УДК 004.89
DOI: 10.33184/bulletin-bsu-2023.1.4

ROC-КРИВАЯ И МАТРИЦА ПУТАНИЦЫ КАК ЭФФЕКТИВНОЕ СРЕДСТВО ДЛЯ ОПТИМИЗАЦИИ КЛАССИФИКАТОРОВ МАШИННОГО ОБУЧЕНИЯ

© В. М. Горяев*, В. Д. Бурлыков, С. Н. Прошкин, В. В. Лиджи-Гаряев, Е. Н. Джахнаева

Калмыцкий государственный университет им. Б. Б. Городовикова
Россия, Республика Калмыкия, 358000 г. Элиста, ул. Пушкина, 11.

Тел.: +7 (84722) 4 10 05.
**Email: goryaeff@mail.ru*

Модель классификации машинного обучения может использоваться для прямого прогнозирования фактического класса точки данных или прогнозирования вероятности ее принадлежности к разным классам. Вероятность дает нам больше контроля над результатом. Можно определить свой собственный порог для интерпретации результата классификатора, что, как правило, лучше, чем просто создание совершенно новой модели. Установка различных пороговых значений для классификации положительных классов для точек данных непреднамеренно изменяет чувствительность и специфичность модели, а один из этих порогов, вероятно, и даст лучший результат. Для определения оптимального порога генерируются графики с некоторыми параметрами модели. Важным инструментом для процесса оптимизации классификации являются метрики оценки. Параметры кривой характеристики оператора приемника (ROC) являются метрикой оценки для такого рода задач. Для понимания такого графика генерируют матрицу путаницы (ошибок) для каждой точки, соответствующей порогу, что позволяет рассуждать о производительности классификатора. Для данного исследования было использовано подмножество набора данных Lending Club. Выполнена оценка прогноза, где точность модели составила 96%, отзыв составил 92%, а показатель f1 был равен 94%. Алгоритм KNN показал результат лучше, чем регрессия, с показателем AUC 0.93 и коэффициентом Gini=1.

Ключевые слова: *машинное обучение, классификаторы, метод ближайших соседей, логистическая регрессия, кривая ROC, матрица путаницы.*

Введение

В цифровом мире все меняется с большой скоростью, и бизнес внимательно следит за этим темпом, изменяя способ конкуренции на рынке, быть в курсе самых передовых технологий для удовлетворения сегодняшних бизнес-потребностей. Технологические достижения в области анализа данных (DA) создают новые возможности. Современный DA позволяет выявить скрытые закономерности данных благодаря автоматизации, прогнозированию и адаптивности, он влияет на каждый аспект нашей жизни, независимо от того, какой вопрос мы задаем, аналитика дает ответы и помогает принимать эффективные решения, DA дает возможность узнать правильный прогноз бизнес-целей, благодаря сочетанию личных, деловых и больших данных с помощью которых можно будет получить реалистичный ответ. Машинное обучение (ML) – это новая отрасль науки, которая достаточно быстро входим в нашу повседневную жизнь. Машинное обучение сейчас повсюду – от целевой рекламы до распознавания преступников. Можно констатировать, что ИИ настолько хорош, насколько хороши модели машин, которые им управляют. Для того чтобы определить, так ли хороша модель машинного обучения, как хотелось бы, очевидно, что нужны объективные средства измерения производительности данной модели ML и определения, достаточно ли она хороша для внедрения [1–2].

Существуют несколько способов оценки алгоритмов классификации. Анализ таких показателей и его значимость должны быть правильно интерпретированы для оценки различных алгоритмов обучения. Большинство из этих показателей являются скалярными метриками, а некоторые из них являются графическими методами [3–4]. Кривая ROC (рабочая характеристика приемника) – это график, который показывает производительность модели классификации при всех пороговых значениях классификации [5]. Это кривая вероятности, которая отображает два параметра, истинную положительную скорость (TPR) и ложноположительную скорость (FPR), при разных пороговых значениях и отделяет так называемый «сигнал» от «шума». ROC отображает истинную положительную скорость и ложноположительную скорость при разных порогах классификации. Если пользователь понижает порог классификации, больше элементов классифицируются как положительные, что увеличивает как ложные срабатывания, так и истинные положительные. Важное значение имеет еще одна характеристика, по сути, являющаяся кратким описанием ROC-кривой-площадь под ROC-кривой (AUC), которая измеряет всю двумерную область, расположенную под всей ROC-кривой от (0.0) до (1.1). AUC измеряет способность классификатора различать классы, и чем выше AUC, тем лучше модель может различать положи-

тельные и отрицательные классы [6]. AUC предоставляет совокупную меру производительности модели по всем возможным пороговым значениям классификации. Для более точной интерпретации ROC-кривой выполняют построение матрицы путаницы (МП). Матрица путаницы – это табличное представление для описания производительности модели классификации на наборе тестовых данных, для которых известны фактические значения. Это позволяет легко идентифицировать путаницу между классами, т.е. когда один класс ошибочно помечен как другой.

