DOI 10.37539/2949-1991.2026.40.5.040

Подковыров Алексей Игоревич, Студент,
Сахалинский государственный университет

Научный руководитель:
Осипов Геннадий Сергеевич,
д.т.н., профессор кафедры информатики,
Сахалинский государственный университет

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ КЛАССИФИКАТОРОВ В СРЕДЕ WOLFRAM MATHEMATICA

Аннотация. В статье проводится сравнительный анализ девяти методов машинного обучения при решении задачи классификации набора данных «Ирисы Фишера». Исследование выполнено в среде Wolfram Mathematica. Проведена оценка точности, длительности обучения, объёма занимаемой оперативной памяти и времени классификации одного примера каждого метода. На основе нормализации метрик получена интегральная оценка эффективности каждого метода.

Ключевые слова: Классификация, машинное обучение, логистическая регрессия, нейронная сеть, дерево решений, градиентный бустинг, метод ближайших соседей, случайный лес, метод опорных векторов, модель Маркова, наивный Байес.

Постановка задачи классификации

Задачи классификации представляют собой одну из наиболее распространенных категорий задач машинного обучения на размеченных данных, где метод должен отнести объекты к одной из заранее определенных категорий на основе их признаков [1].

Классическим примером такой задачи является набор данных «Ирисы Фишера» [2]. Он содержит 150 образцов ирисов трех видов: Iris-setosa, Iris-versicolor, Iris-virginica, каждый из которых описывается четырьмя числовыми признаками: длиной и шириной чашелистика, длиной и шириной лепестка.

В рамках поставленной задачи необходимо провести сравнительный анализ девяти методов классификации на наборе данных «Ирисы Фишера» в среде Wolfram Mathematica, оценив их точность, объём памяти, занимаемый методом, время обучения и среднее время классификации одного примера, а также получить интегральную оценку на основе нормализации метрик для выбора наилучшего метода.

Характеристика исследуемых методов классификации

• Логистическая регрессия – линейный вероятностный классификатор, оценивающий вероятность принадлежности объекта к классу с помощью логистической функции. К её достоинствам относятся простота, высокая скорость работы, хорошая интерпретируемость (коэффициенты показывают влияние каждого признака) и возможность получения вероятностной оценки. Главный недостаток – предположение о линейной разделимости классов, что делает метод недостаточно гибким для сложных нелинейных данных.

• Нейронная сеть – многослойная структура искусственных нейронов, способная выявлять сложные нелинейные закономерности за счет иерархического извлечения признаков. Среди преимуществ – исключительная гибкость, способность моделировать зависимости любой сложности и достижение наилучших результатов во многих областях. Однако метод требует больших объемов данных и вычислительных ресурсов, сложен в настройке, а его работа практически не поддается интерпретации («черный ящик»).

• Дерево решений – логический метод, последовательно разбивающий данные по правилам «если-то» до достижения чистоты классов в листьях. Достоинства: высокая интерпретируемость, отсутствие необходимости в масштабировании признаков, работа как с числовыми, так и с категориальными данными, автоматический отбор значимых признаков. Недостатки – склонность к переобучению и неустойчивость: малые изменения в данных могут привести к построению совершенно другого дерева.

• Градиентный бустинг – ансамблевый метод, последовательно строящий деревья, каждое из которых исправляет ошибки предыдущих. Плюсы метода – очень высокая точность на табличных данных и устойчивость к переобучению при правильной настройке параметров. Минусы: высокая ресурсоемкость, длительное обучение, чувствительность к выбросам и сложность интерпретации по сравнению с одиночным деревом.

• Метод ближайших соседей – метрический метод, относящий объект к наиболее частому классу среди k ближайших соседей в пространстве признаков. Он прост в реализации и понимании, не требует отдельного обучения, хорошо работает с нелинейными границами классов. В то же время метод чувствителен к масштабу признаков (требует нормализации), медленно работает на этапе предсказания и неустойчив к шуму и выбросам.

