УДК 004.89
https://doi.org/10.37661/1816-0301-2023-20-1-55-74

Оригинальная статья
Original Paper

Классификация займов с использованием логистической регрессии

В. И. Бегунков✉, М. Я. Ковалев[1]

[1]*Объединенный институт проблем информатики Национальной академии наук Беларуси, ул. Сурганова, 6, Минск, 220012, Беларусь*
✉*E-mail: vbegunkov@gmail.com*

Аннотация

Цели. Решение задачи классификации займов имеет большое значение для финансовых институтов, которые должны эффективно распределять денежные активы между субъектами в рамках предоставления финансовых услуг. Поэтому финансовым организациям необходим инструмент наиболее точного определения надежных заемщиков. Одним из инструментов принятия таких решений служит машинное обучение. Целью работы является анализ возможности эффективного применения логистической регрессии для решения задачи классификации займов.

Метод. На основе алгоритма логистической регрессии с использованием исторических данных по выданным займам рассчитываются следующие метрики: стоимостная функция, *Accuracy*, *Precision*, *Recall* и мера F_1. Полиномиальная регрессия и метод главных компонент применяются для определения оптимального набора входных данных для исследуемого алгоритма логистической регрессии.

Результаты. Оценено влияние нормализации данных на конечный результат, дана оценка влияния сбалансированности целевых значений, рассчитано оптимальное граничное значение для алгоритма логистической регрессии, рассмотрено влияние увеличения входных показателей посредством заполнения отсутствующих значений и использования полиномов разной степени. Имеющийся набор входных показателей проанализирован на избыточность.

Заключение. Результаты исследований подтверждают, что применение алгоритма логистической регрессии для решения задач классификации займов является целесообразным. Данный алгоритм позволяет быстро получить работающий инструмент классификации займов.

Ключевые слова: классификация займов, скоринг, логистическая регрессия, машинное обучение, нормализация данных

Для цитирования. Бегунков, В. И. Классификация займов с использованием логистической регрессии / В. И. Бегунков, М. Я. Ковалев // Информатика. – 2023. − Т. 20, № 1. – С. 55–74.
https://doi.org/10.37661/1816-0301-2023-20-1-55-74

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 11.01.2023
Подписана в печать | Accepted 10.02.2023
Опубликована | Published 29.03.2023

Loan classification using logistic regression

Uladzimir I. Behunkou✉, Mikhail Y. Kovalyov[1]

[1]The United Institute of Informatics Problems of the National Academy of Sciences of Belarus, st. Surganova, 6, Minsk, 220012, Belarus
✉E-mail: vbegunkov@gmail.com

Abstract

Objectives. The studied problem of loan classification is particularly important for financial institutions, which must efficiently allocate monetary assets between entities as part of the provision of financial services. Therefore, it is more important than ever for financial institutions to be able to identify reliable borrowers as accurately as possible. At the same time, machine learning is one of the tools for making such decisions. The aim of this work is to analyze the possibility of efficient use of logistic regression for solving the task of loan classification.

Methods. Based on the logistic regression algorithm using historical data on loans issued, the following metrics are calculated: cost function, *Accuracy*, *Precision*, *Recall* и F_1 score. Polynomial regression and principal component analysis are used to determine the optimal set of input data for the being studied logistic regression algorithm.

Results. The impact of data normalization on the final result is estimated, the optimal regularization parameter for solving this problem is determined, the impact of the balance of target values is assessed, the optimal boundary value for the logistic regression algorithm is calculated, the influence of increasing input indicators by means of filling in missing values and using polynomials of different degrees is considered and the existing set of input indicators is analyzed for redundancy.

Conclusion. The research results confirm that the application of the logistic regression algorithm for solving loan classification problems is appropriate. The use of this algorithm allows to get quickly a working loan classification tool.

Keywords: loan classification, scoring, logistic regression, machine learning, data normalization

For citation. Behunkou U. I., Kovalyov M. Y. *Loan classification using logistic regression*. Informatika [*Informatics*], 2023, vol. 20, no. 1, pp. 55−74 (In Russ.). https://doi.org/10.37661/1816-0301-2023-20-1-55-74

Conflict of interest. The authors declare of no conflict of interest.

Введение. Кредитование давно стало важным инструментом ускорения роста экономик как развитых, так и развивающихся стран. В связи с тем что в любой экономике имеются субъекты хозяйствования с избытком и недостатком денежных средств, кредитование позволяет распределять денежные активы с максимальной выгодой для всех участников, а значит, максимально использовать потенциал экономики отдельно взятой страны. Ключевая роль в данном процессе принадлежит финансовым институтам, которые должны эффективно распределять денежные активы между субъектами в рамках предоставления своих услуг. Поэтому финансовым организациям как никогда важно иметь возможность наиболее точно определять надежных заемщиков, необходимы современные инструменты, посредством которых можно удаленно и максимально быстро принимать решения по поступающим запросам на предоставление займов. Машинное обучение является одним из методов, на основе которого такие инструменты принятия решений могут быть разработаны [1]. В свою очередь, решаемую с помощью данных инструментов задачу можно представить как бинарную задачу классификации займа. Хэнд и Хэнли в своей работе [2] представили задачу классификации займа как бинарную следующим образом: разделили претендентов на кредиты на два класса в соответствии с вероятностью погашения займа на хороших (без дефолта) и плохих (с дефолтом). Развитие и использование различных алгоритмов для кредитного скоринга привели к необходимости проведения сравнительного анализа таких алгоритмов, который Баесенс с коллегами успешно реализовали в рамках исследования, опубликованного в 2003 г. [3]. Однако с течением времени стало ясно, что проведенные эксперименты начали терять свою актуальность по следующим причинам [4]:

1. В алгоритмах скоринга использованы в основном небольшие наборы данных и малое количество независимых показателей. В свою очередь, результатом развития информационных технологий является возможность получения доступа ко все большему объему данных с более широким набором исходных показателей.

2. Появились новые алгоритмы скоринга, сравнительные исследования по которым не проводились ранее.

3. Исследование было сфокусировано на анализе индивидуальных классификаторов (использующих какую-либо одну модель для получения результата), но с течением времени широкое распространение получили ансамблевые классификаторы (использующие множество моделей).

Поэтому Баесенс с группой исследователей провели новый сравнительный анализ алгоритмов скоринга, который был опубликован в 2015 г. [4]. В этом эксперименте рассмотрены три основные группы алгоритмов классификации: индивидуальные, однородные и разнородные ансамблевые классификаторы. По результатам анализа в финальную выборку попал и классификатор, основанный на логистической регрессии. Целью настоящей работы является исследование возможности эффективного применения логистической регрессии для решения задачи классификации займа.

Описание данных. Для решения задачи все используемые данные можно разделить на три группы: входные данные, настраиваемые параметры рассматриваемых методов и выходные данные.

Входные данные. Для настройки параметров и проведения экспериментов рассматриваемыми методами используются исторические данные по выданным на платформе для кредитования от человека человеку LendingClub займам[1], состоящие из 2 260 668 строк. Пусть m является отдельной позицией (строкой, которая соответствует определенному выданному займу). При этом для решения задачи были взяты все доступные позиции в количестве 2 260 668 (займы, выданные за период с апреля 2016 по сентябрь 2018 г.). Каждый займ характеризуется $n = 55$ независимыми показателями, которые удовлетворяют следующим свойствам:

– показатели были известны на момент принятия решения о выдаче займа;

– для них определены итоговые результаты по каждому займу;

– значения показателей определены (т. е. не отсутствуют) по всему перечню выданных займов.