Материалы и методы

Для данного исследования было использовано подмножество набора данных Lending Club [7]. В наборе данных поле loan amnt – это указанная сумма кредита, предоставленного заемщиком, int – процентная ставка по кредиту, присвоенному кредитному рейтингу, на основе кредитной истории заемщика, продолжительности emp – продолжительности работы заемщика в годах, n – Текущее состояние кредита (например, полностью выплачено или списано), это метка, которую мы собираемся спрогнозировать с помощью модели. Всего набор Lending Club (LC) представлен 848 453 записями. Для анализа данных был использован Python v.3.7 [8].

loan.tail()# Просмотр последние 5 строк данных с помощью функции .tail() (*табл.* 1).

Основные оценочные показатели

Матрица путаницы двоичных классификаторов имеет четыре результата: истинные положительные результаты (TP), истинные отрицательные результаты (TN), ложные положительные результаты (FP) и ложные отрицательные результаты (FN). В этом исследовании мы обсуждаем следующие показатели: точность (ACC), чувствительность (SN), специфичность (SP), частота истинных положительных результатов (TPR), отзыв (REC), частота ложных положительных результатов (FPR), точность (PREC) [9–10].

AUC – инструмент для оценки производительности модели.

Шкала модели имеет коэффициент AUC в пределах от 0 до1, соответственно при нуле показатель разделимости (ПР) наихудший и при стремлении к 1 ПР улучшается, где AUC=1 означает, что это отличная модель, близость к 0 означает, что она возвращает результат, предсказывая отрицательный класс как положительный и, наоборот, показывая 0 как 1, а 1 как 0. Наконец, если AUC равен 0.5, это показывает, что модель вообще не имеет возможности разделения классов. Таким образом, когда результат в диапазоне 0.5<AUC <1, существует высокая вероятность того, что классификатор может различать положительные значения класса и отрицательные значения класса. Это потому, что классификатор может обнаруживать большее количество истинных положительных и отрицательных результатов вместо ложных отрицательных и положительных результатов. Далее необходимо определить связь между чувствительностью, специфичностью, FPR и порогом (*табл.* 2). Сокращение TP означает истинно положительный, а TN – истинно отрицательный. Соответственно FP – ложноположительный результат, а FN – ложноотрицательный результат [11–12].

Чувствительность, или отзыв (recall) – это показатель, который показывает способность модели предсказывать истинные положительные результаты всех доступных категорий. Она показывает, какая доля положительного класса была классифицирована правильно. Например, при попытке выяснить, сколько людей носят медицинские маски, чувствительность или истинный положительный показатель, измеряют долю людей в масках, и были правильно предсказаны, как имеющие их. Математически расчет чувствительности:

$$Recall = \frac{TP}{TP+FN} \quad (1)$$

Таблица 1

Кадр последних 5 записей набора данных LC

	loan_amnt	term	sub_grade	loan_status	emp	revol_util	num_op_rev_tl	…	n
848449	22400.0	2.0	22.0	0.0	10.0	51.5	12.0	…	1
848450	19400.0	2.0	24.0	1.0	1.0	63.8	9.0	…	0
848451	11200.0	2.0	22.0	0.0	5.0	54.7	7.0	…	1
848452	23800.0	2.0	24.0	1.0	10.0	89.5	8.0	…	0
848453	24000.0	2.0	24.0	1.0	6.0	68.1	8.0	…	0

Таблица 2

Матрица путаницы

	Истинный прогноз	Ложный прогноз
Положительный класс	Истинно-положительный (**TP**)	Ложноотрицательный (**FN**)
Отрицательный класс	Ложноположительный (**FP**)	Истинно-отрицательный (**TN**)

Специфичность, или истинно-отрицательный коэффициент (**TPR**) – метрика оценивает способность модели предсказывать истинные отрицательные результаты всех доступных категорий, показывает, какая доля отрицательного класса была классифицирована правильно.