• Случайный лес – ансамблевый метод, строящий множество деревьев на случайных подвыборках данных и подмножествах признаков с последующим усреднением результатов. К сильным сторонам относятся высокая точность, устойчивость к переобучению, возможность оценки важности признаков и работа с разными типами данных. Недостатки – меньшая интерпретируемость по сравнению с одиночным деревом, повышенные требования к памяти и времени обучения.

• Метод опорных векторов – строит разделяющую гиперплоскость с максимальным зазором между классами; для нелинейных данных используются ядра. Метод обеспечивает высокую точность, эффективен в многомерных пространствах и обладает хорошей обобщающей способностью. Однако он чувствителен к выбору ядра и гиперпараметров, сложен в интерпретации и относительно медленно работает на больших данных.

• Модель Маркова – вероятностный метод, учитывающий последовательности состояний и переходы между ними. Её преимущество –способность моделировать временные или логические зависимости между объектами. Среди недостатков – сложность настройки и менее широкое распространение для стандартных задач классификации по сравнению с другими методами.

• Наивный Байес – вероятностный метод, основанный на теореме Байеса с «наивным» предположением о независимости всех признаков. Он очень быстр в обучении и предсказании, хорошо масштабируется на большие и многомерные данные, эффективен при малом объеме выборки. Главный недостаток – предположение о независимости признаков часто не выполняется на реальных данных, что может существенно ограничивать точность классификации.

Результаты моделирования и сравнительный анализ

Исследование выполнено в среде Wolfram Mathematica – мощной системе для инженерных и научных вычислений, обладающей встроенными средствами машинного обучения [3].

Обучение классификаторов выполнялось с помощью встроенной функции Classify с указанием соответствующих методов [4]. Для каждого метода фиксировались: точность, количество ошибок классификации, объём памяти, занимаемый методом, время обучения и среднее время классификации одного примера.

В качестве примера разобран процесс обучения и оценки классификатора методом логистической регрессии. На рисунке 1 приведен вызов функции Classify для построения классификатора.

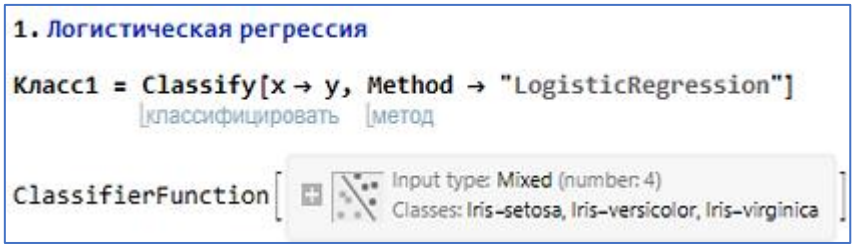

Рисунок 1. Построение классификатора

На рисунке 2 приведена подробная информация о параметрах обучения: точность, среднее время классификации одного примера, объём памяти, занимаемый методом, и время, затраченное на обучение, а также график изменения точности классификации в процессе обучения.

