Математические структуры и моделирование
2024. № 1 (69). С. 56–68

УДК 004.896
DOI 10.24147/2222-8772.2024.1.56-68

ПРИМЕНЕНИЕ МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ДИАГНОСТИКИ ФЕНОТИПОВ ЗАБОЛЕВАНИЯ ЖЕЛУДОЧНО-КИШЕЧНОГО ТРАКТА

А.А. Березин[1]
аспирант, e-mail: andreyberezin55@gmail.com
С.А. Агалаков[2]
к.ф.-м.н., доцент, e-mail: agalakovsa@gsuite.omsu.ru

[1]Омский государственный технический университет, Омск, Россия

[2]Омский государственный университет им. Ф.М. Достоевского, Омск, Россия

Аннотация. Работа посвящена задаче диагностики фенотипов заболеваний желудочно-кишечного тракта с помощью моделей машинного обучения. Главной целью является поиск модели, решающей задачу классификации пациентов медицинского учреждения. В ходе работы сначала были рассмотрены простые модели, такие как логистическая регрессия и кластерный анализ. Ввиду неэффективности этих методов был выполнен поиск более сложных моделей и методов анализа данных: деревья решений, ансамбли деревьев, искусственные нейронные сети. Также была проведена работа по предварительной обработке набора данных и выбора значимых признаков для классификации. В результате была найдена наилучшая модель для рассматриваемого набора данных – CatBoost, которая диагностирует фенотипы с точностью 92,85 % на тренировочной выборке и на тестовой выборке – 79,31 %. Кроме того, в работе оценивается применимость методов машинного обучения в клинической практике и определяются направления будущих исследований для более точной диагностики.

Ключевые слова: машинное обучение, задача классификации, логистическая регрессия, нейронные сети, ансамбли решающих деревьев, бустинг.

Введение

В настоящее время машинное обучение и анализ данных становятся всё более актуальными в различных областях: в экономике, науке, здравоохранении, технике, маркетинге и т. д. Модели машинного обучения могут решать многие проблемы [1], но в эпоху пандемии и увеличения числа видов заболеваний как никогда актуальны задачи диагностики в здравоохранении. В общем и целом, медицина – это одна из самых важных сфер жизни людей, поэтому в данной работе рассматривается задача классификации фенотипов заболеваний желудочно-кишечного тракта (далее – ЖКТ) с помощью моделей машинного обучения.

Данные для работы были предоставлены Омским государственным медицинским университетом, они содержат информацию о реальных пациентах. Этот факт при-

даёт особую ценность и значимость проведённому исследованию, обогащая его реальным клиническим контекстом. В результате данного анализа предпринимается попытка улучшить процесс диагностики, что может иметь непосредственное влияние на практическую медицинскую практику и благополучие пациентов.

В качестве инструментов для практической работы с методами машинного обучения были использованы: язык Python, компоненты библиотек scikit-learn, Keras, NumPy.

1. Описание задачи

В данной работе поставлена определённая задача классификации фенотипов заболевания ЖКТ. Задача состоит в следующем. Есть результаты анкетирования пациентов и их медицинские показатели, представленные в таблице (112 признаков, 281 пациент), и на основе этих данных нужно отнести каждого пациента к одному из шести фенотипов заболевания ЖКТ: здоровые лица, ПИ-СРК, с ожирением, коморбидность, эссенциальный и смешанный фенотип.

Все исходные признаки разделены на пять групп вопросов/показателей: пищевые привычки (P), уровень стресса (S), симптомы болезни (B), показатели качества жизни (K), лабораторные маркеры (M).

Данная задача является примером задачи машинного обучения «с учителем», т. е. на вход модели будут поступать данные с правильными ответами, что необходимо для проверки качества обучения модели классификации. В качестве основной метрики будет использоваться *точность*.

Точность – это доля правильно классифицированных объектов среди всех объектов.

1.1. Описание данных

Группа «Пищевые привычки» насчитывает 70 признаков. Это очень большая группа вопросов, где людей просили ответить на вопросы относительно их предпочтений в еде и о приёме пищи. Например:

- Вы принимаете пищу примерно в одно и то же время?
- В течение последнего года пытались ли Вы изменить свои привычки питания в сторону здорового образа жизни – употреблять меньше жира?
- Сколько чашек кофе Вы обычно выпиваете за день (шт.)?
- Довольны ли Вы количеством пищи, потребляемым за сутки?
- Сколько граммов фруктов Вы употребляете в сутки?