Приведем названия и определения независимых показателей:

1. *loan_amnt* – запрошенная сумма займа.

2. *term* – количество платежей по займу в месяцах, что эквивалентно сроку займа.

3. *int_rate* – процентная ставка по займу.

4. *installment* – ежемесячный платеж по займу.

5. *emp_length* – продолжительность трудоустройства заемщика, выраженная в годах. Значения находятся в диапазоне от 0 до 10, где 0,5 означает занятость менее года, а 10 – занятость в течение 10 лет или выше. При этом допускается, что все значения n/a (not available) приравниваются к нулю. Таким образом, считается, что заемщики с этим значением не имеют трудового опыта.

6. *home_ownership* – информация о наличии в собственности недвижимости, которая была предоставлена заемщиком.

7. *annual_inc* – сумма годового дохода, представленная заемщиком при регистрации.

8. *verification_status* – статус, который определяет, что был проверен доход и источник дохода или проверка не проводилась.

9. *issue_d* – месяц, в котором займ был предоставлен.

10. *purpose* – назначение займа, которое указывается заемщиком.

11. *addr_state* – штат, в котором заемщик проживает.

---

[1]Loan data // Lending Club. – Mode of access: https://www.kaggle.com/datasets/wordsforthewise/lending-club. – Date of access: 04.09.2019.

12. *dti* – соотношение суммы всех месячных платежей по займам (за исключением ипотеки) и месячного дохода заемщика.

13. *delinq_2yrs* – количество просрочек продолжительностью более 30 дней по платежам за последние два года согласно кредитной истории заемщика.

14. *earliest_cr_line* – месяц, в котором заемщику была открыта первая кредитная линия.

15. *inq_last_6mths* – количество запросов на займ за последние шесть месяцев (за исключением запросов на кредит на покупку автомобиля и на ипотеку).

16. *open_acc* – количество открытых кредитных линий у заемщика согласно отчету о кредитной истории.

17. *pub_rec* – количество отрицательных публичных записей в кредитной истории заемщика.

18. *revol_bal* – общий кредитный оборотный баланс.

19. *revol_util* – отношение суммы, используемой заемщиком, ко всей доступной сумме револьверного (возобновляемого) кредита.

20. *total_acc* – общее количество кредитных линий (включая закрытые) у заемщика согласно отчету о кредитной истории.

21. *initial_list_status* – изначальный статус листинга займа: частичный или полный (т. е. заемщик возьмет займ, если будет предоставлена полностью запрошенная сумма).

22. *application_type* – определяет, является ли обращение за займом индивидуальным или совместным (с другими заемщиками).

23. *acc_now_delinq* – количество просроченных счетов у заемщика.

24. *tot_coll_amt* – общая сумма задолженности, когда-либо подвергнутая взысканию.

25. *tot_cur_bal* – итоговый текущий баланс всех счетов.

26. *total_rev_hi_lim* – общий лимит по револьверному (возобновляемому) кредиту.

27. *acc_open_past_24mths* – количество открытых кредитных линий за последние 24 месяца.

28. *avg_cur_bal* – средний текущий баланс всех счетов.

29. *bc_open_to_buy* – суммарная доступная сумма кредита по возобновляемым банковским картам.

30. *bc_util* – соотношение суммарного текущего баланса к кредитному лимиту по всем банковским картам.

31. *chargeoff_within_12_mths* – количество списаний (безнадежной задолженности) в течение последних 12 месяцев.

32. *mo_sin_old_rev_tl_op* – количество месяцев с момента открытия старейшего револьверного счета.

33. *mo_sin_rcnt_rev_tl_op* – количество месяцев с момента открытия последнего револьверного счета.

34. *mo_sin_rcnt_tl* – количество месяцев с момента открытия последнего счета.

35. *mort_acc* – количество ипотечных счетов.

36. *mths_since_recent_bc* – количество месяцев с момента открытия последнего карточного счета.

37. *num_accts_ever_120_pd* – количество счетов, по которым были просрочки платежей продолжительностью 120 дней и более.

38. *num_actv_bc_tl* – количество текущих активных карточных счетов.

39. *num_actv_rev_tl* – количество текущих активных возобновляемых кредитных счетов.

40. *num_bc_sats* – количество удовлетворительных карточных счетов (по которым погашения производились вовремя).

41. *num_bc_tl* – общее количество карточных счетов.

42. *num_il_tl* – количество счетов с периодическими платежами.

43. *num_op_rev_tl* – количество открытых возобновляемых кредитных счетов.

44. *num_rev_accts* – количество возобновляемых кредитных счетов.

45. *num_rev_tl_bal_gt_0* – количество возобновляемых кредитных счетов с балансом больше нуля.

46. *num_sats* – количество удовлетворительных счетов.

47. *num_tl_op_past_12m* – количество счетов, открытых за последние 12 месяцев.

48. *pct_tl_nvr_dlq* – процент счетов, по которым не было просрочек платежей.
49. *pub_rec_bankruptcies* – количество публично зарегистрированных банкротств.
50. *tax_liens* – количество налоговых залогов.
51. *tot_hi_cred_lim* – итоговый кредитный лимит.
52. *total_bal_ex_mort* – итоговый кредитный баланс за исключением ипотеки.
53. *total_bc_limit* – итоговый кредитный лимит по картам.
54. *total_il_high_credit_limit* – итоговый лимит по счетам с периодическими платежами.
55. *disbursement_method* – метод предоставления заемщику кредита: наличными средствами или прямым платежом.

Предполагается, что значения данных показателей были известны до принятия решения о выдаче соответствующего займа. Обозначим значение показателя j в займе i из исходного набора данных через элементы $x_j^{(i)}$ матрицы X размером n на m, где $j = 1, \ldots, n,\ i = 1, \ldots, m$. Обозначим через x_j столбец матрицы X, а через $x^{(i)}$ – строку матрицы X, которая содержит значения независимых показателей в отдельной позиции (займе) i набора данных.

Также в качестве исходных данных используются целевые значения $y^{(i)}$ (итоговый результат по займам, где $i = 1, \ldots, m$), которые определены в поле *loan_status* исходного набора данных. Показатель $y^{(i)}$ принимает два значения:

1. Возвратный займ (со значением *Fully Paid*). Такие займы были погашены. Соответствует значению $y^{(i)} = 1$.

2. Невозвратный займ (*Charged Off* или *Default*). Займы, по которым был объявлен дефолт или погашение займа просрочено более чем на 180 дней. Соответствует значению $y^{(i)} = 0$.

Займы со значениями *Current, In Grace period, Late* (16–30 *days*) и *Late* (31–120 *days*) исключаются из анализа, так как однозначно нельзя определить, будут они возвратными или невозвратными.

Параметры используемого алгоритма. В рассматриваемом алгоритме задействованы настраиваемые параметры:

θ_j – набор коэффициентов модели, где $j = 0, \ldots, n$. Эти коэффициенты определяются в результате решения задачи оптимизации с помощью алгоритма путем минимизации стоимостной функции, как описано далее;

$a_\theta(x)$ – функция активации.

