, auto, poly, 0, 0.401, False
(RF) Random Forest Classifier	79.54	bootstrap, criterion, max_depth, max_features, n_estimators, random_state	True, gini, 22, log2, 310, 0
(LDA) Linear Discriminant Analysis	79.25	solver, store_covariance, tol	svd, False, 0.0
(LGBM) Light Gradient Boosting Machine	79.24	boosting_type, learning_rate, max_depth, n_jobs, num_leaves, objective, random_state	dart, 0.368, -1, 6, 2, binary, 0
(MLP) Multilayer Perceptron Classifier	79.24	activation, alpha, hidden_layer_sizes, random_state, tol, verbose, warm_start	identity, 0.6, 100, 0, 0.001, False, False
(ET) Extra Trees Classifier	79.23	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, n_estimators, n_jobs, random_state	gini, 16, 1, 3, 0.0, 40, 6, 0
(GB) Gradient Boosting Classifier	78.91	criterion, learning_rate, loss, max_depth, n_estimators, random_state, tol	friedman_mse, 0.4, deviance, 4, 100, 0, 0.0001
(LR) Logistic Regression	78.91	C, class_weight, fit_intercept, intercept_scaling, l1_ratio, max_iter, multi_class, n_jobs, penalty, random_state, solver, tol	0.21, , True, 0.01, 0.6, 5000, auto, 1, elasticnet, 0, saga, 1.01
(AB) AdaBoost Classifier	78.59	algorithm, learning_rate, n_estimators, random_state	SAMME, 0.38, 455, 0
(BG) Bagging Classifier	78.26	bootstrap, bootstrap_features, n_estimators, n_jobs, random_state, verbose, warm_start	True, False, 25, 1, 0, 0, False
(KNN) K-Neighbors Classifier	77.96	algorithm, leaf_size, n_jobs, n_neighbors, p, weights	auto, 2, 1, 29, 1, uniform
(NB) Naive Bayes Classifier	77.32	var_smoothing	0.2846000001
(XGB) Extreme Gradient Boosting	77.3	learning_rate, max_depth, n_estimators, nthread	0.051, 1, 440, 6

Таблица 2.4 – Хроническая болезнь почек

Алгоритм	Точность (%)	Параметры	Значения параметров
(AB) AdaBoost Classifier	100.0	algorithm, learning_rate, n_estimators, random_state	SAMME, 0.35, 400, 0
(CART) Classification and Regression Tree	100.0	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, splitter	gini, 1, 11, 2, 0.1, best
(ET) Extra Trees Classifier	100.0	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, n_estimators, n_jobs, random_state	gini, 10, 1, 2, 0.0, 20, 6, 0
(LGBM) Light Gradient Boosting Machine	100.0	boosting_type, learning_rate, max_depth, n_jobs, num_leaves, objective, random_state	gbdt, 0.35, –1, 6, 2, binary, 0

Алгоритм	Точность (%)	Параметры	Значения параметров
(LR) Logistic Regression	100.0	C, class_weight, fit_intercept, intercept_scaling, l1_ratio, max_iter, multi_class, n_jobs, penalty, random_state, solver, tol	0.01, , False, 0.01, 0.1, 5000, auto, 1, elasticnet, 0, saga, 0.01
(LSVC) Linear Support Vector Classification	100.0	C, class_weight, dual, fit_intercept, intercept_scaling, loss, max_iter, multi_class, penalty, random_state, tol, verbose	0.01, , False, True, 0.01, squared_hinge, 5000, ovr, l1, 0, 0.01, 0
(MLP) Multilayer Perceptron Classifier	100.0	activation, alpha, hidden_layer_sizes, random_state, tol, verbose, warm_start	identity, 0.0, 110, 0, 0.001, False, False
(NB) Naive Bayes Classifier	100.0	var_smoothing	1, 00E-10
(RF) Random Forest Classifier	100.0	bootstrap, criterion, max_depth, max_features, n_estimators, random_state	False, gini, 22, auto, 210, 0
(SVC) C-Support Vector Classification	100.0	C, degree, gamma, kernel, random_state, tol, verbose	0.21, 1, scale, linear, 0, 0.101, False
(XGB) Extreme Gradient Boosting	100.0	learning_rate, max_depth, n_estimators, nthread	0.001, 1, 400, 6
(KNN) K-Neighbors Classifier	97.6	algorithm, leaf_size, n_jobs, n_neighbors, p, weights	auto, 2, 1, 2, 1, distance
(BG) Bagging Classifier	98.4	bootstrap, bootstrap_features, n_estimators, n_jobs, random_state, verbose, warm_start	True, False, 35, 1, 0, 0, False
(GB) Gradient Boosting Classifier	98.4	criterion, learning_rate, loss, max_depth, n_estimators, random_state, tol	friedman_mse, 0.1, deviance, 1, 400, 0, 0.0001
(LDA) Linear Discriminant Analysis	99.2	solver, store_covariance, tol	svd, False, 0.4