Группа показателей «Уровень стресса» состоит из 6 признаков. Данные показатели оценивают уровень тревоги и депрессии пациента.

Группа «Симптомы болезни» насчитывает 20 показателей, среди которых: длительность заболевания, тошнота, неприятный вкус, урчание по ходу кишечника, метеоризм, индекс массы тела, частота стула, абдоминальная боль и др.

Группа «Показатели качества жизни» насчитывает 11 показателей: физическое функционирование, жизненная активность, социальное функционирование, психическое здоровье, ролевое функционирование, обусловленное эмоциональным состоянием, и др.

Группа «Лабораторные маркеры» состоит из 5 показателей: кортизол утром и вечером, серотонин, дофамин и зонулин.

1.2. Предварительная обработка данных

В данной работе на этапе разделения выборки на тестовую и тренировочную проводится нормализация данных с помощью StandardScaler из библиотеки scikit-learn для масштабирования данных. Это распространённый подход для предобработки данных перед применением моделей машинного обучения.

StandardScaler выполняет масштабирование данных путём преобразования их в стандартное нормальное распределение. А именно, данный метод применяет следующую формулу для каждого признака:

$$z = \frac{x - u}{s};$$

где z – новое масштабированное значение признака; x – исходное значение признака; u – среднее значение признака в наборе данных; s – стандартное отклонение признака в наборе данных.

Данный способ является стандартным и, как правило, необходимым для достижения большей точности при выполнении классификации.

Также были проведены эксперименты над различными пропорциями в разделении наблюдений на тренировочную и тестовую выборки: 70 % на 30 %, 80 % на 20 %, и лучшим для данной задачи оказалось разделение 90 % на 10 %.

На этапе обучения используется кросс-валидация по 5 фолдам: тренировочная выборка разделена на 5 частей и по очереди одна из них будет использована для тестирования в процессе обучения модели. Это улучшает точность на тестовой выборке.

2. Ход работы

В самом начале работы исследование началось с применения таких простых моделей, как логистическая регрессия и кластеризация методом k-средних на разных группах признаков.

2.1. Факторный анализ

Ввиду малого количества наблюдений (281) относительно большого количества переменных в данных (112), была выдвинута идея факторного анализа. Данная идея заключается в том, что с помощью факторного анализа можно сократить размерность данных, заменяя исходные переменные новыми факторами, которые объясняют основную часть изменчивости в данных, выполняя факторный анализ по группам вопросов, а не по всем показателям сразу.

В данной работе факторный анализ проводился с помощью пакета-программы SPSS по группам признаков отдельно, т. е. отдельно в группах «пищевые привычки», «уровень стресса» и остальных. При сокращении размерности использовался *метод главных компонент* [2]. Для определения количества факторов использовался основной критерий: доля объяснённой дисперсии должна быть около 80 %, при этом сначала нужно было выбрать те факторы, у которых собственные значения больше единицы.

В результате факторного анализа суммарное количество переменных сократилось почти в пять раз: со 112 признаков до 24. А именно для групп «Пищевые привычки», «Уровень стресса», «Симптомы болезни», «Показатели качества жизни», «Лабораторные маркеры» удалось провести факторный анализ и значительно сократить размерность: с 70, 6, 20, 11, 5 признаков до 15, 2, 2, 3, 2 соответственно.

После того как был проведён факторный анализ и получены новые переменные, была выполнена логистическая регрессия для этих значений (рис. 1). Также для сравнения результатов логистическая регрессия была выполнена и на всех исходных переменных (рис. 2).

```
Train_size =  0.9
Выборка -  X_FACTORS
Best params: {'C': 0.1, 'max_iter': 100, 'penalty': 'l2', 'solver': 'newton-cg'}
Train score: 0.5595238095238095
Test score: 0.4482758620689655
```

Рис. 1. Результаты логистической регрессии для всех новых факторов

```
Train_size =  0.9
Выборка -  X_ALL
Best params: {'C': 0.1, 'max_iter': 100, 'penalty': 'l1', 'solver': 'saga'}
Train score: 0.6428571428571429
Test score: 0.5172413793103449
```

Рис. 2. Результаты логистической регрессии для исходных признаков

Результаты логистической регрессии на новых факторах и на всём наборе данных оказались неудовлетворительными: 44,83 % точности для тестового набора на новых факторных переменных, 51,72 % – на всех переменных. Получается, что результаты на новых факторах ещё хуже, чем на исходных данных, поэтому можно сделать вывод, что факторный анализ не подходит для данной задачи классификации фенотипов заболевания ЖКТ.