$$TPR = \frac{TP}{TP+FN} \quad (2)$$

Последствия, или частота ложноположительных результатов (**FPR**) – обозначает частоту ложных срабатываний и показывает, какая доля отрицательного класса была неправильно классифицирована. Эта формула показывает, как мы вычисляем FPR:

$$FPR = \frac{FP}{FP+TN} = 1 - TNR \quad (3)$$

И, наконец, очень важная характеристика – это **пороговое значение (PT)** – указанная точка отсечения, при которой наблюдается бинарная классификация. Обычно по умолчанию в качестве порогового значения используется PT=0,5.

$$PT = \frac{\sqrt{FPR}}{\sqrt{TPR}+\sqrt{FPR}} \quad (4)$$

Чувствительность и специфичность обратно пропорциональны, поэтому, если мы повышаем чувствительность, специфичность падает, и наоборот. Кроме того, при классификации обычно получается больше положительных значений при уменьшении порога, тем самым повышая чувствительность и снижая специфичность. С другой стороны, при увеличении порогового значения получается больше отрицательных значений, что приводит к более высокой специфичности и снижению чувствительности. И поскольку FPR=1, когда TPR увеличивается, FPR также увеличивается, и наоборот [13–14].

Результаты исследования и обсуждение

Рассмотрим стандартную ситуацию в банковском учреждении, когда служащий на основе исходных данных намерен предсказать вероятность того, вернет клиент кредит или попытается погасит его [15–16]. Такой тип задачи называется задачей классификации, где нужно найти вероятность того, что событие произойдет или нет, или будет ли оно истинным/ложным. Для решения этой задачи используем популярные контролируемые ML-алгоритмы: логистическую регрессию (LR-Logistic Regression) и алгоритм K-ближайшего соседа (KNN) [17–19]. LR оценивает вероятность, используя лежащую в ее основе логистическую функцию, которая называется сигмоидной функцией. Сигмоидная функция – это математическая функция, которая выдает значение вероятности между 0 и 1 при любом значении x [20].

Кривая рабочих характеристик приемника (ROC) – это оценочный показатель для двоичного классификатора, который помогает нам визуализировать производительность модели по мере изменения ее порога. В двоичной классификации мы принимаем двоичное решение, используя непрерывное значение вероятности для данной выборки, принадлежащей одному из двух классов. Мы хотим найти оптимальный порог для этого значения вероятности.

Сначала необходимо разделить набор данных на обучающий и тестовый наборы. Наиболее типичное соотношение – 80% для обучения и 20% для тестирования. Передаем обучающий набор в модель и подгоняем ее на основе этого набора. Затем тестируем или оцениваем модель на основе тестового набора. Вначале можно сравнить прогнозы модели с фактическими данными тестового набора и определить, достаточно ли хороши прогнозы модели.

Чтобы разделить данные на обучающий и тестовый наборы, воспользуемся библиотекой sklearn [21–22], где разделим данные на входные (наши признаки), которые являются X, и выходные, которые прогнозируем (метка истинности), которые являются y. Затем разделим их на тестовый и обучающий наборы с помощью функции train_test_split.

Сначала требуется обучить модель классификатора в наборе данных:

```
import train_test_split from sklearn.model_selection
from sklearn. naive_bayes import GaussianNB
# Разделение на train и тестовые наборы
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size = 0.20)
# Создание объектной модели
logreg = LogisticRegression(max_iter = 1000)
logreg2 = KNNClassifier(max_iter = 1000)
# Подгонка модели к модели обучающих данных
logreg.fit(X_train, y_train)
# Прогнозирование классов по тестовым данным
y_pred = model.predict(X_test)
# Прогнозирование классов по тестовым данным
и возврат вероятности
y_proba = model.predict_proba(X_test)
```

Определяем функцию для вычисления TPR и FPR для каждого экземпляра на основе уравнений, представленных ранее. Далее создается классификатор логической регрессии, и модель обучается на учебном множестве с помощью функции fit.

Accuracy точность классификатора ЛР на обучающем множестве: 0.7944.

Accuracy точность классификатора ЛР на тестовом множестве: 0.7933.

Recall чувствительность классификатора ЛР на тестовом множестве: 0.0965.

Precision точность классификатора ЛР на тестовом множестве: 0.5110.

ROC/AUC классификатора ЛР на тестовом множестве: 0.7083.