Выходные данные. Предположим, что выходными данными поставленной ранее бинарной задачи классификации (т. е. определения займа как возвратного или потенциально невозвратного) являются величины $\hat{y}^{(i)} \in \{0, 1\}$, где единица соответствует возвратному, а ноль – потенциально невозвратному займу $i,\ i = 1, \ldots, m$.

Преобразование входных данных. Для решения бинарной задачи классификации с использованием рассматриваемого алгоритма необходимо, чтобы значения всех входных показателей были числами. В связи с тем что некоторые входные показатели имеют качественные и текстовые значения, следует провести их соответствующие преобразования:

Показатель *term* содержит текстовые значения. Его преобразование в числовой вид осуществляется в соответствии с табл. 1.

Таблица 1
Преобразование показателя *term*

Table 1
Conversion of term feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
36 *months*	1
60 *months*	2

Преобразование показателей *emp_length*, *verification_status*, *home_ownership* и *purpose* осуществляется в соответствии с табл. 2–5.

Таблица 2
Преобразование показателя *emp_length*

Table 2
Conversion of emp_length feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
n/a	0
< 1 *year*	0,5
1 *year*	1
2 *years*	2
3 *years*	3
4 *years*	4
5 *years*	5
6 *years*	6
7 *years*	7
8 *years*	8
9 *years*	9
10 + *years*	10

Таблица 3
Преобразование показателя *purpose*

Table 3
Conversion of purpose feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
Car	1
credit_card	2
debt_consolidation	3
home_improvement	4
House	5
major_purchase	6
Medical	7
Moving	8
Other	9
renewable_energy	10
small_business	11
Vacation	12
Wedding	13

Таблица 4
Преобразование показателя *home_ownership*

Table 4
Conversion of home_ownership feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
ANY	1
MORTGAGE	2
OWN	3
RENT	4

Таблица 5
Преобразование показателя *verification_status*

Table 5
Conversion of verification_status feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
Source Verified	1
Verified	2
Not Verified	3

Так как показатели *earliest_cr_line* и *issue_d* содержат месяц и год выдачи первого и текущего займа, которые по отдельности мало полезны для решения поставленной задачи, то в качестве новых значений показателя *earliest_cr_line* примем разницу между месяцем выдачи текущего займа (*issue_d*) и месяцем, в котором заемщику была открыта первая кредитная линия (*earliest_cr_line*). Соответственно, после преобразования показатель *issue_d* удаляется.

Преобразование показателей *addr_state*, *initial_list_status*, *application_type* и *disbursement_method* осуществляется в соответствии с табл. 6–9.

Таблица 6
Преобразование показателя *addr_state*

Table 6
Conversion of addr_state feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*	Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
AK	1	*MT*	30
AL	2	*NA*	31
AR	3	*NC*	32
AS	4	*ND*	33
AZ	5	*NE*	34
CA	6	*NH*	35
CO	7	*NJ*	36
CT	8	*NM*	37
DC	9	*NV*	38
DE	10	*NY*	39
FL	11	*OH*	40
GA	12	*OK*	41
GU	13	*OR*	42
HI	14	*PA*	43
IA	15	*PR*	44
ID	16	*RI*	45
IL	17	*SC*	46
IN	18	*SD*	47
KS	19	*TN*	48
KY	20	*TX*	49
LA	21	*UT*	50
MA	22	*VA*	51
MD	23	*VI*	52
ME	24	*VT*	53
MI	25	*WA*	54
MN	26	*WI*	55
MO	27	*WV*	56
MP	28	*WY*	57
MS	29		

Таблица 7
Преобразование показателя *initial_list_status*

Table 7
Conversion of initial_list_status feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
W	1
F	2

Таблица 8
Преобразование показателя *application_type*

Table 8
Conversion of application_type feature

Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
Joint App	1
Individual	2

Таблица 9
Преобразование показателя *disbursement_method*

Table 9
Conversion of disbursement_method feature

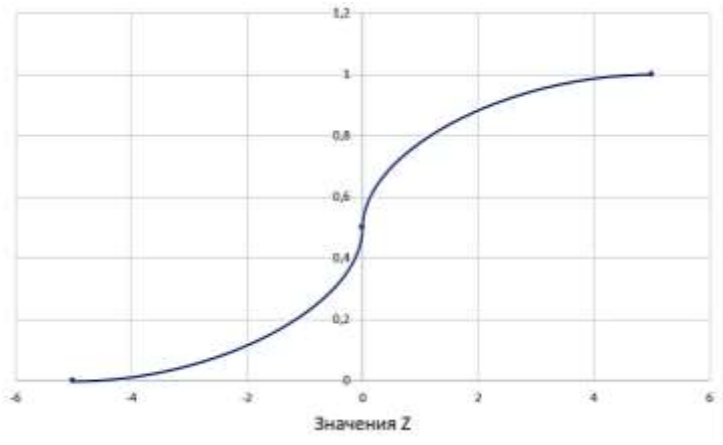
Исходное текстовое значение *Original text value*	Преобразованное альтернативное числовое значение *Converted alternative numeric value*
Cash	1
DirectPay	2

Результатом выполненных преобразований является конечный набор входных данных в числовом виде, который состоит из m = 1 221 731 позиций и n = 54 входных показателей.

Постановка задачи. Значения функции логистической регрессии (рисунок) всегда находятся в диапазоне $0 \le a_\theta(x^{(i)}) \le 1$. Для бинарной задачи классификации займа i под значением $a_\theta(x^{(i)})$ можно понимать вероятность возврата. При этом значение функции определяется выражением [5]

$$a_\theta(x^{(i)}) = \frac{1}{1+e^{-z}}, \quad (1)$$

где $Z_\theta(x^{(i)}) = \theta_0 + \theta_1 \cdot x_1^{(i)} + \cdots + \theta_n \cdot x_n^{(i)}$.



Логистическая (сигмовидная) функция

Logistic (sigmoid) function

Таким образом, функция $a_\theta(x^{(i)})$ определяет вероятность того, что значение выходного показателя $\hat{y}$ будет равно единице, т. е. предоставляемый займ считается возвратным.

Для того чтобы иметь возможность классифицировать займ и определить значение выходного показателя $\hat{y}^{(i)} \in \{0, 1\}$ с учетом симметричности логистической функции относительно значения 0,5, примем следующие неравенства [6]:

$$a_\theta(x^{(i)}) \ge 0{,}5 \rightarrow \hat{y}^{(i)} = 1;$$
$$a_\theta(x^{(i)}) < 0{,}5 \rightarrow \hat{y}^{(i)} = 0.$$

В качестве альтернативы алгоритм может определять вероятность возврата займа от 0 до 1 и предоставлять сотруднику возможность классифицировать займ как возвратный или невозвратный.