Чтобы улучшить результаты логистической регрессии, нужно выполнить детальный анализ групп, чтобы сократить количество исходных признаков и проверить работу логистической регрессии на уменьшенном наборе данных. Возможно, есть признаки, которые негативно влияют на классификации, и, удалив их, можно добиться большей точности на представленных моделях.

2.2. Детальный анализ групп признаков

Проблема отбора значимых признаков для анализа данных заключается в том, что мы не знаем, на основе какого критерия нужно убирать из рассмотрения определённые признаки, потому что для этого нужно обладать дополнительными знаниями в области медицины. Поэтому мы можем судить только на основе следующего критерия: необходимо с помощью экспериментов над различными группами показателей пациентов определить, какую группу признаков можно полностью убрать из рассмотрения ввиду низких результатов классификации, а какую выборку – выбрать для дальнейших исследований.

Для начала следует разбить выборку на различные группы по признакам:

- Во-первых, проведём классификацию по одиночным группам признаков: P (пищевые привычки), S (уровень стресса), B (симптомы болезни), K (уровень качества жизни), M (лабораторные маркеры).

- Во-вторых, выполним логистическую регрессию по смешанным группам признаков.

В ходе анализа отдельных одиночных групп вопросов получили то, что группа показателей «Симптомы болезни» оказывает весомый вклад при классификации, потому что для этой группы признаков точность логистической регрессии на тестовой выборке составляет 68,97 %. Этот результат намного выше относительно остальных групп признаков, что неудивительно, поскольку объективно симптомы болезни часто хорошо описывают заболевание. Также можно сказать, что группа вопросов «Пищевые привычки» является наименее пригодной для исследования, потому что на ней самые низкие результаты логистической регрессии – 41,38 %, и эта группа вопросов составляет 70 признаков из 112, что больше половины.

На втором этапе анализа групп рассматриваются различные комбинации групп признаков, чтобы определить группу вопросов с наилучшими результатами логистической регрессии и максимальным количеством признаков, для того чтобы использовать этот набор данных для других моделей машинного обучения.

Получили, что точность логистической регрессии на тестовом наборе для всех 112 признаков равняется 51,72 %, что среди остальных смешанных групп вопросов является наименьшим результатом. Учитывая, что результаты среди остальных смешанных групп кардинально не отличаются (рис. 3), то в качестве основной выборки для обучения дальнейших моделей была выбрана смешанная группа SBKM из 42 признаков – именно эта группа содержит все признаки четырёх групп за исключением группы «Пищевые привычки».

Таким образом, на данном этапе мы не используем факторы, а будем выполнять дальнейшие исследования на исходных признаках без группы показателей «Пищевые привычки» по следующим причинам:

- Для группы SBKM точность равна 68,96 %, что является лучшим результатом среди всех групп для логистической регрессии. Таким образом, без группы признаков «Пищевые привычки» точность модели хуже не стала.

```
Train_size =  0.9
Выборка -  X_ALL
Best params: {'C': 0.1, 'max_iter': 100, 'penalty': 'l1', 'solver': 'saga'}
Train score: 0.6428571428571429
Test score: 0.5172413793103449

Train_size =  0.9
Выборка -  X_SBKM
Best params: {'C': 1, 'max_iter': 1000, 'penalty': 'l1', 'solver': 'saga'}
Train score: 0.75
Test score: 0.6896551724137931

Train_size =  0.9
Выборка -  X_BKM
Best params: {'C': 0.1, 'max_iter': 100, 'penalty': 'l2', 'solver': 'newton-cg'}
Train score: 0.6746031746031746
Test score: 0.6896551724137931

Train_size =  0.9
Выборка -  X_SBK
Best params: {'C': 0.1, 'max_iter': 100, 'penalty': 'l2', 'solver': 'newton-cg'}
Train score: 0.7023809523809523
Test score: 0.6551724137931034
```

Рис. 3. Результаты логистической регрессии на смешанных группах признаков

- Если более детально изучать признаки в группе «Пищевые привычки», то в ней есть много субъективных вопросов, достоверность ответов на которые неизвестна. Также непонятно, могут ли эти признаки влиять на фенотипы заболевания или нет. Например, среди признаков есть вопросы по типу: «довольны ли Вы разнообразием потребляемых продуктов питания?» или «сколько раз в день Вы пьёте кофе?»

