Оценка качества

моделей машинного обучения: выбор, интерпретация и применение метрик

Алексей Михнин

Алексей Михнин

Оценка качества моделей машинного обучения: выбор, интерпретация и применение метрик

http://www.litres.ru/pages/biblio_book/?art=69618313
SelfPub; 2023

Аннотация

В данной книге рассматриваются метрики качества моделей машинного обучения, обеспечивая понимание их выбора, интерпретации и применения. Описываются различные метрики, их особенности и применение в задачах машинного обучения. Книга содержит практические примеры использования метрик для наглядности. Она будет полезна специалистам в области машинного обучения, бизнес-аналитикам и новичкам, желающим освоить оценку качества моделей и принимать обоснованные решения на основе анализа результатов моделирования.

Содержание

Алексей Михнин
Оценка качества моделей машинного обучения: выбор, интерпретация и применение метрик

Введение

Машинное обучение становится все более важным инструментом в разнообразных отраслях, от медицины и финансов до транспорта и производства. В связи с растущей популярностью машинного обучения, все больше внимания уделяется оценке качества моделей, основанных на этом подходе. Основным инструментом для оценки качества моделей являются метрики, которые позволяют оценить эффективность работы модели на определенных данных и выбрать наилучшие параметры для повышения производительности.

Выбор и интерпретация метрик может быть сложным процессом, особенно для тех, кто только начинает изучать машинное обучение. В данной книге мы стремимся объяснить сложные аспекты на доступном языке, чтобы помочь вам лучше понять, как выбирать, интерпретировать и применять метрики качества моделей машинного обучения.

В этой книге вы узнаете о разных метриках качества модели, их особенностях, применении в разных задачах машинного обучения и их интерпретации. Мы также предоставим практические примеры использования метрик для лучшего понимания их работы в реальных условиях.

Мы надеемся, что эта книга станет полезным ресурсом для тех, кто хочет углубить свои знания о выборе и применении метрик для оценки качества моделей машинного обучения. Книга будет полезна как специалистам в области машинного обучения, так и бизнес-аналитикам, применяющим модели машинного обучения для решения различных задач. Понимание метрик качества модели поможет им принимать более обоснованные решения, основанные на анализе результатов моделирования, и лучше понимать влияние изменений параметров модели на ее производительность. Кроме того, книга может быть полезна начинающим специалистам в области машинного обучения, которые только начинают осваивать теорию и практику оценки качества модели.

Термины и определения

Модель машинного обучения – алгоритм, который использует статистические методы для обучения на данных и прогнозирования результатов на новых данных.

Метрика качества модели – инструмент для оценки производительности модели машинного обучения. Метрики качества модели позволяют измерить точность и качество работы модели на данных.

Задача классификации – задача машинного обучения, при которой модель должна отнести объекты к определенным классам на основе характеристик объектов.

Задача регрессии – задача машинного обучения, при которой модель должна предсказать численный выход на основе входных данных.

Задача кластеризации – задача машинного обучения, при которой модель должна группировать объекты в кластеры на основе сходства между объектами.

Задача обнаружения аномалий – задача машинного обучения, при которой модель должна определять объекты, которые отличаются от нормального поведения.

Задача обнаружения объектов – задача машинного обучения, при которой модель должна обнаруживать объекты на изображениях и видео.

Accuracy (Точность) – метрика качества модели для задач классификации, которая определяет долю правильных ответов, которые модель дает для всех классов.

Precision (Точность) – метрика качества модели для задач классификации, которая определяет долю истинно положительных ответов относительно всех положительных ответов.

Recall (Полнота) – метрика качества модели для задач классификации, которая определяет долю истинно положительных ответов относительно всех истинно положительных и ложно отрицательных ответов.

F1-score (F-мера) – метрика качества модели для задач классификации, которая является гармоническим средним между точностью и полнотой.

ROC AUC – метрика качества модели для задач классификации, которая измеряет способность модели различать между классами.