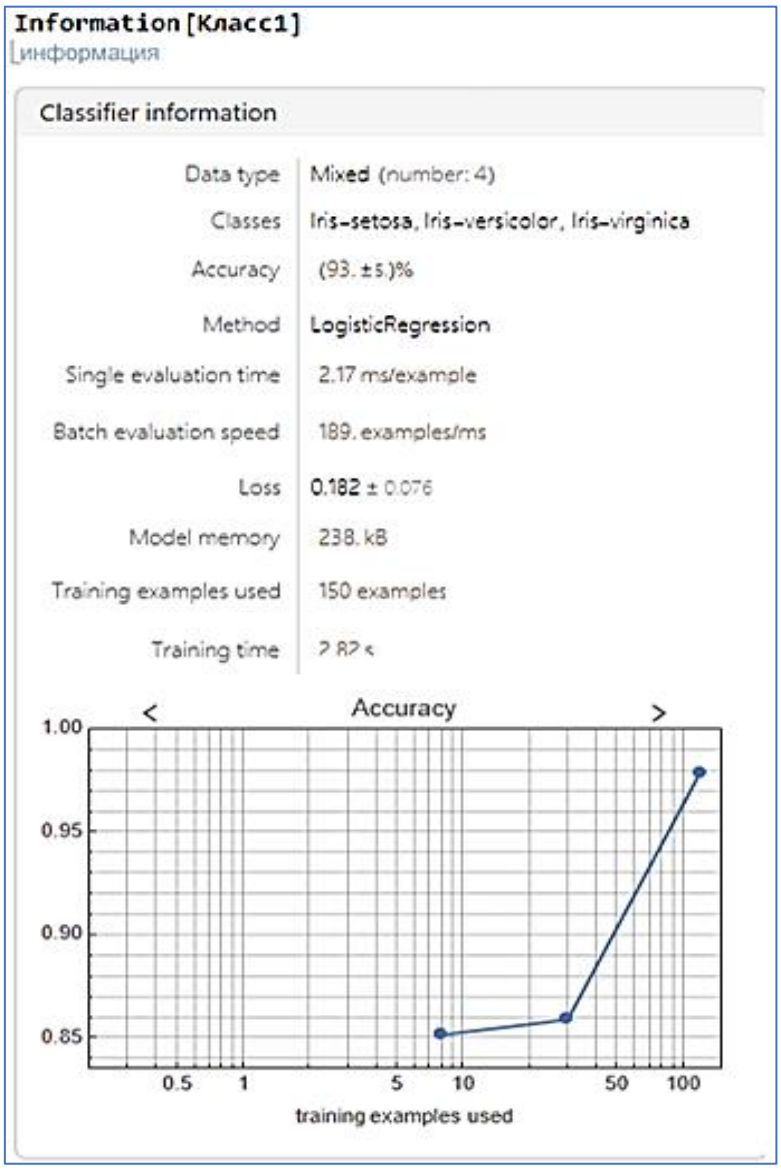

Рисунок 2. Параметры обучения и график изменения точности классификатора

На рисунке 3 показаны результаты тестирования обученного классификатора на рассматриваемом наборе данных с помощью функции ClassifierMeasurements, включая матрицу ошибок и доверительный интервал для точности классификации [5].

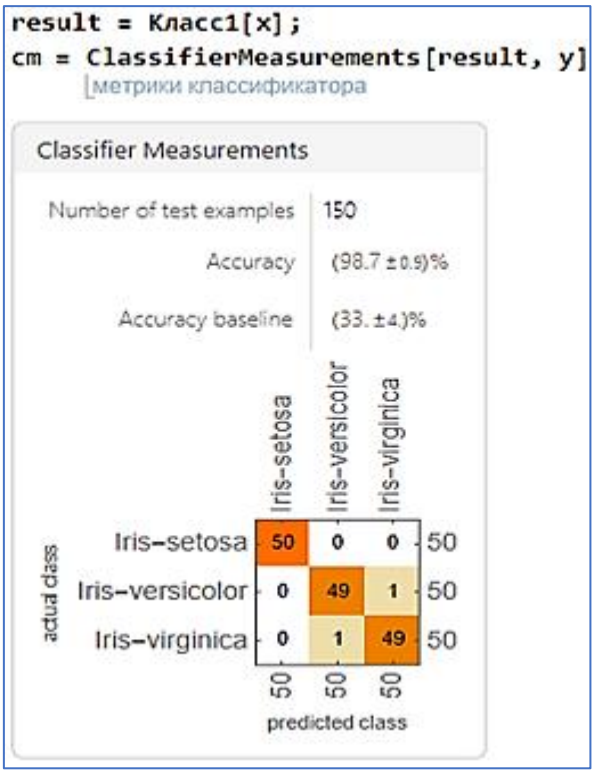

Рисунок 3. Информация об ошибках и точности классификации

Как видно из рисунков 1-3, среда Wolfram Mathematica позволяет строить классификаторы, основанные на различных методах, а также предоставляет подробную информацию об обучении: точность классификации, время, затраченное на обучение, объём занимаемой оперативной памяти и время классификации одного примера. Помимо этого, есть возможность протестировать классификатор на наборе данных и получить информацию об ошибках и точности – матрицу ошибок и доверительный интервал для точности классификации.

Аналогичные данные были получены для всех девяти рассматриваемых методов, а собранные характеристики сведены в таблицу 1. Это позволяет наглядно сравнить методы между собой и выявить сильные и слабые стороны каждого.