Для определения параметров θ_j используется стоимостная функция, которую требуется минимизировать. Данную функцию можно представить в виде равенства[2]

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right), \quad (2)$$

где $x^{(i)}$ и $y^{(i)}$ – заданные показатели, θ – вектор неизвестных переменных. Задачи минимизации невыпуклых функций весьма сложные. Стоимостная функция от переменных θ будет выпуклой, если функцию $Cost$ представить следующим образом [6]:

$$Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right) := \begin{cases} -ln\left(a_\theta\left(x^{(i)}\right)\right), \text{если } y^{(i)} = 1; \\ -ln\left(1 - a_\theta\left(x^{(i)}\right)\right), \text{если } y^{(i)} = 0. \end{cases} \quad (3)$$

Функция (3) обладает свойствами

$$\begin{cases} Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right) \to 0, \text{если } y^{(i)} = a_\theta\left(x^{(i)}\right); \\ Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right) \to \infty, \text{если } y^{(i)} = 0 \text{ и } a_\theta\left(x^{(i)}\right) \to 1; \\ Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right) \to \infty, \text{если } y^{(i)} = 1 \text{ и } a_\theta\left(x^{(i)}\right) \to 0. \end{cases} \quad (4)$$

Формулу (4) можно представить в более компактном виде[3]:

$$Cost\left(a_\theta\left(x^{(i)}\right), y^{(i)}\right) = -y^{(i)} \cdot ln\left(a_\theta\left(x^{(i)}\right)\right) - \left(1 - y^{(i)}\right) \cdot ln\left(1 - a_\theta\left(x^{(i)}\right)\right). \quad (5)$$

В результате полная стоимостная функция, которая учитывает весь набор данных *m*, рассчитывается по формуле [6]

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} \cdot ln\left(a_\theta\left(x^{(i)}\right)\right) + \left(1 - y^{(i)}\right) \cdot ln\left(1 - a_\theta\left(x^{(i)}\right)\right)\right]. \quad (6)$$

В задачах подобного типа при большом количестве входных показателей часто возникает проблема переобучения, при которой функция $a_\theta\left(x^{(i)}\right)$ достаточно точно описывает имеющийся набор данных, но не очень полезна для прогнозирования, так как не справляется с задачей обобщения имеющегося набора данных.

При решении проблемы переобучения можно либо вручную выбрать некоторые из имеющихся входных показателей для расчета функции $a_\theta\left(x^{(i)}\right)$, либо задействовать l_2-регуляризацию [7], при которой включаются в расчет все имеющиеся входные показатели, специальным образом уменьшаются величины соответствующих им коэффициентов θ_j и изменяется функция $J(\theta)$. Стоит отметить, что коэффициент θ_0 не подлежит изменению, так как не связан с каким-либо входным показателем.

В настоящей работе при необходимости используется l_2-регуляризация и к стоимостной функции[4] добавляется $\frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$, где λ – параметр регуляризации. В результате итоговая стоимостная функция рассчитывается следующим образом:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} \cdot ln\left(a_\theta\left(x^{(i)}\right)\right) + \left(1 - y^{(i)}\right) \cdot ln\left(1 - a_\theta\left(x^{(i)}\right)\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2. \quad (7)$$

---

[2]Cost function for logistic regression // Supervised Machine Learning: Regression and Classification. – Mode of access: https://www.coursera.org/learn/machine-learning/lecture/0hpr8/cost-function-for-logistic-regression. – Date of access: 16.11.2019.

[3]Simplified Cost Function for Logistic Regression // Supervised Machine Learning: Regression and Classification. – Mode of access: https://www.coursera.org/learn/machine-learning/lecture/Zjj2j/simplified-cost-function-for-logistic-regression. – Date of access: 16.11.2019.

[4]Regularized logistic regression // Supervised Machine Learning: Regression and Classification. – Mode of access: https://www.coursera.org/learn/machine-learning/lecture/cAxpF/regularized-logistic-regression. – Date of access: 16.11.2019.

Далее для минимизации стоимостной функции применяется метод градиентного спуска [8], посредством которого определяются оптимальные коэффициенты θ_j. Изначально для этого коэффициентам θ_j присваиваются нулевые значения. Затем путем последовательного изменения данных коэффициентов оптимизируется (в данном случае уменьшается) стоимостная функция $J(\theta)$. Таким образом, согласно определению метода градиентного спуска требуется последовательно повторять шаги по обновлению коэффициентов θ_j с помощью следующей формулы до тех пор, пока стоимостная функция не достигнет своего минимального значения[5]:

$$\theta_j := \theta_j - \alpha \cdot \frac{d}{d\theta_j} J(\theta). \quad (8)$$

При этом все коэффициенты θ_j должны обновляться одновременно, а параметр α определяет размер шага градиентного спуска.

Учитывая, что θ_0 не подлежит регуляризации, и определив частную производную стоимостной функции, пошаговое нахождение оптимальных коэффициентов θ_j с помощью метода градиентного спуска можно осуществить посредством преобразованных формул

$$\theta_0 := \theta_0 - \alpha \cdot \left[\frac{1}{m}\sum_{i=1}^{m}\left(a_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x_0^{(i)}\right], \quad (9)$$

$$\theta_j := \theta_j - \alpha \cdot \left[\frac{1}{m}\sum_{i=1}^{m}\left(a_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x_j^{(i)} + \frac{\lambda}{m} \cdot \theta_j\right], \; j=1, \ldots, n. \quad (10)$$

Для проведения экспериментов использовался объектно-ориентированный язык программирования *Python version* 3.7.5[6]. Оптимальное значение α вычислялось на основе экспериментов.

После нахождения коэффициентов θ_j необходимо рассчитать точность прогнозирования, т. е. с использованием оптимальных коэффициентов θ_j и входных показателей x_j определить целевые величины $\hat{y}^{(i)}$ по всему набору данных для обучения и сравнить с известными целевыми значениями $y^{(i)}$. Далее требуется провести оценку эффективности данной логистической регрессии. Для этого необходимо, используя $\hat{y}^{(i)}$ и $y^{(i)}$, рассчитать четыре основных параметра [9]: истинно положительный (*TP*), ложноположительный (*FP*), истинно отрицательный (*TN*) и ложноотрицательный (*FN*). На основе данных параметров строится матрица несоответствий (табл. 10).

Таблица 10
Матрица несоответствий

Table 10
Confusion matrix

Прогнозное значение $\hat{y}^{(i)}$ *Forecast value* $\hat{y}^{(i)}$	Тестовое значение $y^{(i)}$ *Test value* $y^{(i)}$	
	Класс 1 *Class 1*	Класс 0 *Class 0*
Класс 1	Истинно положительный	Ложноположительный
Класс 0	Ложноотрицательный	Истинно отрицательный

[5]Gradient Descent Implementation // Supervised Machine Learning: Regression and Classification. – Mode of access: https://www.coursera.org/learn/machine-learning/lecture/Ha1RP/gradient-descent-implementation. – Date of access: 16.11.2019.

[6]Python version 3.7.5 // Python Software Foundation. – Mode of access: https://www.python.org/downloads/release/python-375/. – Date of access: 21.10.2019.

С помощью матрицы рассчитывается базовый коэффициент эффективности *Accuracy* (A), который означает отношение количества правильно спрогнозированных классов ко всему набору спрогнозированных значений, т. е. $A = (TP + TN) / (TP + FP + TN + FN)$ [10]. Данный коэффициент отражает общую эффективность прогнозирования регрессии, но не учитывает возможный дисбаланс в распределении классов по всему набору данных, т. е. не учитывает, что в имеющемся наборе данных количество позиций класса 1 существенно больше, чем класса 0. Таким образом, для более детального измерения эффективности нейронной сети на основе матрицы несоответствий определяются дополнительные метрики:

Precision (P) – точность, или соотношение правильно спрогнозированных положительных классов к общему количеству положительно спрогнозированных классов, т. е. $P = TP / (TP + +FP)$. В контексте рассматриваемой задачи означает, какой процент займов, спрогнозированных как возвратные, действительно таковыми являются.