Оценка точности рассчитывается на основе предположения, что класс выбран, если вероятность его предсказания превышает 50%. Это означает, что будет рассмотрен только 1 случай (одна рабочая точка) из многих [23]. Допустим, мы хотим классифицировать экземпляр как «0», даже если он имеет вероятность более 30%, это может произойти в том случае, если один из классов более важен, чем другой, а его априорная вероятность очень мала. В этом случае будет совсем другая матрица путаницы с другой точностью ([TP + TN]/[ALL]).

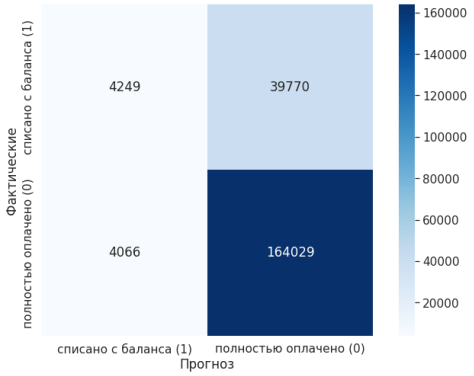

Рис. 1. Матрица путаницы.

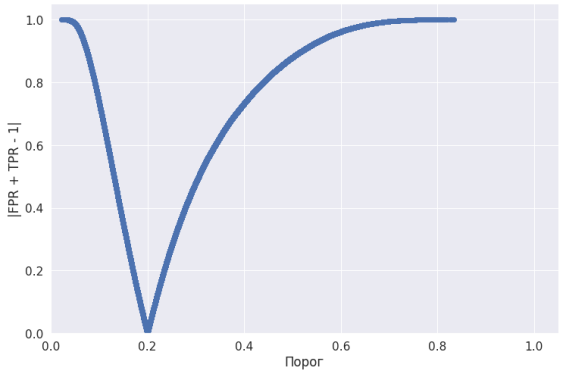

Рис. 2. Функции порога.

Оценка ROC-AUC проверяет все эти рабочие точки и дает оценку модели в целом. Оценка 50% означает, что модель соответствует случайному выбору классов на основе априорных вероятностей классов. Чем ОКР выше, тем лучше модель. Таким образом, в приведенном выше случае можно констатировать, что вышепоказанная модель не имеет хорошей предсказательной силы.

Выполним построение матрицы путаницы (*рис.* 1):

```
logreg_conf_matrix = confusion_matrix(y_test, y_pred)
```

Матрица путаницы показывает, что истинные отрицательные значения (TN) = 164 413 и указывают на то, что заявители прогнозировали выплату кредита в полном объеме, и они фактически полностью выплатили кредит. Истинно положительные результаты (TP) составляют 3 022 и указывают на то, что заявители прогнозировали списание, и они фактически списались. Число ложноотрицательных (FN) = 40 997 означает предсказание, что заявители должны были выплатить кредит полностью, но они фактически списали деньги. Число ложноположительных результатов (FP) = 3 682 и указывает на то, что заявителям было предсказано списать средства, но на самом деле они полностью выплатили свои кредиты.

Важно, что каждый отдельный заявитель, который может не вернуть кредит, оказывает в десять раз большее финансовое воздействие на банк, чем тот доход, который могли бы получить от окупившегося заявителя [24]. Поэтому стоит фокусироваться на правильном определении кандидатов, которые, скорее всего, не вернут кредит. На основе значений матрицы путаницы можно рассчитать показатели, которые предоставляют дополнительную информацию о текущей модели: расчет оптимального порога принятия решения и построение ROC-кривой (*рис.* 3).

```
logit_roc_auc = roc_auc_score(y_test,
logreg.predict_proba(X_test)[:,1])

fpr, tpr, thresholds = roc_curve(y_test,
logreg.predict_proba(X_test)[:,1])

plt.figure(figsize = (12,8))

plt.plot(fpr, tpr,
label='Лог. регрессия (площадь = %0.2f)' %
logit_roc_auc)
```

AUC составляет 0.71 на основе гиперпараметров логистической регрессии по умолчанию, и он выглядит далеко не идеальным. В будущем стоит поэкспериментировать с настройкой гиперпараметров, чтобы построить модель, которая может лучше разделять классы. Как только будет достигнуто желаемое значение AUC, можно установить порог принятия решения для оптимизации выбранной метрики модели ML. Разные пороговые значения принятия решений дадут разные TPR, FPR и другие показатели.