Таблица 1.

Сравнительная таблица методов машинного обучения

Метод	**Точность**	**Всего ошибок**	**Объём занимаемой памяти**	**Время обучения**	**Время оценки 1 примера**
Логистическая регрессия	**98,7%**	**2**	238 kB	2,82 с	2,17 мс
Нейронная сеть	98%	3	297 kB	23,5 с	3,03 мс
Дерево решений	97,3%	4	**146 kB**	**384 мс**	2,17 мс

Градиентный бустинг	97,3%	4	603 kB	2,97 с	20 мс
Ближайший сосед	96,7%	5	153 kB	409 мс	**2,07 мс**
Случайный лес	96,7%	5	241 kB	548 мс	5,34 мс
Опорные вектора	96,7%	5	241 kB	3,08 с	4,38 мс
Модель Маркова	95,3%	7	190 kB	514 мс	5,09 мс
Наивный Байес	93,3%	10	168 kB	388 мс	3,49 мс

Как видно из таблицы 1, максимальная точность классификации (98,7%) достигается при использовании метода логистической регрессии, допустившей всего 2 ошибки. Дерево решений оказалось наименее требовательным по объёму оперативной памяти (146 kB) и самым быстрым в обучении (384 мс) методом, а метод ближайшего соседа – самым быстрым при оценке одного примера (2,07 мс).

Для наглядного восприятия данных построены гистограммы по каждому из показателей. На каждой гистограмме представлены соответственно значения точности, объёма занимаемой оперативной памяти, времени обучения и среднего времени оценки одного примера для каждого из девяти рассматриваемых методов, что позволяет наглядно оценить разброс между лучшим и худшим результатом.

На рисунке 4 представлено сравнение точности классификации рассмотренных методов.

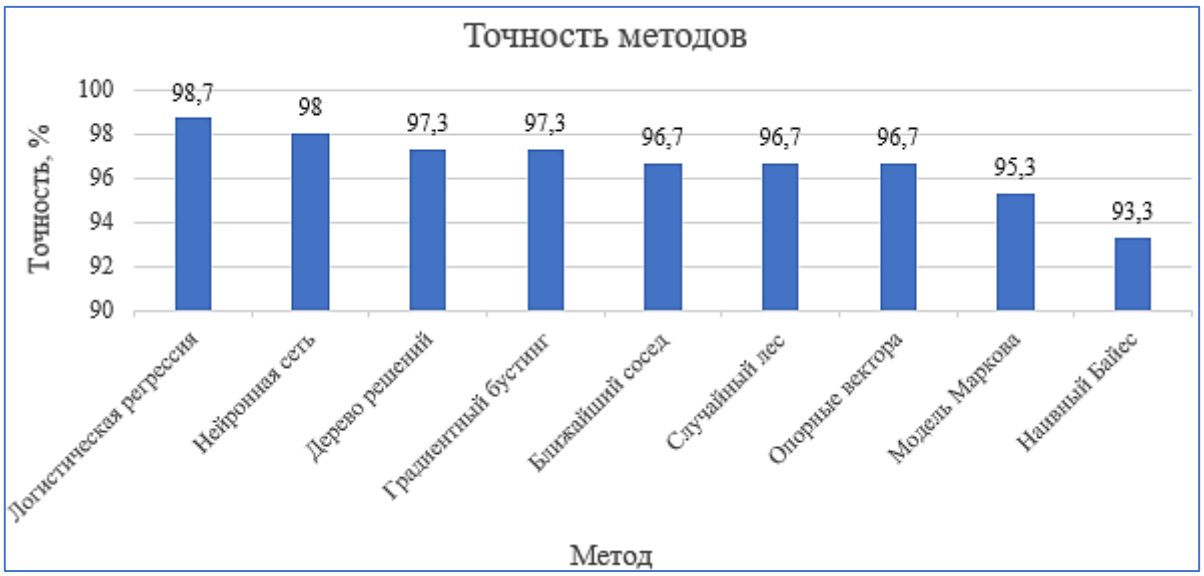

Рисунок 4. Гистограмма точности методов классификации

Максимальная точность классификации достигнута при использовании метода логистической регрессии (98,7%), минимальная – при использовании метода наивного Байеса (93,3%).