- Определить, какие признаки значимы внутри группы, мы не можем, потому что этим должен заниматься специалист в области медицины.

Здесь же стоит сказать, что кластеризация методом k-средних показала низкие результаты. Точность составила 23,49 % на группе признаков SBKM – это один из худших показателей среди всех моделей (рис. 4). Можно сказать, что точность данной модели является вероятностью случайного угадывания.

Также и для других групп признаков точность кластеризации составляет от 10 до 24 %. Следовательно, для исходной задачи нужно искать более сложную модель.

3. Применение более сложных моделей машинного обучения

После того как были исследованы признаки, проведён факторный анализ и использованы простые модели для классификации, нужно приступать к поиску других, более сложных моделей. Следующей моделью стало решающее дерево.

```
execute_clustering(X_SBKM, y)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870:
  warnings.warn(
[0, 0, 0, 0, 0, 0, 3, 0, 5, 5, 5, 4, 4, 0, 5, 4, 5, 5, 3, 3, 0, 0, 0, 2
[0, 0, 0, 0, 0, 0, 4, 2, 1, 5, 1, 5, 3, 3, 4, 5, 3, 3, 1, 1, 0, 3, 2, 5
Adjusted Rand Index: 0.043265883908104 49
Accuracy: 23.49%
```

Рис. 4. Результаты кластеризации методом k-средних на наборе SBKM

3.1. Решающее дерево

Решающее дерево представляет такую структуру, что в каждой его вершине находится некоторый вопрос, а рёбра дерева соответствуют вариантам ответов на этот вопрос. В случае бинарного ответа: да или нет. Если в качестве признака выступает числовое значение, то определяется оптимальное пороговое значение, которое разделяет данные на две группы [3].

Решающее дерево даже при большой своей глубине не показывает результата лучше относительно логистической регрессии, несмотря на то, что использовалась кросс-валидация и подбор гиперпараметров по сетке.

Точность составила 62,07 % на тестовом наборе (рис. 5).

```
Best params: {'max_depth': 18, 'min_samples_leaf': 4, 'min_samples_split': 10}
Train score: 0.8253968253968254
Test score: 0.6206896551724138
```

Рис. 5. Точность и лучшие параметры модели решающего дерева

3.2. Случайный лес

Как правило, на практике в машинном обучении более интересной и плодотворной идеей является использование не одного дерева, а комбинации нескольких решающих деревьев. Особенность случайного леса заключается в том, что он генерирует несколько случайных деревьев, каждое из которых обучается на случайном наборе признаков и случайной подвыборке наблюдений из обучающего набора данных. При выдаче ответа алгоритм придерживается мнения большинства деревьев.

Случайный лес является одной из популярных реализаций алгоритма ансамблей деревьев в scikit-learn [4]. Обычно эта модель хорошо работает с небольшими и средними объёмами данных. Случайный лес может эффективно работать даже при наличии нескольких сотен признаков.

Данный алгоритм для рассматриваемой группы признаков даёт результат лучше относительно логистической регрессии, и точность на тестовом наборе составляет 72,41 % (рис. 6).

```
Best params: {'max_depth': 6, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 200}
Train score: 0.7777777777777778
Test score: 0.7241379310344828
```

Рис. 6. Точность и лучшие параметры для случайного леса

3.3. Градиентный бустинг

Случайный лес – это очень мощный алгоритм машинного обучения, но есть более продвинутый алгоритм построения ансамбля решающих деревьев – *бустинг*. Бустинг также строит множество деревьев, но в этом алгоритме построение деревьев происходит последовательно, так что каждое следующее дерево старается помаксимуму исправить ошибки всей предыдущей совокупности деревьев [3].