Mean Squared Error (MSE) – метрика качества модели для задач регрессии, которая измеряет среднеквадратическую ошибку между прогнозируемым и фактическими значениями.

Root Mean Squared Error (RMSE) – метрика качества модели для задач регрессии, которая является корнем из среднеквадратической ошибки.

Mean Absolute Error (MAE) – метрика качества модели для задач регрессии, которая измеряет среднюю абсолютную ошибку между прогнозируемым и фактическим значением.

R-squared (коэффициент детерминации) – метрика качества модели для задач регрессии, которая измеряет долю дисперсии, которая может быть объяснена моделью.

Silhouette coefficient (коэффициент силуэта) – метрика качества модели для задач кластеризации, которая измеряет степень разделения кластеров.

Calinski-Harabasz index (индекс Калински-Харабаса) – метрика качества модели для задач кластеризации, которая измеряет степень разделения кластеров и межкластерное расстояние.

Davies-Bouldin index (индекс Дэвиса-Болдина) – метрика качества модели для задач кластеризации, которая измеряет суммарное сходство кластеров и их компактность.

AUROC (площадь под кривой операционной характеристики получателя) – метрика качества модели для задач обнаружения аномалий и классификации, которая измеряет способность модели различать между классами и находить аномалии.

Mean Average Precision (mAP) – метрика качества модели для задач обнаружения объектов, которая измеряет среднюю точность распознавания объектов на изображениях.

Intersection over Union (IoU) – метрика качества модели для задач обнаружения объектов, которая измеряет степень перекрытия между прогнозируемыми и фактическими объектами на изображениях.

Overfitting (переобучение) – явление, когда модель слишком хорошо запоминает данные обучения и не может обобщать на новые данные.

Underfitting (недообучение) – явление, когда модель не может достичь достаточной точности на данных обучения и не может обобщать на новые данные.

Cross-validation (кросс-валидация) – метод оценки производительности модели путем разделения данных на несколько частей и обучения модели на одной части и тестирования на другой. Этот процесс повторяется несколько раз с разными разбиениями данных, чтобы усреднить оценку производительности модели.

Hyperparameters (гиперпараметры) – параметры модели машинного обучения, которые настраиваются перед обучением и влияют на ее производительность и способность обобщать на новые данные.

Bias (смещение) – ошибка модели, которая происходит из-за ее недостаточной сложности и невозможности захватить сложные зависимости в данных.

Variance (дисперсия) – ошибка модели, которая происходит из-за ее слишком большой сложности и способности переобучаться на данных обучения.

Regularization (регуляризация) – метод, используемый для уменьшения переобучения модели путем добавления штрафа за сложность модели.

Feature engineering (инженерия признаков) – процесс преобразования и выбора признаков для улучшения производительности модели и увеличения ее способности обобщать на новые данные.

Введение в метрики качества модели

Что такое метрики качества модели?

Метрики качества модели – это инструменты для оценки производительности модели машинного обучения. Они позволяют определить, насколько хорошо модель работает на конкретных данных и насколько она точна в решении задачи, для которой она была обучена.

В данной книге представлен далеко не полный список метрик, и существуют и другие метрики, которые могут быть использованы для оценки качества моделей. Выбор подходящей метрики зависит от типа задачи, особенностей данных и целей проекта. Метрики представленные в данной книге наиболее распространенные при анализе качества типовых моделей машинного обучения.

Зачем нужны метрики качества модели?

Метрики качества модели необходимы для того, чтобы выбирать лучшие параметры модели и оптимизировать ее производительность. Они позволяют сравнить производительность нескольких моделей и выбрать наилучшую из них. Также метрики качества модели могут помочь в идентификации проблем в данных или модели и определении, где нужно внести изменения, чтобы улучшить ее производительность.

Как выбрать подходящую метрику качества модели?