На рисунке 5 показаны объёмы памяти, требуемые классификатору для обучения каждым методом.

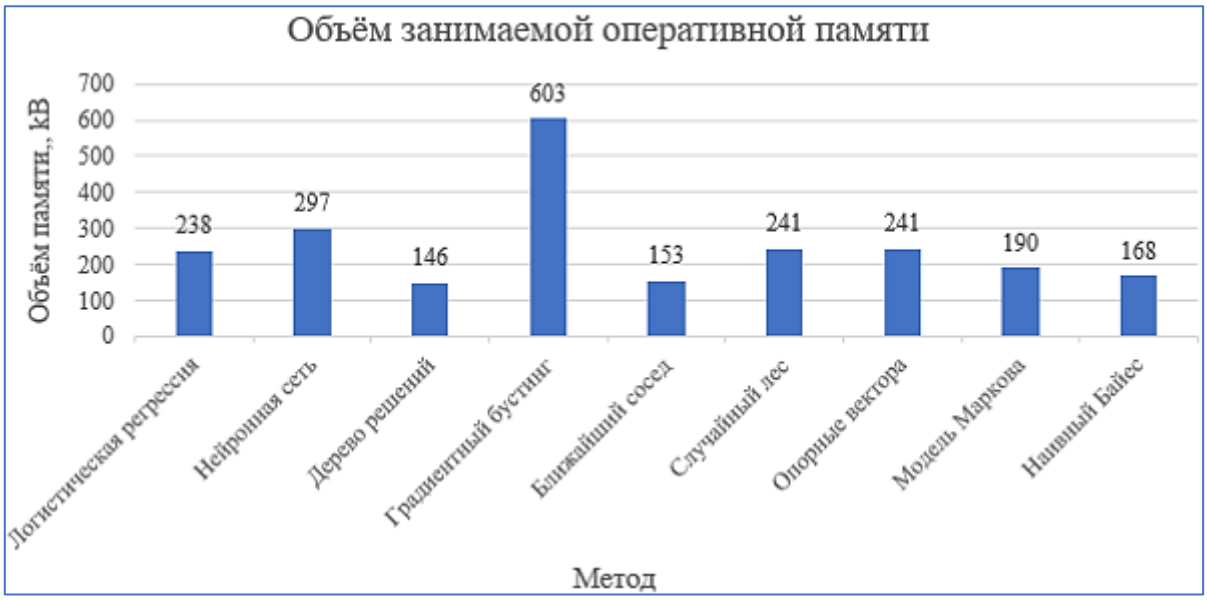

Рисунок 5. Гистограмма объёма памяти, занимаемого методами классификации

Наименьший объём памяти требуется классификатору при использовании метода дерева решений (146 kB), наибольший – при использовании метода градиентного бустинга (603 kB).

На рисунке 6 приведено сравнение методов по времени обучения.