Recall (R) – полнота, также известная как чувствительность, или соотношение правильно спрогнозированных положительных классов к общему количеству действительно положительно классов: $R = TP / (TP + FN)$. В данном случае показывает, какая доля возвратных займов предсказана верно.

Поскольку однозначный ответ о превосходстве какой-либо из метрик P и R отсутствует, для их совместного использования при оценке эффективности прогнозирования определяется мера F_1 как среднее гармоническое метрик P и R по формуле $F_1 = (2 \cdot P \cdot R) / (P + R)$. Однако при решении задачи многоклассовой классификации мера F_1 может быть рассчитана с помощью класса *sklearn.metrics.classification_report*[7] как микроусредненная величина (без учета разделения данных на классы), макроусредненная величина (рассчитываются значения F_1 для каждого класса в отдельности, а далее находится среднее), а также как средневзвешенная величина (рассчитываются значения F_1 для каждого класса, а затем общее значение F_1 с учетом веса каждого класса в наборе данных). В связи с тем что значение коэффициента эффективности A равняется микроусредненному значению F_1, микроусредненной величине P и микроусредненной величине R, а эти макроусредненные величины не учитывают вес каждого из классов в наборе исходных данных, для оценки эффективности будет использоваться коэффициент эффективности A и средневзвешенная мера F_1 на основе средневзвешенных значений P и R. Чем больше значения данных метрик, тем выше точность прогнозирования модели. Для получения метрик весь набор исходных данных разделяется на два множества: тренировочный и тестовый наборы данных. Первый используется для обучения, а второй – для проверки точности прогнозирования, так как наиболее корректной является проверка точности с использованием данных, которые алгоритм еще не обрабатывал. При этом тренировочный набор данных будет состоять из 70 %, а тестовый набор – из 30 % от всего множества займов.

Нормализация исходных данных. Многие входные показатели x_j имеют значения, которые сильно отличаются по своему диапазону. Например, показатель ‘*loan_amnt*’ имеет значения в диапазоне (1000, 40 000), а значения показателя ‘*annual_inc*’ находятся в интервале (16, 10 999 200). Такие различия в диапазонах величин могут привести к сложностям для некоторых алгоритмов машинного обучения [11] или более медленному выполнению метода градиентного спуска[8]. Это связано с тем, что метод может сходиться быстрее к точке экстремума при меньшем диапазоне значений входных показателей либо найти более оптимальное значение.

Для решения обозначенной проблемы необходимо привести значения входных показателей (m-мерных векторов-столбцов $x_j = (x_j^{(1)}, x_j^{(2)}, \ldots, x_j^{(m)})^T$) к примерно одинаковому диапазону (в векторном виде удовлетворяющему соотношению $-1 \leq x_j \leq 1$ для большинства значений

[7]Sklearn.decomposition.classification_report // Sklearn Decomposition. – Mode of access: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html#sklearn.metrics.classification_report. – Date of access: 20.11.2022.

[8]Feature scaling part 1 // Supervised Machine Learning: Regression and Classification. – Mode of access: https://www.coursera.org/learn/machine-learning/lecture/KMDV3/feature-scaling-part-1. – Date of access: 25.11.2019.

векторов). Для этого используется инструмент средней нормализации, при котором значения каждого входного показателя изменяются в соответствии с векторной формулой [11]

$$x_j := \frac{x_j - \mu_j}{sd(x_j)}, \quad (11)$$

где μ_j – среднее значение всех компонент вектора x_j, а $sd(x_j)$ – стандартное отклонение компонент вектора x_j. При этом средняя нормализация применяется только к непрерывным[9] показателям, а к номинальным[10], ординальным[11] и категориальным[12] показателям (в данном случае *term, verification_status, home_ownership, purpose, addr_state, initial_list_status, application_type, disbursement_method*) не применяется. Все значения данных показателей делятся на максимальное число этих же показателей, что позволяет привести значения показателей к диапазону от нуля до единицы, т. е. делает их размерность сопоставимой с количественными показателями.

Для оценки влияния использования нормализации входных показателей на результаты машинного обучения в рамках логистической регрессии и для рассматриваемой задачи были проведены компьютерные исследования, состоящие из 10 000 итераций (количество итераций алгоритма до остановки оптимизации). В рамках данных исследований для каждого из вариантов (с нормализацией и без нормализации) также было найдено значение α, которое обеспечило минимальное значение стоимостной функции на основе логарифмической шкалы, предполагающей, что следующее число получается умножением предыдущего на 10. При этом диапазон исследования для α составил от 1e-10 до 10. Полученные результаты представлены в табл. 11 и 12.

Таблица 11
Результаты эксперимента при использовании нормализации
Table 11
Experiment results when using normalization

Исследуемый параметр *Parameter under study*	Значение без нормализации *Value without normalization*	Значение с нормализацией *Normalized value*
Оптимальный α	1e-10	1
Длительность обучения алгоритма (с)	0,153 65	0,180 76
Значение стоимостной функции	0,501 43	0,458 00
Accuracy training (%)	79,646 78	79,906 36
Accuracy testing (%)	79,727 44	80,000 82

Таблица 12
Ключевые метрики при использовании нормализации на тестовых данных
Table 12
Key metrics when using normalization on test data

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,538 95	0,092 92	0,158 51
Возвратные займы	0,809 46	0,979 79	0,886 52
Средневзвешенное	0,754 63	0,800 01	0,738 94

[9]Continuous or discrete variable // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Continuous_or_discrete_variable. – Date of access: 19.11.2022.

[10]Level of measurement // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Level_of_measurement#Nominal_level. – Date of access: 19.11.2022.

[11]Ordinal data // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Ordinal_data. – Date of access: 19.11.2022.

[12]Categorical variable // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Categorical_variable. – Date of access: 19.11.2022.

По результатам использования указанных ранее входных показателей с исходными данными для решения задачи классификации займа можно сделать следующие выводы:

1. Применение регуляризации на этапе нормализации нецелесообразно, так как значения коэффициента эффективности A, полученные на тестовых и тренировочных данных, отличаются незначительно, что свидетельствует об отсутствии проблемы переобучения.

2. Использование нормализации уменьшает значение стоимостной функции на 8,66 % и улучшает точность прогнозирования (значение A) на 0,27 %.

3. Значение P = 0,80946 для возвратных займов означает, что 80,946 % займов, определенных с помощью модели как возвратные, действительно таковыми являются. При этом значение R = 0,97979 свидетельствует о том, что при рассмотрении всего множества действительно возвратных займов модель правильно классифицировала 97,979 % из них.

Очевидно, что нормализация ведет к улучшению точности прогнозирования. Поэтому в дальнейших компьютерных экспериментах будет использован именно такой вариант.

Не менее важным является определение коэффициентов θ_j (j =1, …, n) для каждого из входных векторных показателей x_j. По результатам проведенного исследования рассчитаны значения θ_j (табл. 13).