Выбор подходящей метрики качества модели зависит от типа задачи, для которой модель была обучена. Например, метрики качества модели для задачи классификации будут отличаться от метрик качества модели для задачи регрессии. Также необходимо учитывать особенности данных, на которых модель будет применяться, и целей проекта.

Для выбора подходящей метрики качества модели необходимо задаться несколькими вопросами:

Какую задачу решает модель? (классификация, регрессия, кластеризация, обнаружение аномалий и т.д.)

Какие особенности данных нужно учитывать? (размер датасета, баланс классов, наличие выбросов и т.д.)

Какие цели нужно достигнуть? (максимизация точности, минимизация ошибок, оптимизация скорости и т.д.)

Выбор подходящей метрики качества модели может быть сложной задачей, поэтому необходимо тщательно изучать свойства и особенности каждой метрики и выбирать ту, которая наилучшим образом соответствует задаче и целям проекта.

Например, для задачи классификации можно использовать метрики качества, такие как точность (accuracy), точность (precision), полнота (recall), F-мера (F1-score) и ROC

AUC. Точность (accuracy) определяет долю правильных ответов, которые модель дает для всех классов. Точность (precision) определяет долю истинно положительных ответов относительно всех положительных ответов, а полнота (recall) определяет долю истинно положительных ответов относительно всех положительных результатов. F-мера (F1-score) является гармоническим средним между точностью и полнотой, а ROC AUC измеряет способность модели различать между классами.

Для задач регрессии могут использоваться метрики качества, такие как среднеквадратическая ошибка (MSE), корень среднеквадратической ошибки (RMSE), средняя абсолютная ошибка (MAE), коэффициент детерминации (R-squared) и другие.

Для задач кластеризации могут использоваться метрики качества, такие как коэффициент силуэта (silhouette coefficient), индекс Калински-Харабаса (Calinski-Harabasz index), индекс Дэвиса-Болдина (Davies-Bouldin index) и другие.

Для задач обнаружения аномалий можно использовать метрики, такие как показатель точности (precision), показатель полноты (recall), F-меру (F1-score), площадь под кривой операционной характеристики получателя (AUROC) и другие.

Для задач обнаружения объектов метрики качества могут включать среднюю точность (mAP), коэффициент пересечения (IoU), точность (precision), полноту (recall) и другие.

В данной книге мы рассмотрим более подробно каждую метрику и ее применение в различных задачах машинного обучения. Мы также рассмотрим способы интерпретации метрик и примеры их использования на практике. Мы надеемся, что это поможет вам лучше понимать, как выбрать подходящую метрику качества модели и как правильно интерпретировать ее результаты.

Метрики качества модели для задач классификации

Метрики качества модели для задач классификации, такие как Accuracy, Precision, Recall, F1-score, ROC AUC, Log Loss и Confusion Matrix (Матрица ошибок), применяются в различных жизненных ситуациях, где необходимо оценить производительность алгоритмов классификации. Вот несколько примеров:

Медицинская диагностика: В медицине алгоритмы классификации могут использоваться для диагностики заболеваний, определения стадий рака, предсказания риска развития определенных заболеваний или идентификации патогенов. Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Confusion Matrix, могут быть использованы для оценки эффективности этих алгоритмов и улучшения точности диагностики.

Фильтрация спама: В системах фильтрации спама алгоритмы классификации используются для определения спам-писем и разделения их от легитимных сообщений. Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Log Loss, могут быть использованы для оценки производительности этих систем и определения того, насколько хорошо они фильтруют спам.

Определение мошенничества: В банковской и финансовой сфере алгоритмы классификации используются для обнаружения подозрительных транзакций, мошенничества с кредитными картами или неправомерного использования. Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Confusion Matrix, могут быть использованы для оценки производительности этих систем и определения областей для дальнейшего улучшения.

Рекомендательные системы: В рекомендательных системах, таких как интернет-магазины, потоковые сервисы и социальные сети, алгоритмы классификации используются для предоставления персонализированных предложений пользователям. Метрики, такие как Accuracy, Precision, Recall, F1-score и ROC AUC, могут помочь оценить эффективность рекомендаций и улучшить качество предложений.