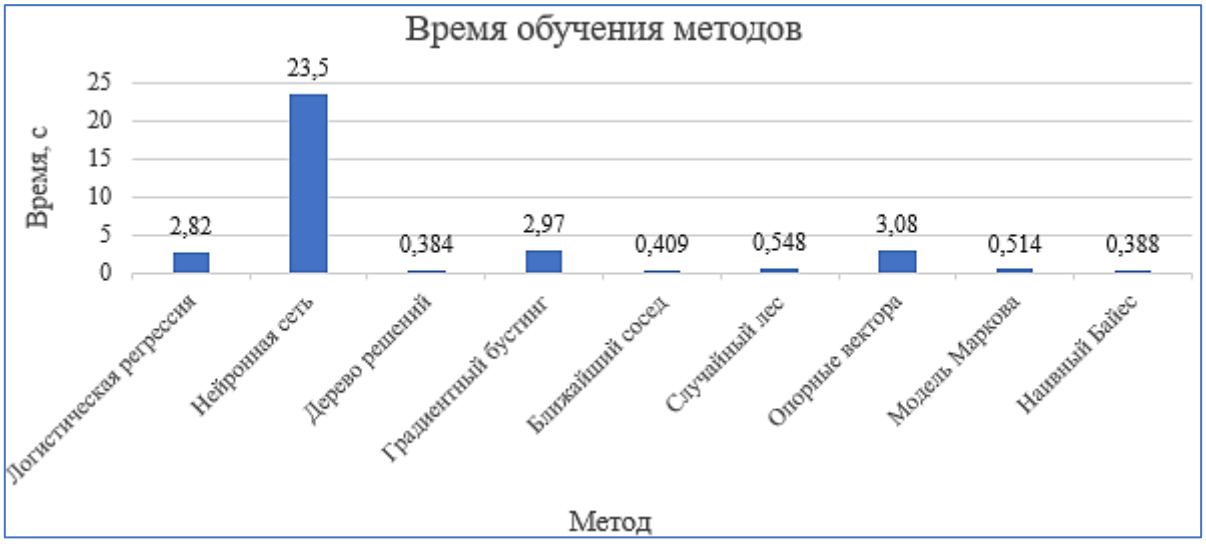

Рисунок 6. Гистограмма времени обучения методов классификации

Наименьшее время обучения классификатора достигнуто при использовании метода дерева решений (0,384 с), наибольшее – при использовании метода нейронной сети (23,5 с).

На рисунке 7 показано время оценки одного примера классификатором.

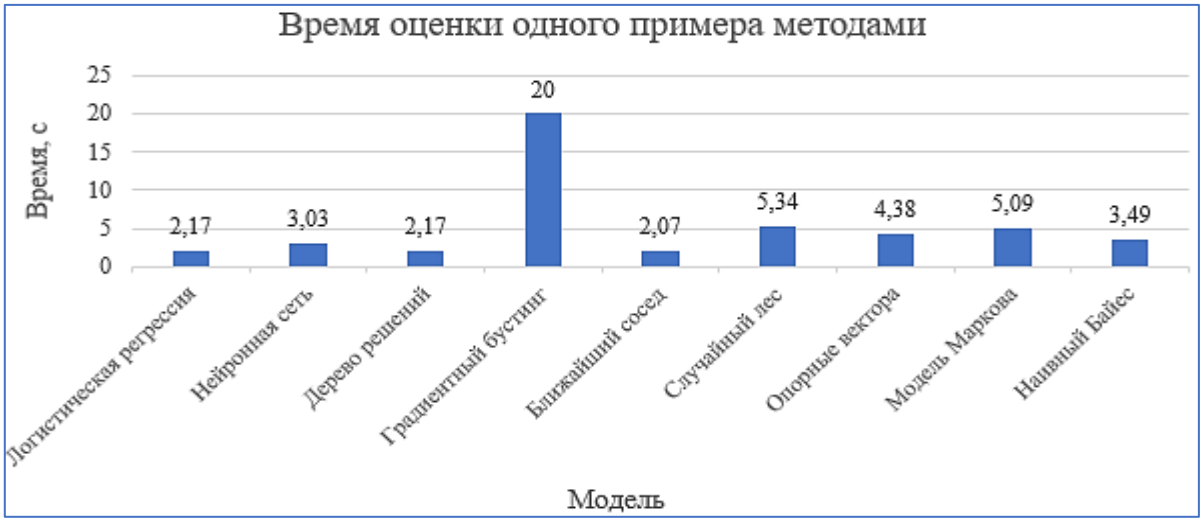

Рисунок 7. Гистограмма времени оценки одного примера классификации

Меньше всего времени для обработки одного примера требуется классификатору при использовании метода ближайшего соседа (2,07 мс), больше всего – при использовании метода градиентного бустинга (20 мс).

Интегральная оценка методов

Таблица 1 позволяет сопоставить методы по каждому показателю в отдельности, однако не дает возможности сделать однозначный вывод об их общей эффективности. Например, высокая точность классификации может достигаться за счет значительного увеличения времени обучения или объема потребляемой памяти. И наоборот, высокая скорость работы нередко сопровождается снижением точности.