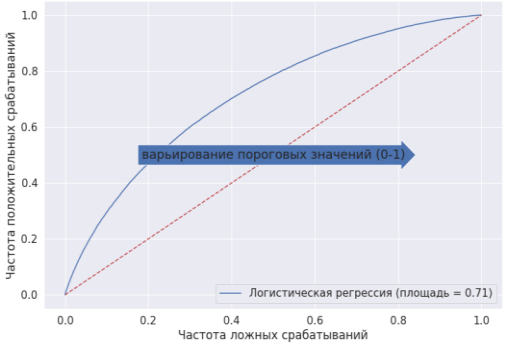

Рис. 3. Пороговые значения модели ML.

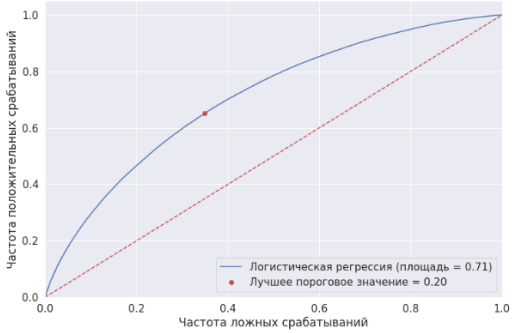

Рис. 4. Оптимальное значение порога.

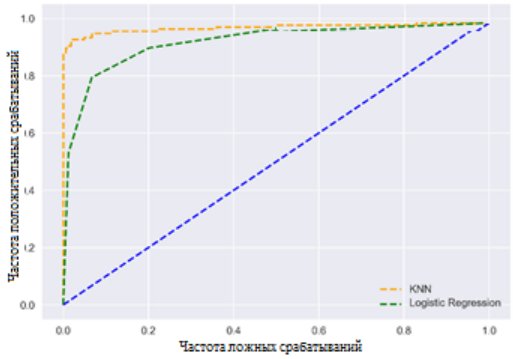

Рис. 5. ROC-кривые для LR и KNN.

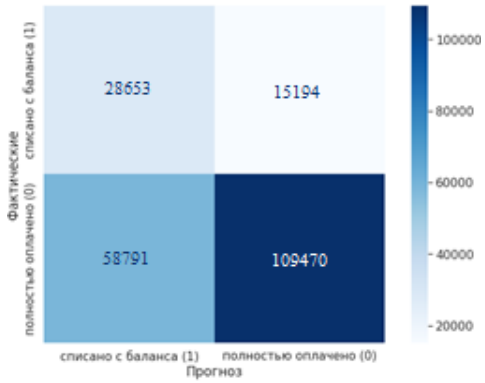

Рис. 6. Матрица путаницы.

Предположим, есть два претендента на получение кредита. Для первого заявителя наша модель ML предсказывает, что класс 0 имеет вероятность 0.9 (вероятность выплаты кредита), а класс 1 имеет вероятность 0.1 (вероятность списания средств). В этом случае можно быть уверенным, что этот кандидат, скорее всего, вернет кредит, учитывая значение вероятности. Однако второй кандидат, получивший вероятность 0.71 для класса 0 и вероятность 0.29 для класса 1 и учитывая бизнес-контекст и показатели, будет уже не столь дисциплинирован. Визуализация найденного порогового значения на *рис.* 4. Построим ROC-кривую для двух классификаторов тестового пула набора данных.

Следующий фрагмент кода визуализирует ROC-кривую для двух обученных моделей и показывает их оценку AUC в легенде (*рис.* 5):

```
plt.plot(fpr1, tpr1, linestyle='--',
color='orange', label='Logistic Regression')
plt.plot(fpr2, tpr2, linestyle='--',
color='green', label='KNN')
```

Чтобы сделать значения матрицы путаницы (*рис.* 6) более понятными, надо получить отчет о классификации для тестовых данных и получить некоторые общие показатели, такие как точность, отзыв и оценка f1.

```
from sklearn.metrics import classification_report
print(classification_report(y_test, y__hat_test))
```

	precision	recall	f1-score	support
0	0.90	0.84	0.82	28653
1	0.88	0.94	0.93	58791
accuracy			0.92	212114
macro avg	0.92	0.93	0.91	212114
weighted avg	0.92	0.92	0.93	212114

Общая точность для LR равна 92%, а precision, recall и F1-мера для предсказания «списать» выше 80%, а для прогноза «оплатить» – 91%.