Текстовый анализ и анализ тональности: В области анализа текста алгоритмы классификации используются для определения темы, жанра или эмоциональной окрас ки текста. Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Confusion Matrix, могут быть использованы для оценки эффективности этих алгоритмов и улучшения качества анализа.

Распознавание изображений: В задачах распознавания изображений, таких как определение объектов на фотографиях, классификация видов животных или распознавание лиц, алгоритмы классификации играют ключевую роль.

Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Confusion Matrix, могут быть использованы для оценки производительности этих систем и определения областей для дальнейшего улучшения.

Классификация новостей: В задачах классификации новостей алгоритмы классификации используются для определения темы статьи, источника информации или оценки достоверности новости. Метрики, такие как Accuracy, Precision, Recall, F1-score, ROC AUC и Confusion Matrix, могут быть использованы для оценки эффективности этих алгоритмов и улучшения качества анализа.

Для некоторых метрик качества модели для задач классификации возможно определить хорошие, средние и плохие значения. Однако для других, таких как Log Loss и Confusion Matrix, такие диапазоны не могут быть определены без контекста и масштаба данных. Тем не менее, я представлю таблицу значений для некоторых из метрик:

Метрика	Хорошее значение	Среднее значение	Плохое значение
Accuracy	0.9 - 1.0	0.7 - 0.9	0 - 0.7
Precision	0.9 - 1.0	0.5 - 0.9	0 - 0.5
Recall	0.9 - 1.0	0.5 - 0.9	0 - 0.5
F1-score	0.9 - 1.0	0.5 - 0.9	0 - 0.5
ROC AUC	0.9 - 1.0	0.7 - 0.9	0.5 - 0.7
Log Loss	-	-	-
Confusion Matrix	-	-	-

Для Log Loss и Confusion Matrix не существует фиксированных границ для хороших, средних и плохих значений, потому что они зависят от контекста и масштаба данных. Оценка Log Loss должна проводиться в сравнении с другими моделями на том же наборе данных, а Confusion Matrix должна быть анализирована для определения различных видов ошибок, которые допускает модель.

Важно учитывать, что эти диапазоны являются общими ориентирами и могут варьироваться в зависимости от конкретной области применения и задачи. Например, в критически важных областях, таких как медицинская диагностика, требуется более высокая точность и полнота, чем в менее критических сценариях, таких как рекомендации контента.

Метрика Accuracy (Точность)

Метрика Accuracy (Точность) является одной из наиболее базовых и понятных метрик для оценки качества работы алгоритма классификации. Она измеряет долю правильно классифицированных объектов относительно общего числа объектов в наборе данных.

Метрика Accuracy рассчитывается следующим образом:

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

где:

TP (True Positives) – количество правильно классифицированных положительных объектов;

TN (True Negatives) – количество правильно классифицированных отрицательных объектов;

FP (False Positives) – количество неправильно классифицированных положительных объектов (ложные срабатывания);

FN (False Negatives) – количество неправильно классифицированных отрицательных объектов (пропущенные срабатывания).

Accuracy принимает значения в диапазоне от 0 до 1 (или от 0% до 100%). Чем ближе значение Accuracy к 1 (или 100%), тем лучше работает алгоритм классификации.

Однако, стоит отметить, что метрика Accuracy не всегда является оптимальным выбором для оценки качества классификации, особенно если в наборе данных есть сильный дисбаланс классов. В таких случаях использование других метрик, таких как Precision, Recall или F1-score, может быть более информативным и адекватным.