Для получения единой интегральной оценки, объединяющей все показатели, была применена нормализация по формуле: $x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$, где x – значение показателя для данного метода, а x_{min} и x_{max} – наименьшее и наибольшее значения среди рассмотренных методов.

Нормализация позволила привести все показатели к единому масштабу [0; 1], делая их сопоставимыми. Для точности, которую необходимо максимизировать, значение используется напрямую. Для параметров, которые, наоборот, необходимо минимизировать (время обучения, память, время оценки), берется величина $1 - x_{norm}$.

Результаты нормализации для каждого метода представлены в таблице 2. Значения, стремящиеся к единице по всем параметрам, свидетельствуют о высоком результате по соответствующей метрике (для точности) или о низких затратах (для объёма занимаемой памяти, времени обучения и времени оценки одного примера).

Это позволяет определить наиболее сбалансированные методы классификации.

Таблица 2.

Нормализованные значения показателей эффективности

Метод	**Точность**	**Объём занимаемой памяти**	**Время обучения**	**Время оценки 1 примера**
Логистическая регрессия	**1**	0,799	0,895	0,994
Нейронная сеть	0,87	0,67	0	0,946
Дерево решений	0,74	**1**	**1**	0,994
Градиентный бустинг	0,74	0	0,888	0

Ближайший сосед	0,63	0,985	0,999	**1**
Случайный лес	0,63	0,792	0,993	0,818
Опорные вектора	0,63	0,792	0,883	0,871
Модель Маркова	0,37	0,904	0,994	0,832
Наивный Байес	0	0,952	**1**	0,921

Как видно из таблицы 2, каждый метод имеет свои сильные и слабые стороны. Метод логистической регрессии лидирует по точности, метод дерева решений – по компактности и скорости обучения, а метод ближайшего соседа – по скорости оценки.

Далее представлено визуальное сравнение нормализованных значений каждого из методов в виде гистограммы (рис. 6).