Выполнение GradientBoostingClassifier из пакета scikit-learn может выполняться на протяжении нескольких часов, что довольно долго при подборе параметров, и в итоге эта модель даёт небольшие результаты – 59 % точности.

Результат является осмысленным, потому что это мощный алгоритм, и он лучше подходит для задач с большим датасетом, который насчитывает десятки и сотни тысяч наблюдений.

3.4. XGBoost

Далее была найдена реализация алгоритма бустинга, которая отличается от предыдущей оптимизацией, позволяющей выполнять обучение и тестирование модели намного быстрее.

Результаты стали хуже относительно случайного леса – 62,06 % (рис. 7), и это также можно объяснить тем, что модель лучше подходит для большого массива данных. Градиентный бустинг, особенно его более продвинутые варианты (например, XGBoost, LightGBM), часто требуют большего объёма данных для достижения высокой точности.

```
Best params: {'learning_rate': 0.01, 'max_depth': 2, 'n_estimators': 200}
Train score: 0.6785714285714286
Test score: 0.6206896551724138
```

Рис. 7. Результаты и параметры XGBoost

3.5. MLPClassifier

Многослойный персептрон (MLP) представляет собой класс искусственных нейронных сетей с полносвязным типом связи нейронов [5]. Напомним, что нейронная сеть – это последовательность искусственных нейронов, связанных друг с другом. MLPClassifier из пакета scikit-learn является реализацией такой сложной структуры, как многослойный персептрон, но в то же время является простым в использовании и довольно быстрым алгоритмом по времени обучения.

Данную модель приходится учить и подбирать параметры для неё вручную, потому что в случае использования валидации модели по сетке параметров лучшая модель будет выбираться по точности на тренировочном наборе, а это не является показателем лучшей модели для данной задачи, так как нас интересует лучший результат на тестовом наборе, который нейросеть ещё не видела.

Проблема связана с тем, что, в отличие от предыдущих моделей, нейронные сети чаще склонны к переобучению, особенно на небольших датасетах, и высокий показатель на тренировочном наборе не гарантирует, что будет высокий показатель на тестовом наборе. Это связано с тем, что модель может просто запоминать тренировочный датасет и не выявлять характерные признаки данных.

Множество экспериментов показало, что лучше всего для данных подходит стохастический градиентный спуск «sgd», количество скрытых слоёв равно трём и количество нейронов (42, 50, 6) соответственно, alpha = 0,1 – коэффициент регуляризации и максимальное количество итераций – 1000 (рис. 8).

```
# Создание и обучение модели
model = MLPClassifier(solver='sgd', hidden_layer_sizes=(42, 50, 6), alpha = 0.1, max_iter=1000)
model.fit(X_train, y_train)
```

Рис. 8. Параметры модели MLPClassifier

В ходе подбора гиперпараметров было лучшим результатом 79,31 % точности на тестовом наборе (рис. 9).

```
          MLPClassifier
Accuracy train: 0.8174603174603174
Accuracy test: 0.7931034482758621
```

Рис. 9. Результаты модели MLPClassifier

Это неплохой результат, но больше достигнуть не удалось, потому что при добавлении дополнительных слоёв нейронов сеть быстро переобучается и выдаёт низкие результаты на тестовом наборе.

3.6. Нейронная сеть с использованием Keras

Минусом работы с MLPClassifier является то, что нет возможности гибко задавать параметры между слоями, что является проблемой в обучении. Например, тяжело найти момент во время обучения, когда эта модель переобучается.

В Python есть возможность реализовать нейросеть с гибкой настройкой параметров и слоёв нейронной сети. Для этого можно воспользоваться библиотекой Keras [6].

Однако при помощи данной библиотеки с тонкой настройкой параметров не удалось повысить точность: либо модель сильно переобучается, и тогда точность на тестовом наборе меньше 50 %, либо получается точность, сопоставимая с логистической регрессией, например, для сети с 4 слоями, количеством нейронов – (36,

128, 64, 6) и с использованием дропаута, который является методом регуляризации, точность равна 68,97 % (рис. 10).

```
Test loss TRAIN: 0.730056881904602
Test accuracy TRAIN: 0.7182539701461792
1/1 [==============================] - 0s
Test loss TEST: 0.8424545526504517
Test accuracy TEST: 0.6896551847457886
```

Рис. 10. Результаты нейросети, написанной с помощью Keras

3.7. CatBoost

После подбора параметров нейросети было решено вернуться к поиску алгоритмов с деревьями. В какой-то момент была найдена модель градиентного бустинга от «Яндекса» – CatBoost [7].