Таблица 2.5 – Заболевания печени

Алгоритм	Точность (%)	Параметры	Значения параметров
(LSVC) Linear Support Vector Classification	73.23	C, class_weight, dual, fit_intercept, intercept_scaling, loss, max_iter, multi_class, penalty, random_state, tol, verbose	0.81, , True, True, 0.61, squared_hinge, 5000, ovr, l2, 0, 2.51, 0
(ET) Extra Trees Classifier	73.01	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, n_estimators, n_jobs, random_state	gini, 16, 1, 3, 0.0, 60, 6, 0
(LR) Logistic Regression	72.79	C, class_weight, fit_intercept, intercept_scaling, l1_ratio, max_iter, multi_class, n_jobs, penalty, random_state, solver, tol	0.81, , True, 0.01, 0.1, 5000, multinomial, 1, elasticnet, 0, saga, 0.01
(CART) Classification and Regression Tree	72.56	criterion, max_depth, min_samples_leaf, min_samples_split, min_weight_fraction_leaf, splitter	gini, 16, 16, 7, 0.0, random
(KNN) K-Neighbors Classifier	72.34	algorithm, leaf_size, n_jobs, n_neighbors, p, weights	auto, 2, 1, 27, 2, uniform
(MLP) Multilayer Perceptron Classifier	72.13	activation, alpha, hidden_layer_sizes, random_state, tol, verbose, warm_start	tanh, 0.4, 110, 0, 0.001, False, False

Алгоритм	Точность (%)	Параметры	Значения параметров
(LGBM) Light Gradient Boosting Machine	72.12	boosting_type, learning_rate, max_depth, n_jobs, num_leaves, objective, random_state	gbdt, 0.372, –1, 6, 9, binary, 0
(LDA) Linear Discriminant Analysis	71.68	solver, store_covariance, tol	svd, False, 0.1
(RF) Random Forest Classifier	71.46	bootstrap, criterion, max_depth, max_features, n_estimators, random_state	True, entropy, 2, auto, 10, 0
(SVC) C-Support Vector Classification	71.46	C, degree, gamma, kernel, random_state, tol, verbose	0.01, 1, scale, linear, 0, 0.001, False
(XGB) Extreme Gradient Boosting	71.46	learning_rate, max_depth, n_estimators, nthread	0.001, 0, 400, 6
(GB) Gradient Boosting Classifier	71.24	criterion, learning_rate, loss, max_depth, n_estimators, random_state, tol	friedman_mse, 0.1, exponential, 1, 100, 0, 0.0001
(AB) AdaBoost Classifier	70.12	algorithm, learning_rate, n_estimators, random_state	SAMME, 0.45, 470, 0
(BG) Bagging Classifier	68.58	bootstrap, bootstrap_features, n_estimators, n_jobs, random_state, verbose, warm_start	True, True, 95, 1, 0, 0, False
(NB) Naive Bayes Classifier	58.84	var_smoothing	1.00E-10

Анализ полученных результатов, проведенный для каждого набора данных, позволил определить наиболее эффективный алгоритм машинного обучения, обладающий наиболее высокой точностью.

Заключение

Предложен подход, основанный на использовании оптимизированных алгоритмов машинного обучения, который позволяет обеспечить эффективное прогнозирование заболеваний на этапе ранней диагностики. Проведен анализ производительности алгоритмов машинного обучения для классификации биомедицинских данных. Исследована эффективность наиболее популярных алгоритмов классификации данных. Для каждого алгоритма определены оптимальные значения гиперпараметров для наборов данных по сердечно-сосудистым заболеваниям, раку молочной железы, диабету, хронической болезни почек, заболеваниям печени.

Найдены алгоритмы машинного обучения, обладающие наивысшей производительностью: для сердечно-сосудистых заболеваний – Extra Trees Classifier – с точностью 84.80%, для рака молочной железы – Logistic Regression – с точностью 98.68%, для диабета – Classification and Regression Tree – с точностью 81.17%, для хронической болезни почек – AdaBoost Classifier – с точностью 100.0%, для заболеваний печени – Linear Support Vector Classification – с точностью 73.23%.

Полученные результаты могут быть использованы в здравоохранении для разработки систем поддержки принятия врачебных решений.

В перспективе с целью повышения точности и надежности результатов прогнозирования планируется проведение исследований в направлении разработки нейросетевых моделей на основе современных архитектур глубокого обучения, объединяющие в себе различные типы слоев.