	precision	recall	f1-score	support
0	0.91	0.83	0.81	87444
1	0.89	0.92	0.91	124670
accuracy			0.91	212114
macro avg	0.91	0.93	0.92	212114
weighted avg	0.93	0.93	0.93	212114

Общая точность для KNN равна 92%, а precision, recall и F1-мера для предсказания «списать» выше 80%, а для прогноза «оплатить» – 92%. Здесь можно видеть, что точность модели составляет 92% против 91%, отзыв составляет 84/83%, а показатель f1 равен 91/92%. Среди этих двух моделей нет идеальной, но они обе находятся далеко от базовой линии (непригодной модели). Алгоритм KNN здесь лучший, с показателем AUC 0.92 и коэффициентом Gini=1. Точность и отзыв более интерпретируемы, чем оценка f1, поскольку они измеряют ошибку типа 1 и ошибку типа 2. Тем не менее оценка f1 измеряет компромисс между этими двумя показателями. На графике можно увидеть, что AUC для кривой ROC KNN выше, чем для логистической регрессии. Таким образом, можно констатировать, что KNN несколько лучше справилась с классификацией положительного класса в текущем наборе данных.

Заключение

Для набора данных, основанных на Lending Club Loan Data, который содержит полные данные по кредитам, выданным за указанный период времени, включая текущий статус кредита, была поставлена задача оценить классификаторы на эффективность с помощью визуализированных метрик.

ROC-графики – очень полезный инструмент для визуализации и оценки классификаторов. Они способны обеспечить более высокий показатель эффективности классификации, чем скалярные показатели, такие как точность, частота ошибок или стоимость ошибок. Поскольку они отделяют производительность классификатора от перекоса классов и затрат на ошибки, то имеют преимущества по сравнению с другими оценочными показателями, такими как графики точного воспроизведения и кривые подъема. Однако, как и в случае с любой метрикой оценки их разумное использование требует знания их характеристик и методики их применения. Кривые ROC позволяют сравнивать набор оценок с непрерывным значением с набором меток с двоичным

значением путем применения к этим оценкам изменяющегося порога различения. Если оценки уже являются двоичными, тогда нет необходимости применять какой-либо порог – просто вычислить истинно положительный и ложноположительный коэффициент, напрямую сравнивая оценки с метками. Общая точность в эксперименте для KNN была равна 92%, что выше LR=91%, а f1 – оценка соответственно 92 против 91%. При этом видно, что модели достаточно хорошо обучены. Точность и отзыв более интерпретируемы, чем оценка f1, поскольку они измеряют ошибку типа 1 и ошибку типа 2. Тем не менее оценка f1 измеряет компромисс между этими двумя показателями. Общий вывод: KNN чуть лучше справилась с классификацией положительного класса в текущем наборе данных. В будущем предстоит поэкспериментировать с настройками гиперпараметров и другими типами характеристик ML, чтобы построить модель, которая сможет лучше разделять классы.