CatBoost позволяет использовать категориальные признаки без необходимости их предварительно обрабатывать. Данная модель хорошо подходит для небольшого набора данных, что важно в данной задаче. Также преимуществом является возможность обеспечивать повышенную точность за счёт уменьшения переобучения и проводить обучение на нескольких GPU.

Данная модель среди всех показывает лучший результат по точности: 92,85 % на тренировочных данных и 79,31 % правильно определённых фенотипов заболевания ЖКТ на тестовых (рис. 11). Лучший показатель был достигнут при следующих параметрах: глубина деревьев – 12, количество итераций – 200, темп обучения – 0,01.

```
Best params: {'depth': 12, 'iterations': 200, 'learning_rate': 0.01}
Train score: 0.9285714285714286
Test score: 0.7931034482758621
```

Рис. 11. Результаты и параметры модели CatBoost

Если посмотреть на дополнительные метрики *precision* и *recall*, то данный результат является довольно-таки высоким. *Precision* (точность) – процент представителей одного класса среди тех, которых алгоритм отнёс к этому классу. *Recall* (полнота) – процент найденных объектов одного класса среди всех объектов этого класса.

Например, если обратить внимание на полноту по классам на тестовой выборке (рис. 12), то для двух классов модель определяет все наблюдения, для двух других около 89 %, но есть один класс под номером «1» (ПИ-СРК), для которого полнота равна нулю. Детально проанализировав предсказанные значения, было выявлено, что в тестовую выборку попадают 2 представителя первого класса и оба определяются неправильно, отсюда полнота равна нулю. Это может указывать, что есть проблема с разделением наблюдений по классам.

```
Precision для каждого класса TRAIN:
Класс 0: 1.00
Класс 1: 1.00
Класс 2: 0.96
Класс 3: 0.91
Класс 4: 0.82
Класс 5: 1.00
Recall для каждого класса TRAIN:
Класс 0: 1.00
Класс 1: 0.69
Класс 2: 1.00
Класс 3: 0.98
Класс 4: 0.93
Класс 5: 0.93
```

```
Precision для каждого класса TEST:
Класс 0: 1.00
Класс 1: 0.00
Класс 2: 0.50
Класс 3: 0.73
Класс 4: 0.75
Класс 5: 1.00
Recall для каждого класса TEST:
Класс 0: 1.00
Класс 1: 0.00
Класс 2: 1.00
Класс 3: 0.89
Класс 4: 0.75
Класс 5: 0.88
```

Рис. 12. *Precision* и *recall* для модели CatBoost

3.8. Результаты работы моделей

Таким образом, получили следующие результаты на группе из 42 признаков (табл. 1).

Таблица 1. Результаты моделей на группе признаков SBKM, %

Модель	Train	Test
Логистическая регрессия	75,00	68,97
Кластеризация методом k-средних	—	23,49
MLPClassifier	81,74	79,31
Нейросеть Keras	73,00	68,97
Решающее дерево	82,53	62,06
Случайный лес	77,78	72,41
XGBoost	67,86	62,07
CatBoost	92,85	79,31

Для того чтобы убедиться, что лучшие результаты достигаются на группе исходных признаков, состоящей из 4 групп вопросов (без пищевых привычек), покажем результаты на новых факторах (табл. 2) и на всех исходных 112 признаках с группой P («Пищевые привычки») (табл. 3).

Заключение

Данная работа посвящена поиску модели машинного обучения, которая решает задачу классификации фенотипов заболевания ЖКТ. Были рассмотрены модели и методы машинного обучения: нейронные сети, решающие деревья, логистическая регрессия, факторный анализ, кластерный анализ.