Таблица 13
Значения θ_j

Table 13
θ_j values

Номер показателя *Indicator number*	θ_j	ID показателя *Indicator ID*	Номер показателя *Indicator number*	θ_j	ID показателя *Indicator ID*
1	–0,075 79	*loan_amnt*	28	0,079 38	*avg_cur_bal*
2	–1,232 21	*term*	29	–0,033 85	*bc_open_to_buy*
3	–0,344 14	*int_rate*	30	–0,041 07	*bc_util*
4	–0,073 28	*installment*	31	0,000 31	*chargeoff_within_12_mths*
5	0,074 93	*emp_length*	32	0,134 07	*mo_sin_old_rev_tl_op*
6	–0,680 81	*home_ownership*	33	–0,002 03	*mo_sin_rcnt_rev_tl_op*
7	0,063 93	*annual_inc*	34	0,021 66	*mo_sin_rcnt_tl*
8	0,217 13	*verification_status*	35	0,055 89	*mort_acc*
9	0,586 29	*disbursement_method*	36	0,040 98	*mths_since_recent_bc*
10	–0,380 98	*purpose*	37	–0,003 42	*num_accts_ever_120_pd*
11	–0,001 91	*addr_state*	38	0,009 26	*num_actv_bc_tl*
12	–0,168 92	*dti*	39	–0,085 44	*num_actv_rev_tl*
13	–0,074 16	*delinq_2yrs*	40	–0,064 60	*num_bc_sats*
14	–0,142 87	*earliest_cr_line*	41	0,025 55	*num_bc_tl*
15	–0,048 90	*inq_last_6mths*	42	–0,068 90	*num_il_tl*
16	–0,007 80	*open_acc*	43	0,042 27	*num_op_rev_tl*
17	–0,027 50	*pub_rec*	44	–0,077 29	*num_rev_accts*
18	–0,000 35	*revol_bal*	45	–0,051 10	*num_rev_tl_bal_gt_0*
19	–0,023 65	*revol_util*	46	0,039 87	*num_sats*
20	0,202 42	*total_acc*	47	–0,012 82	*num_tl_op_past_12m*
21	0,201 88	*initial_list_status*	48	0,008 09	*pct_tl_nvr_dlq*
22	–0,198 58	*application_type*	49	–0,001 17	*pub_rec_bankruptcies*
23	–0,009 82	*acc_now_delinq*	50	0,008 20	*tax_liens*
24	0,002 06	*tot_coll_amt*	51	0,009 95	*tot_hi_cred_lim*
25	0,024 63	*tot_cur_bal*	52	–0,164 83	*total_bal_ex_mort*
26	0,113 75	*total_rev_hi_lim*	53	0,132 93	*total_bc_limit*
27	–0,162 46	*acc_open_past_24mths*	54	0,172 28	*total_il_high_credit_limit*

Как следует из результатов эксперимента, $\theta_2 = -1{,}232\,21$ является коэффициентом с наибольшим абсолютным значением. Таким образом, показатель x_2 (срок займа) оказывает наибольшее отрицательное влияние на результат функции $a_\theta\left(x^{(i)}\right)$. Соответственно, чем больше срок кредита, тем менее вероятно, что займ будет классифицирован как возвратный.

Влияние сбалансированности исторических целевых значений на классификацию займов. Как было указано ранее, набор входных данных состоит из 1 221 731 позиции (заявки на займ). Однако возвратным займам соответствует 973 421 (~ 79,7 %) позиция, а невозвратным – 248 310 (~ 20,3 %) позиций. Следовательно, набор входных данных не сбалансирован по целевым значениям и смещен в сторону позиций с возвратными займами.

Так как некоторые подходы к машинному обучению показывают лучшие результаты при обучении сбалансированными данными [11], необходимо провести анализ влияния сбалансированности входных данных на результаты прогнозирования в рамках рассматриваемой задачи. Для этого из входного набора данных создается подмассив [8], который состоит из всех 248 310 позиций входных данных, соответствующих невозвратным займам, и только из 248 310 позиций, соответствующих возвратным займам. В итоге набор входных данных в подмассиве будет сбалансирован, но общее количество позиций уменьшится до 496 620.

Результаты компьютерного эксперимента, состоящего из 10 000 итераций при $\alpha = 1$, представлены в табл. 14 и 15.

Таблица 14
Результаты эксперимента при сбалансированности исторических целевых значений

Table 14
Experiment results when the historical target values are balanced

Исследуемый параметр *Parameter under study*	Значение *Value*
Длительность обучения алгоритма (с)	0,072 73
Значение стоимостной функции	0,590 68
Accuracy training (%)	68,677 98
Accuracy testing (%)	68,882 31

Таблица 15
Ключевые метрики при использовании сбалансированных данных

Table 15
Key metrics when using balanced data

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,691 74	0,679 36	0,685 49
Возвратные займы	0,686 02	0,698 26	0,692 09
Средневзвешенное	0,688 87	0,688 82	0,688 79

По итогам проведенного исследования можно сделать вывод, что абсолютная сбалансированность не привела к улучшению значения стоимостной функции, величин A и меры F_1 модели при использовании предложенного алгоритма машинного обучения в задаче классификации займа. Уменьшение точности прогнозирования и увеличение значения стоимостной функции объясняются существенным уменьшением набора входных позиций с 1 221 731 до 496 620 в связи с намерением сбалансировать набор входных данных. Вместе с тем следует отметить улучшение метрик P, R и F_1 для невозвратных займов. Поэтому применять сбалансированные входные данные можно в случае, когда точность прогнозирования невозвратных займов более важна, чем возвратных. Учитывая, что величины A и F_1 модели оказались хуже значений, полученных при отсутствии сбалансированности, в дальнейшем будет использован весь набор входных данных, состоящий из 1 221 731 позиции.

Классификация займов при увеличении входных показателей. При формировании входных данных задействовались показатели с определенными значениями по всему перечню вы-

данных займов. Однако в первоначальном наборе имелись показатели, которые были известны на момент выдачи займа, но имели до 30 % отсутствующих значений. Задействование этих входных показателей может привести к увеличению точности прогнозирования. Таким образом, необходимо оценить влияние исключенных из-за неполноты данных, но известных на момент выдачи входных показателей. Для этого требуется преобразовать входные показатели путем заполнения отсутствующих значений следующим образом: так как все показатели содержат количественные значения, то пустые позиции буду заменены поочередно на модальную величину соответствующего показателя, а также на среднее и медианное значения. Далее входные показатели нормализуются аналогично другим непрерывным показателям, как было описано ранее. Впоследствии требуется провести исследования и представить сравнительный анализ данных трех вариантов.