ЛИТЕРАТУРА

1. Практическая статистика для специалистов Data Science / пер. с англ. П. Брюс, Э. Брюс, П. Гедек. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2021. 352 с.
2. Рындина С. В. Бизнес-аналитика на основе больших данных: обучение без учителя на языках Python и R: учеб.-метод. пос. Пенза: изд-во ПГУ, 2020. 76 с.
3. Tharwat A. Classification assessment methods//Applied Computing and Informatics. 2021. Vol. 17. No. 1. P. 168–192.
4. Мерфи К. П. Вероятностное машинное обучение: введение / пер. с англ. А. А. Слинкина. М.: ДМК Пресс, 2022. 940 с.
5. Powers D. Estimation: from accuracy, recall and F-measurement to ROC, informativeness, labelling and correlation//Journal of Machine Learning Technology. 2011. Vol. 2(1). P. 37–63.
6. Calders T. et.all. Efficient AUC optimization for classification. Knowledge discovery in databases: PKDD 2007. Lecture notes in computer science. Berlin, Heidelberg: Springer, 2007. Vol. 4702. P. 42–53.
7. Devlin J. Lending Club Loan Data 2007-11 // Data.world. 2016. URL: https://data.world/jaypeedevlin/lending-club-loan-data-2007-11 (дата обращения: 12.11.2022).
8. Вандер П. Python для сложных задач: наука о данных и машинное обучение. СПб.: Питер, 2018. 576 с.
9. Maratea A., Petrosino A., Manzo A. Adjusted f-measure and kernel scaling for imbalanced data learning//Inf. Sci. 2014. Vol. 257. P. 331–341.
10. Hernandez-Orallo J., Flach P., Ferri, C. A unified view of performance metrics: transforming threshold selection into expected classification loss//Journal of Machine Learning Research. 2012. Vol. 13. P. 2813–2869.
11. Груздев А. В. Метод бинарной логистической регрессии в банковском скоринге//Риск-менеджмент в кредитной организации. 2012. №2. С. 76–91.
12. Аббасов М. Е. Методы оптимизации. СПб: ВВМ, 2014. 64 с.
13. Aurelien Geron. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2017. P. 718.
14. Полетаева Н. Г. Классификация систем машинного обучения // Вестник Балтийского фед. ун-та им. И. Канта. Серия: Физико-матем. и техн. науки. 2020. №1. С. 5–22.
15. Аггарвал Ч. Нейронные сети и глубокое обучение: учебный курс. СПб.: Диалектика, 2020. С. 472–474.
16. Goryaev V. M., Basangova E. O. et al. Forecasting steppe fires using remote sensing data of time series. IOP Conference Series//Materials Science and Engineering. Krasnoyarsk Science and Technology City Hall. Krasnoyarsk. 2021. Vol. 1047. No. 1. P. 12092.
17. Гудфеллоу Дж. Глубокое обучение. М.: ДМК. Пресс, 2018. 652 с.
18. Горяев В. М., Бембитов Д. Б., Мучкаев Д. М., Аль-Килани В. Х. Модель SARIMA и статистика скользящего окна для локальных метеоданных // Современные наукоемкие технологии. 2019. №6. С. 31–38.
19. Мастицкий С. Э., Шитиков В. К. Статистический анализ и визуализация данных с помощью R. М.: ДМК Пресс, 2015. 496 с.
20. Грас Д. Data Science. Наука о данных с нуля / пер. с англ. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2021. 416 с.
21. Шурыгин А. М. Математические методы прогнозирования: учеб. пос. для вузов. М.: Горячая линия – Телеком, 2009. 180 с.
22. Python. Scikit-learn // 3.3. Метрики и оценки: количественная оценка качества прогнозов. 2016. URL: https://scikit-learn.ru/3-3-metrics-and-scoring-quantifying-the-quality-of-predictions/ (дата обращения: 12.11.2022).
23. Старовойтов В., Голуб Ю. Об оценке результатов классификации несбалансированных данных по матрице ошибок // Обработка сигналов, изображений и данных и распознавание образов signal. 2021. №18. С. 61–71.
24. Илюхина М. В. Риски коммерческих банков при кредитовании физических лиц и способы их минимизации // Державинские чтения: мат-лы XXV Всерос. науч. конф. 2020. С. 124–134.

Поступила в редакцию 15.11.2022 г.

DOI: 10.33184/bulletin-bsu-2023.1.4

ROC-CURVE AND CONFUSION MATRIX AS AN EFFECTIVE TOOL FOR OPTIMIZING MACHINE LEARNING CLASSIFIERS

© V. M. Goryaev*, V. D. Burlykov, S. N. Proshkin, V. V. Lidzhi-Garyaev, E. N. Dzhakhnaeva

Kalmyk State University
11 Pushkin Street, 358000 Elista, Republic of Kalmykia, Russia.

Phone: +7 (84722) 4 10 05.
**Email: goryaeff@mail.ru*

A machine learning classification model can be used to directly predict the actual class of a data point or to predict the probability of it belonging to different classes. Probability gives us more control over the result. One can define one's own threshold for interpreting the result of the classifier, which is usually better than simply creating a completely new model. Setting different thresholds for classifying positive classes for data points unintentionally changes the sensitivity and specificity of the model, and one of these thresholds is likely to give the best result. Graphs with some model parameters are generated to determine the optimal threshold. Evaluation metrics are an important tool for the classification optimization process. The receiver operator characteristic curve (ROC) parameters are the evaluation metric for this kind of task. To understand such graph, a confusion matrix is generated for each point corresponding to a threshold, which allows reasoning about the performance of the classifier. A subset of the Lending Club dataset was used for this study. Prediction evaluation was performed: the accuracy of the model was 96%, the recall was 92%, and the f1 score was 94%. The KNN algorithm performed better than the regression, with an AUC of 0.93 and Gini = 1.

Keywords: machine learning, classifiers, nearest neighbor method, logistic regression, ROC curve, confusion matrix.