Таблица 2. Результаты моделей на новых факторах, %

Модель	Train	Test
Логистическая регрессия	55,95	44,82
Кластеризация методом k-средних	—	18,86
MLPClassifier	73,80	58,62
Решающее дерево	55,55	58,62
Случайный лес	89,28	51,72
XGBoost	100	48,27

Таблица 3. Результаты моделей на всех исходных признаков, %

Модель	Train	Test
Логистическая регрессия	64,28	51,72
Кластеризация методом k-средних	—	11,39
MLPClassifier	99,20	44,82
Решающее дерево	56,34	44,82
Случайный лес	80,15	68,96
XGBoost	86,11	55,17

Выявлено, что классифицировать фенотипы заболеваний по данной выборке можно, но с разной точностью. Была найдена модель бустинга CatBoost, которая является продвинутой реализацией градиентного бустинга, разработанного в «Яндексе». CatBoost для исходного набора данных на 42 признаках лучше всего подходит для данной задачи и выдаёт следующие лучшие результаты: 92,85 % точности на тренировочной выборке и на тестовой выборке – 79,31 %. Данный результат можно считать неплохим, учитывая проблемы с малым количеством данных, а также рассмотрев дополнительные метрики *precision* и *recall*.

Можно предположить, что в исходном наборе есть аномальные данные, потому что было применено множество моделей, а также для каждой из них был выполнен поиск наилучших параметров, но в итоге точность так и не превысила 90 % на тестовом наборе для всех рассмотренных моделей. В таком случае остаётся искать выбросы в датасете, чтобы повысить точность, но данная задача выходит за рамки данной статьи.

Для дальнейшего продолжения исследования в данной области потребуется собрать больше данных, а также необходим специалист в области медицины, чтобы уточнить достоверность признаков, а также из групп вопросов отобрать признаки, которые с медицинской точки зрения должны вносить большой вклад в диагностику фенотипов заболевания ЖКТ.

Код программ был написан в Google Colab на языке Python. Для создания моде-

лей использовались компоненты библиотеки scikit-learn и Keras. Медицинские данные для исследования были предоставлены Омским государственным медицинским университетом. Результаты работы изображены на графиках, в таблицах и записаны в блокноты Google Colab.

Литература

1. Введение в машинное обучение. URL: `https://habr.com/ru/post/448892/` (дата обращения: 25.02.2024).
2. Агалаков С.А. Статистические методы анализа данных: учеб. пособие. Омск: Изд-во Ом. гос. ун-та, 2017.
3. Быстрый старт в искусственный интеллект. URL: `https://stepik.org/course/80782/` (дата обращения: 31.05.2023).
4. Официальная документация Scikit-learn. URL: `https://scikit-learn.org/stable/index.html` (дата обращения: 25.02.2024).
5. Николенко С., Кадурин А., Архангельская Е. Глубокое обучение. СПб.: Питер, 2018.
6. Официальная документация Keras. URL: `https://keras.io/api/` (дата обращения: 25.02.2024).
7. Официальная документация CatBoost. URL: `https://catboost.ai/en/docs/` (дата обращения: 25.02.2024).

APPLICATION OF MACHINE LEARNING MODELS FOR THE DIAGNOSIS OF GASTROINTESTINAL DISEASE PHENOTYPES

A.A. Berezin[1]
Ph.D. Student, e-mail: andreyberezin55@gmail.com
S.A. Agalakov[2]
Ph.D. (Phys.-Math.), Associate Professor, e-mail: agalakovsa@gsuite.omsu.ru

[1]Omsk State Technical University, Omsk, Russia
[2]Dostoevsky Omsk State University, Omsk, Russia

Abstract. The work is devoted to the task of diagnosing the phenotypes of diseases of the gastrointestinal tract using machine learning models. The main purpose of this work is to find a model that solves the problem of classifying patients in a medical institution. In the course of the work, simple models such as logistic regression and cluster analysis were first considered. Due to the inefficiency of these methods, a search was made for more complex models and methods of data analysis: decision trees, ensembles of trees, artificial neural networks. Work was also carried out on the preliminary processing of the data set and the selection of significant features for classification. As a result of the work, the best model for the data set under consideration was found – CatBoost, which diagnoses phenotypes with an accuracy of 92.85 % in the training sample and 79.31 % in the test sample. In addition, the paper evaluates the applicability of machine learning methods in clinical practice and identifies areas for future research for more accurate diagnosis.

Keywords: machine learning, classification problem, logistic regression, neural networks, ensembles of decision trees, boosting.

Дата поступления в редакцию: 29.02.2024