Приведем перечень дополнительных входных показателей:

mths_since_last_delinq – количество месяцев с момента последней просрочки;

mths_since_last_record – количество месяцев с момента последней публичной записи;

open_acc_6m – количество открытых кредитных счетов за последние шесть месяцев;

open_act_il – количество текущих активных счетов с рассрочкой платежа;

open_il_12m – количество счетов с рассрочкой платежа, открытых за последние 12 месяцев;

open_il_24m – количество счетов с рассрочкой платежа, открытых за последние 24 месяца;

mths_since_rcnt_il – количество месяцев с момента открытия последнего счета с рассрочкой платежа;

total_bal_il – текущий баланс по всем счетам с рассрочкой платежа;

il_util – соотношение суммарного текущего баланса к кредитному лимиту по всем счетам с рассрочкой;

open_rv_12m – количество револьверных счетов, открытых за последние 12 месяцев;

open_rv_24m – количество револьверных счетов, открытых за последние 24 месяца;

max_bal_bc – максимальный текущий баланс задолженности по всем револьверным счетам;

all_util – соотношение баланса к кредитному лимиту по всем счетам;

inq_fi – количество персональных финансовых запросов;

total_cu_tl – количество финансовых счетов;

inq_last_12m – количество запросов на кредит за последние 12 месяцев;

mo_sin_old_il_acct – количество месяцев со времени открытия самого старого счета с рассрочкой платежа;

mths_since_recent_inq – количество месяцев с момента последнего запроса;

percent_bc_gt_75 – процент всех счетов по банковским картам, которые превышают 75 % лимита.

При проведении исследования были использованы модальные, средние и медианные значения соответствующих дополнительных показателей для устранения пустых позиций и рассчитана точность прогнозирования для каждого случая при решении задачи классификации займа. Результаты исследования представлены в табл. 16 и 17.

Таблица 16
Результаты исследования при увеличении количества входных показателей

Table 16
Research results with an increase in the number of input features

Исследуемый параметр *Parameter under study*	Модальные значения *Modal values*	Средние значения *Averages*	Медианные значения *Median values*
Длительность обучения алгоритма (с)	0,223 65	0,208 26	0,219 02
Значение стоимостной функции	0,456 66	0,457 07	0,457 05
Accuracy training (%)	79,906 36	79,930 57	79,914 66
Accuracy testing (%)	79,985 54	80,027 28	80,005 73

Таблица 17
Ключевые метрики при заполнении средними значениями

Table 17
Key metrics when filled with averages

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,541 73	0,095 67	0,162 62
Возвратные займы	0,809 87	0,979 42	0,886 61
Средневзвешенное	0,755 52	0,800 27	0,739 85

Как следует из результатов эксперимента, включение дополнительных показателей в набор выходных исходных данных и их преобразование с помощью средних и медианных значений привели к улучшению точности прогнозирования по сравнению с точностью $A = 80{,}000\,82$ %, определенной ранее. При этом преобразование с помощью средних значений привело к лучшим результатам, чем преобразование с помощью модальных и медианных значений. Относительно стоимостной функции все три варианта показали лучший результат, чем полученный ранее. Поэтому в дальнейших исследованиях будут использоваться показатели, у которых отсутствующие значения заполнены средними величинами, а количество входных показателей n станет равным 73.

Классификация займов при различных граничных значениях. Функция логистической регрессии является симметричной относительно значения 0,5 (см. рисунок). Поэтому это значение было выбрано как граничное для определения того, будет займ возвратным или же невозвратным. Однако данная постановка задачи не может исключать возможности существования более подходящего пограничного значения при решении текущей задачи. В связи с этим требуется провести исследование влияния различных граничных значений на результаты решения задачи классификации займа. Следовательно, необходимо проанализировать влияние разных граничных значений от 0,01 до 1 с шагом 0,01 на точность прогнозирования при классификации займа на основе 10 000 итераций обучения алгоритма логистической регрессии. По результатам обучения находится оптимальное пограничное значение, которому соответствует наибольшая точность прогнозирования, выраженная значением A на тестовых данных.

Результаты анализа показали, что оптимальным граничным значением являлась величина 0,49. При этом значении средняя точность прогнозирования на тренировочных данных составила 79,93267 %, а на тестовых – 80,028102 %. Граничные значения не оказывают влияние на величину стоимостной функции.

Применительно к текущей задаче полученная точность в некоторой степени больше точности 80,027 28 %, рассчитанной при использовании пограничного значения 0,5. Поэтому при дальнейшем анализе предложенного алгоритма машинного обучения будет применяться граничное значение 0,49.

Таким образом, выявлено, что исследование различных граничных значений при решении задачи классификации займа является целесообразным. Вместе с тем из табл. 18 следует, что значения метрик P, R и F_1 также изменились. В частности, значение F_1 всей модели ухудшилось до 0,73767 в сравнении с 0,73985. Следовательно, оптимальное граничное значение модели может варьироваться в зависимости от показателя, выбранного для оптимизации.

Таблица 18
Ключевые метрики при оптимальном граничном значении

Table 18
Key metrics at the optimal boundary value

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,542 46	0,087 62	0,151 01
Возвратные займы	0,808 83	0,981 48	0,886 83
Средневзвешенное	0,755 56	0,800 28	0,737 67

Классификация займов при использовании полиномиальных показателей. Согласно результатам исследований, проведенных ранее, увеличение количества показателей входных данных может привести к улучшению ключевых показателей рассматриваемого алгоритма обучения. Альтернативным вариантом увеличения числа входных показателей является создание дополнительных входных показателей на основе каждого из уже имеющихся показателей с использованием полиномиальной регрессии. В упрощенном варианте, предполагающем наличие одного входного показателя, полиномиальная регрессия выражается следующим образом [5]:

$$Z\left(x^{(i)}\right) = \theta_0 + \theta_1 \cdot x_1^{(i)1} + \theta_2 \cdot x_1^{(i)2} + \cdots + \theta_p \cdot x_1^{(i)p}, \quad (12)$$

где p – степень полинома. В результате каждый входной показатель может быть возведен поочередно до степени p и добавлен к исходному набору входных данных. При этом регрессия приобретает более сложный нелинейный характер.

Для решения задачи классификации займа в качестве эксперимента имеющийся набор входных показателей x_j требуется расширить с помощью полинома от второй до пятой степени включительно и сравнить результаты. Учитывая, что состав входных данных изменился, величина $\alpha = 1$ может перестать быть оптимальной. Поэтому для каждого полинома необходимо найти оптимальное значение α (соответствующее минимальному значению стоимостной функции) в диапазоне от 1e-10 до 10 на основе логарифмической шкалы аналогично подходу, использованному ранее.

Результаты компьютерного эксперимента, состоящего из 10 000 итераций для каждой степени полинома, представлены в табл. 19.

Таблица 19
Результаты исследования при использовании полиномиальных показателей

Table 19
Research results when using polynomial features

Исследуемый параметр *Parameter under study*	Полином второй степени *Second degree polynomial*	Полином третьей степени *Third degree polynomial*	Полином четвертой степени *Fourth degree polynomial*	Полином пятой степени *Fifth degree polynomial*
α	1e-2	1e-4	1e-6	1e-10
Длительность обучения алгоритма (с)	0,346 42	0,433 01	0,532 92	0,663 71
Значение стоимостной функции	0,455 68	0,493 49	0,646 89	0,704 09
Accuracy training (%)	79,899 58	79,640 11	79,384 04	79,248 74
Accuracy testing (%)	79,959 62	79,708 88	79,449 42	79,312 45

Таблица 20
Ключевые метрики при использовании полинома второй степени

Table 20
Key metrics when using of the second degree polynomial

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,548 34	0,064 66	0,115 68
Возвратные займы	0,805 75	0,986 46	0,886 99
Средневзвешенное	0,753 57	0,799 60	0,730 64

Согласно результатам анализа увеличение количества исходных входных показателей с использованием полинома второй степени привело к улучшению (уменьшению) значения стоимостной функции. Вместе с тем из табл. 20 следует, что длительность алгоритма обучения

существенно увеличилась, а значения коэффициента эффективности A и средневзвешенной меры F_1 оказались ниже полученных ранее. Поэтому дальнейшее увеличение количества входных показателей на основе полиномиальной регрессии нецелесообразно.