Published in Russian. Do not hesitate to contact us at bulletin_bsu@mail.ru if you need translation of the article.

1. Prakticheskaya statistika dlya spetsialistov Data Science [Practical statistics for data scientists]. 2nd ed. Saint Petersburg: BXV-Peterburg, 2021.
2. Ryndina S. V. Biznes-analitika na osnove bol'shikh dannykh: obuchenie bez uchitelya na yazykakh Python i R: ucheb.-metod. pos. [Big data business intelligence: unsupervised learning in Python and R: study guide]. Penza: izd-vo PGU, 2020.
3. Tharwat A. Applied Computing and Informatics. 2021. Vol. 17. No. 1. Pp. 168–192.
4. Merfi K. P. Veroyatnostnoe mashinnoe obuchenie: vvedenie [Probabilistic machine learning: introduction]. Moscow: DMK Press, 2022.
5. Powers D. Journal of Machine Learning Technology. 2011. Vol. 2(1). Pp. 37–63.
6. Calders T. Lecture notes in computer science. Berlin, Heidelberg: Springer, 2007. Vol. 4702. Pp. 42–53.
7. Devlin J. Data.world. 2016. URL: https://data.world/jaypeedevlin/lending-club-loan-data-2007-11 (data obrashcheniya: 12.11.2022).
8. Vander P. Python dlya slozhnykh zadach: nauka o dannykh i mashinnoe obuchenie [Python for complex tasks: data science and machine learning]. Saint Petersburg: Piter, 2018.
9. Maratea A., Petrosino A., Manzo A.Inf. Sci. 2014. Vol. 257. Pp. 331–341.
10. Hernandez-Orallo J., Flach P., Ferri, C.Journal of Machine Learning Research. 2012. Vol. 13. Pp. 2813–2869.
11. Gruzdev A. V.Risk-menedzhment v kreditnoi organizatsii. 2012. No. 2. Pp. 76–91.
12. Abbasov M. E. Metody optimizatsii [Optimization methods]. Saint Petersburg: VVM, 2014.
13. Aurelien Geron. Hands-on machine learning with Scikit-learn and TensorFlow: Concepts, tools, and techniques to build intelligent systems. 2017. Pp. 718.
14. Poletaeva N. G. Vestnik Baltiiskogo fed. un-ta im. I. Kanta. Seriya: Fiziko-matem. i tekhn. nauki. 2020. No. 1. Pp. 5–22.22).
15. Aggarval Ch. Neironnye seti i glubokoe obuchenie: uchebnyi kurs [Neural networks and deep learning: tutorial]. Saint Petersburg: Dialektika, 2020. Pp. 472–474.
16. Goryaev V. M., Basangova E. O. Materials Science and Engineering. Krasnoyarsk. 2021. Vol. 1047. No. 1. Pp. 12092.
17. Goodfellow J. Glubokoe obuchenie [Deep learning]. Moscow: DMK. Press, 2018.
18. Goryaev V. M., Bembitov D. B., Muchkaev D. M., Al'-Kilani V. Kh. Sovremennye naukoemkie tekhnologii. 2019. No. 6. Pp. 31–38.
19. Mastitskii S. E., Shitikov V. K. Statisticheskii analiz i vizualizatsiya dannykh s pomoshch'yu R [Statistical analysis and data visualization with R]. Moscow: DMK Press, 2015.
20. Grus J. Data Science. Nauka o dannykh s nulya [Data science from scratch]. 2-e izd. Saint Petersburg: BXV-Peterburr, 2021.
21. Shurygin A. M. Matematicheskie metody prognozirovaniya: ucheb. pos. dlya vuzov [Mathematical methods of forecasting: textbook for universities]. Moscow: Goryachaya liniya – Telekom, 2009.
22. Python. Scikit-learn. 3.3. Metriki i otsenki: kolichestvennaya otsenka kachestva prognozov. 2016. URL: https://scikit-learn.ru/3-3-metrics-and-scoring-quantifying-the-quality-of-predictions/ (data obrashcheniya: 12.11.2022).
23. Starovoitov V., Golub Yu. Obrabotka signalov, izobrazhenii i dannykh i raspoznavanie obrazov signal. 2021. No. 18. Pp. 61–71.
24. Ilyukhina M. V. Derzhavinskie chteniya: mat-ly XXV Vseros. nauch. konf. 2020. Pp. 124–134.

Received 15.11.2022.