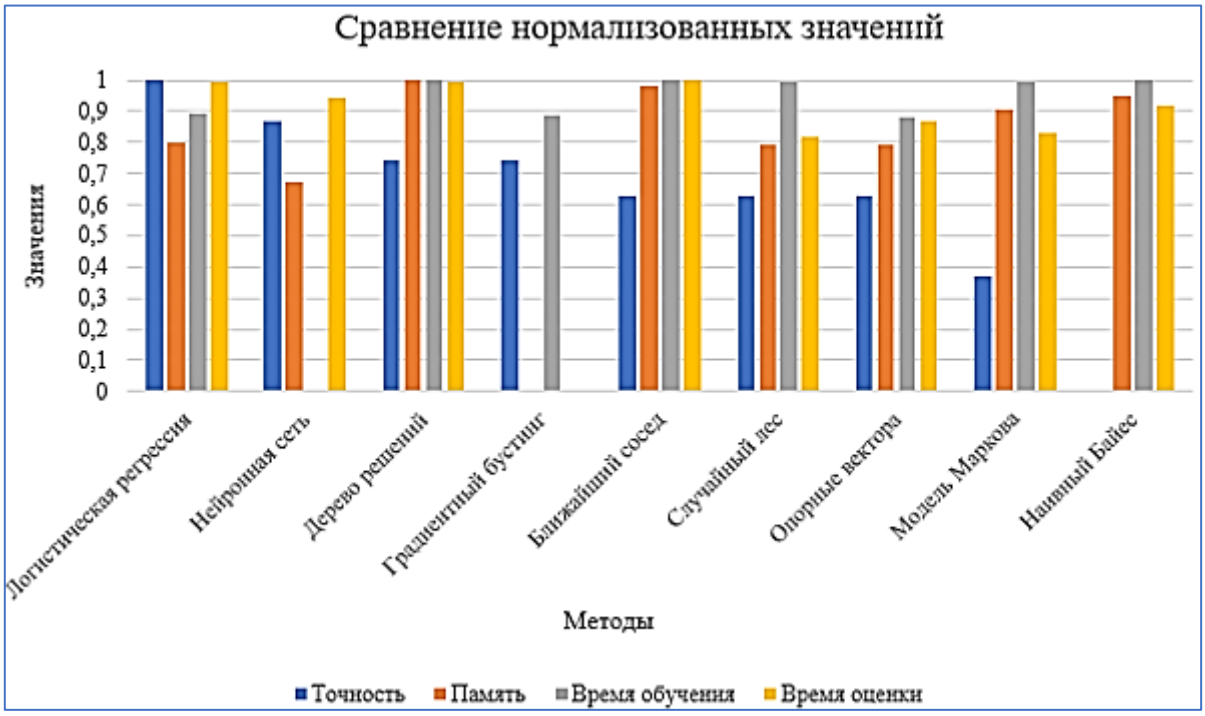

Рисунок 8. Гистограмма нормализованных показателей эффективности

На гистограмме видно, что дерево решений и логистическая регрессия являются наиболее сбалансированными, тогда как градиентный бустинг и нейронная сеть имеют явные слабые места.

Для выявления наиболее сбалансированных методов были просуммированы нормализованные значения для каждого метода. Результаты представлены в таблице 3.

Таблица 3.

Интегральная оценка методов классификации

Место	**Метод**	**Сумма значений**
1	Дерево решений	**3,74**
2	Логистическая регрессия	3,69
3	Ближайший сосед	3,61
4	Случайный лес	3,23
5	Опорные вектора	3,18
6	Модель Маркова	3,10

7	Наивный Байес	2,87
8	Нейронная сеть	2,49
9	Градиентный бустинг	1,63

Как видно из таблицы, наиболее сбалансированными методами оказались дерево решений, логистическая регрессия и метод ближайшего соседа. Эти методы демонстрируют высокие значения по всем четырем показателям, уступая друг другу лишь по отдельным метрикам.

Наименьший результат показал градиентный бустинг, что обусловлено высоким временем оценки одного примера и большим объемом занимаемой памяти. Нейронная сеть также не показала высоких результатов из-за чрезмерно долгого времени обучения.

Вывод

Для классификации ирисов Фишера наиболее сбалансированным методом по совокупности характеристик оказалось дерево решений. Оно немного уступает логистической регрессии по точности, но значительно превосходит её по скорости обучения и объему потребляемой памяти.

Логистическая регрессия показала наивысшую точность, что объясняется линейной разделимостью классов в данном наборе.

Метод ближайшего соседа продемонстрировал максимальную скорость классификации новых объектов, что делает его предпочтительным в системах реального времени.

Наименее подходящими для данной задачи оказались градиентный бустинг и нейронная сеть. Эти методы рассчитаны на выявление нелинейных зависимостей в больших объемах данных. При сопоставимой с простыми методами точности их использование неоправданно.

Таким образом, для поставленной задачи оптимальным выбором является дерево решений. Однако в зависимости от требований могут быть использованы и другие методы, такие как логистическая регрессия или метод ближайшего соседа.

Список литературы:

1. Кравцова Н. Е., Преображенский А. П. О решении задач классификации в методах машинного обучения // Вестник Воронежского института высоких технологий. 2018 [Электронный ресурс]. – Режим доступа: https://vestnikvivt.ru/ru/journal/pdf?id=903 (дата обращения: 31.05.2026).

2. Iris Data Set // UCI Machine Learning Repository [Электронный ресурс]. – Режим доступа: https://web.archive.org/web/20151211073938/

http://archive.ics.uci.edu/ml/datasets/Iris (дата обращения: 6.05.2026).

3. Wolfram Research, Inc. Machine Learning in Wolfram Mathematica [Электронный ресурс]. – Режим доступа: https://reference.wolfram.com/language/guide/MachineLearning.html (дата обращения: 31.05.2026).

4. Официальная документация Wolfram Mathematica. Функция Classify [Электронный ресурс]. – Режим доступа: https://reference.wolfram.com/language/ref/Classify.html (дата обращения: 31.05.2026).

5. Официальная документация Wolfram Mathematica. Функция ClassifierMeasurements [Электронный ресурс]. – Режим доступа: https://reference.wolfram.com/language/ref/ClassifierMeasurements.html (дата обращения: 31.05.2026).