ЛИТЕРАТУРА

1. *Rajkomar*, *A.* Machine Learning in Medicine / A. Rajkomar, J. Dean, I. Kohane // The New Englang journal of medicine. – 2019. – Apr 4. – № 380 (14). – P. 1347–1358. – DOI: 10.1056/NEJMra1814259.

2. *Almarabeh*, *H*. A study of data mining techniques accuracy for healthcare / H. Almarabeh, E. Amer // International Journal of Computer Applications. – Jun 2017. – Vol. 168, № 3. – P. 12–17.

3. *Тимощенко*, *Е.В*. Методы интеллектуального анализа данных в виртуальном практикуме для целей цифровизации образования / Е.В. Тимощенко, А.Ф. Ражков // Цифровая трансформация. – 2021. – № 4. – С. 52–62.

4. *Тимощенко*, *Е.В*. Интеллектуальный анализ данных: лабораторный практикум / Е.В. Тимощенко, А.Ф. Ражков. – Могилев: МГУ имени А.А. Кулешова, 2022. – 72 с.

5. *Ражков*, *А.Ф*. Оптимизация гиперпараметров алгоритмов машинного обучения для решения задач классификации данных / А.Ф. Ражков, Е.В. Тимощенко // Современное программирование:

материалы IV Международной научно-практической конференции (г. Нижневартовск, 8 декабря 2021 года) / отв. ред. Т.Б. Казиахмедов. – Нижневартовск: Нижневартовский государственный университет, 2022. – С. 267–274.

6. *Скобцов*, *В.Ю*. Нейросетевые модели для бинарной классификации данных телеметрической информации малых космических аппаратов / Скобцов В. Ю. // Информационные технологии и системы 2021 (ИТС 2021): материалы международной научной конференции, Минск, 24 ноября 2021 г. / Белорусский государственный университет информатики и радиоэлектроники; редкол.: Л.Ю. Шилин [и др.]. – Минск, 2021. – С. 98–100.

7. *A data-driven approach for multi-scale GIS-based building energy modeling for analysis*, *planning and support decision making* / Ali, Usman & Shamsi, Mohammad Haris & Bohacek, Mark & Purcell, Karl & Hoare, Cathal & Mangina, Eleni & O Donnell, James // Applied Energy. – 2020. – DOI: 279.10.1016/j.apenergy.2020.115834.

8. *Sen*, *P.C*. Supervised Classification Algorithms in Machine Learning: A Survey and Review / P.C. Sen, M. Hajra, M. Ghosh // In Emerging Technology in Modelling and Graphics; Advances in Intelligent Systems and Computing 937. – Springer Nature: Singapore, 2020. – P. 99–111.

9. *UCI Machine Learning Repository* [Electronic resource]. – Mode of access: https://archive.ics.uci.edu/. – Date of access: 20.09.2023.

10. *Тимощенко*, *Е.В*. Методы интеллектуального анализа биомедицинских данных / Е.В. Тимощенко, А.Ф. Ражков // Итоги научных исследований учёных МГУ им. А.А. Кулешова 2019 г.: материалы научно-методической конференции (г. Могилев, 29 января – 10 февраля. 2020 года). – Могилев, 2020. – С. 106–107.

11. *Balasree*, *K*. Big Data on Machine Learning – A Review / K. Balasree, K. Dharmarajan // Engineering and Scientific International Journal. – 2021. – № 8(3). – P. 86–91.

12. *Bardab*, *S.N*. Data mining classification algorithms: An overview / S.N. Bardab, T.M. Ahmed, T.A.A. Mohammed // International Journal of Advanced and Applied Sciences. – 2021. – № 8 (2). – P. 1–5.

13. *Fatima*, *M*. Survey of machine learning algorithms for disease diagnostic / M. Fatima, M. Pasha // Journal of Intelligent Learning Systems and Applications. – 2017. – Vol. 9, № 1. – P. 1–16.

14. Faouzi, Johann. Classic machine learning algorithms / Johann Faouzi, Olivier Colliot // Machine Learning for Brain Disorders. Neuromethods. – 2023. – Vol. 197. – DOI: 10.1007/978-1-0716-3195-9_2.

15. *Raschka*, *S*. Machine Learning in Python: Main Developments and Technology Trends in Data Science, Machine Learning, and Artificial Intelligence / S. Raschka, J. Patterson, C. Nolet // Information. – 2020. – Vol. 11, № 4. – P. 193. – DOI: 10.3390/info11040193.

16. *Erickson*, *B.J*. Magician's Corner: 9 / B.J. Erickson, F. Kitamura // Performance Metrics for Machine Learning Models. Radiology. Artificial Intelligence. – 2021. – № 3 (3). – DOI: 10.1148/ryai.2021200126.

Поступила в редакцию 29.09.2023.

Информация об авторах

Тимищенко Елена Валерьевна – к.ф.-м.н., доцент
Ражков Александр Федорович – аспирант