Использование метода главных компонент для задачи классификации займа. Как следует из полученных ранее результатов, увеличение количества входных показателей с 54 до 73 привело к улучшению метрик исследуемой модели. Учитывая, что дальнейшее увеличение количества показателей с применением полиномов не привело к улучшению точности прогнозирования, целесообразно проанализировать имеющийся набор входных показателей на избыточность. Основными причинами устранения избыточности являются [8]: упрощение использования набора данных, уменьшение вычислительных затрат и шума в данных. Целью исследования является уменьшение размерности входных данных без ухудшения значений основных результирующих показателей рассматриваемой модели. Для такого исследования будет использован метод главных компонент (*principal component analysis*)[13] [7, 8], суть которого заключается в уменьшении линейной размерности с использованием разложения по сингулярным значениям[14]. Для последовательного расчета главных компонент (начиная с первого и до заданного количества) будет использован класс *sklearn.decomposition.PCA*[15]. Главные компоненты берутся в диапазоне от 1 до 73 включительно с целью выявления их оптимального количества, которое обеспечит максимальное значение коэффициента эффективности A на тестовых данных.

Результаты проведенных расчетов показали, что оптимальное количество главных компонент равнялось 72, значения остальных метрик приведены в табл. 21 и 22.

Таблица 21
Результаты исследования при использовании метода главных компонент

Table 21
Results of the study using the method of principal components

Исследуемый параметр *Parameter under study*	Значение *Value*
Длительность обучения алгоритма (с)	0,206 32
Значение стоимостной функции	0,457 06
Accuracy training (%)	79,934 08
Accuracy testing (%)	80,040 65

Таблица 22
Ключевые метрики при использовании метода главных компонент

Table 22
Key metrics when using principal component analysis

Класс *Class*	*Precision*	*Recall*	Мера F_1 *Measure* F_1
Невозвратные займы	0,547 79	0,088 25	0,152 01
Возвратные займы	0,808 94	0,981 48	0,886 89
Средневзвешенное	0,756 00	0,800 41	0,737 92

Из полученных результатов следует, что применение метода главных компонент привело к получению наилучшего значения коэффициента эффективности $A = 80{,}0401$ %. Отсюда можно сделать вывод, что использование данного метода является целесообразным при решении

[13]Principal component analysis // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Principal_component_analysis. – Date of access: 22.12.2022.
[14]Principal component analysis (PCA) // Sklearn Decomposition. – Mode of access: https://scikit-learn.org/stable/modules/decomposition.html#decompositions. – Date of access: 22.12.2022.
[15]Sklearn.decomposition.PCA // Sklearn Decomposition. – Mode of access: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html. – Date of access: 22.12.2022.

задачи классификации займа. При этом значение $R = 98{,}148$ % для возвратных займов означает, что модель правильно классифицировала 98,148 % из них.

Так как количество входных показателей равно 73, а оптимальное количество главных компонент 72, то можно сделать вывод, что исходное пространство показателей в основном не является избыточным. Вместе с тем обнаружено, что применение метода главных компонент с количеством компонент, равным 55, покрывает минимум 99 % дисперсии входных данных. Этот факт может быть полезен при необходимости уменьшения вычислительных затрат.

Заключение. В работе представлены принципы формирования и преобразования данных для задачи классификации займов и рассмотрено применение алгоритма логистической регрессии для ее решения. Выявлено, что использование нормализации улучшает точность прогнозирования при оптимальной величине $\lambda = 1$, а абсолютная сбалансированность целевых значений не приводит к улучшению конечных результатов. Установлено, что оптимальным граничным значением для алгоритма логистической регрессии является 0,49 вместо используемого по умолчанию 0,5. Определено, что увеличение показателей в наборе входных данных и их преобразование с помощью средних и медианных значений способствует улучшению точности прогнозирования. Вместе с тем увеличение количества входных показателей с использованием полиномов не привело к однозначному улучшению показателей модели, но существенно увеличило длительность обучения задействованного алгоритма. Применение метода главных компонент позволило получить максимальное значение коэффициента эффективности A и является целесообразным при решении задачи классификации займов.

Вклад авторов. *В. И. Бегунков* разработал принципы обработки данных и программной модели решения задачи классификации займов, провел эксперименты с интерпретированием результатов. *М. Я. Ковалев* сформулировал задачу исследования и выполнил научное редактирование статьи.

References

1. Gerhard F., Harlalka A., Suvanam R. The coming opportunity in consumer lending. *McKinsey Quarterly*, 2021. Available at: https://www.mckinsey.com/business-functions/risk-and-resilience/our-insights/the-coming-opportunity-in-consumer-lending (accessed 01.05.2021).

2. Hand D. J., Henley W. E. Statistical classification methods in consumer credit scoring: a review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 1997, vol. 160, no. 3, pp. 523–541.

3. Baesens B., Van Gestel T., Viaene S., Stepanova S., Suykens J., Vanthienen J. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, 2003, vol. 54, no. 6, pp. 627–635.

4. Lessmann S., Baesens B., Seow H.-V., Thomas L. C. Benchmarking state-of-the-art classification algorithms for credit scoring: an update of research. *European Journal of Operational Research*, 2015, vol. 247, no. 1, pp. 124–136.

5. Shalev-Shwartz S., Ben-David S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014, pp. 125, 126–127.

6. Geron A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd edition. O'Reilly Media, 2019, pp. 144–149.

7. Murphy K. P. *Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning Series)*. The MIT Press, 2012, pp. 225–227, 387–407.

8. Harrington P. *Machine Learning in Action*, 1st edition. Manning Publication Co, 2012, pp. 86–91, 148, 269–279.

9. Fawcett T. An introduction to ROC analysis. *Pattern Recognition Letters*, 2006, vol. 27, no. 8, pp. 861–874.

10. Metz C. E. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 1978, vol. 8, no. 4, pp. 283–298.

11. Kelleher J. D., Namee B. M., D'Arcy A. *Fundamentals of Machine Learning for Predictive Data Analytics*, 1st edition. The MIT Press, 2015, pp. 142–143, 539.

Информация об авторах

Бегунков Владимир Иванович, магистр технических наук.
E-mail: vbegunkov@gmail.com

Ковалев Михаил Яковлевич, член-корреспондент НАН Беларуси, доктор физико-математических наук, профессор, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: kovalyov_my@newman.bas-net.by
https://orcid.org/0000-0003-0832-0829

Information about the authors

Uladzimir I. Behunkou, M. Sc. (Eng.).
E-mail: vbegunkov@gmail.com

Mikhail Y. Kovalyov, Corresponding Member of the National Academy of Sciences of Belarus, D. Sc. (Eng.), Prof., The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: kovalyov_my@newman.bas-net.by
https://orcid.org/0000-0003-0832-0829