Пример № 1:

Пусть у нас есть 100 пациентов, из которых 90 здоровы, и 10 больны. Модель правильно классифицирует всех 90 здоровых пациентов и 10 больных пациентов. В этом случае:

TP (True Positives) = 10 (правильно классифицированные больные пациенты)

TN (True Negatives) = 90 (правильно классифицированные здоровые пациенты)

FP (False Positives) = 0 (нет ошибок при классификации здоровых пациентов)

FN (False Negatives) = 0 (нет ошибок при классификации больных пациентов)

Теперь рассчитаем Accuracy:

Accuracy = (TP + TN) / (TP + TN + FP + FN) = (10 + 90) / (10 + 90 + 0 + 0) = 100 / 100 = 1.0 или 100%

В данном примере точность модели составляет 100%.

Пример № 2:

В задаче классификации картинок с котами и собаками у нас есть 1000 картинок, и модель правильно классифицировала 900 из них. Допустим, 500 картинок изображают котов, а другие 500 – собак. Пусть модель правильно классифицировала 450 картинок с котами и 450 картинок с собаками. В этом случае:

TP (True Positives) = 450 (правильно классифицированные картинки с котами)

TN (True Negatives) = 450 (правильно классифицированные картинки с собаками)

FP (False Positives) = 50 (картинки с собаками, классифицированные как коты)

FN (False Negatives) = 50 (картинки с котами, классифицированные как собаки)

Теперь рассчитаем Accuracy:

Accuracy = (TP + TN) / (TP + TN + FP + FN) = (450 + 450) / (450 + 450 + 50 + 50) = 900 / 1000 = 0.9 или 90%

В данном примере точность модели составляет 90%.

Метрика Precision (Точность)

Метрика Precision (Точность) – это одна из метрик качества работы алгоритма классификации, которая показывает, насколько точно модель предсказывает положительный класс. Precision фокусируется на правильно классифицированных положительных объектах и ложных срабатываниях (ложноположительные результаты).

Метрика Precision рассчитывается следующим образом:

Precision = TP / (TP + FP)

где:

TP (True Positives) – количество правильно классифицированных положительных объектов;

FP (False Positives) – количество неправильно классифицированных положительных объектов (ложные срабатывания).

Precision принимает значения в диапазоне от 0 до 1 (или от 0% до 100%). Чем ближе значение Precision к 1 (или 100%), тем точнее модель предсказывает положительный класс.

Важно отметить, что метрика Precision не учитывает ошибки второго рода, то есть пропущенные срабатывания (False Negatives). В некоторых ситуациях, особенно когда пропущенные срабатывания могут иметь серьезные последствия (например, в медицинской диагностике), лучше использовать другие метрики, такие как Recall (полнота) или F1-score, которые учитывают и ошибки первого, и второго рода.

Пример № 1: В задаче определения спам-писем почты, модель может быть настроена таким образом, чтобы допустить только небольшое количество ложных срабатываний. Если модель правильно определила 10 спам-писем из 15, то точность модели для класса спам будет 66.7%.

давайте распишем пошаговое решение для метрики Precision (Точность) на примере № 1:

Определите класс, для которого вы хотите рассчитать точность. В данном примере это класс "спам".

Разделите все примеры на 4 категории: True Positive (TP), False Positive (FP), True Negative (TN) и False Negative (FN). В данном примере это:

TP: модель правильно определила спам-письмо как спам (10 писем).

FP: модель неправильно определила не спам-письмо как спам (5 писем).

TN: модель правильно определила не спам-письмо как не спам (0 писем).

FN: модель неправильно определила спам-письмо как не спам (0 писем).

Рассчитайте точность как отношение TP к общему числу положительных ответов (TP + FP):

Precision = TP / (TP + FP) = 10 / (10 + 5) = 0.667 = 66.7%

Таким образом, в данном примере модель правильно определила 10 из 15 спам-писем, что соответствует точности в 66.7%.

Пример № 2:

В задаче классификации новостей на две категории – политика и спорт – модель классифицировала 200 статей, из которых 150 статей по политике и 50 статей по спорту. Модель правильно определила 120 статей по политике и 40 статей по спорту. Однако, 30 статей по политике модель неправильно классифицировала как спортивные статьи, а 10 спортивных статей – как статьи по политике. Рассчитаем метрику Precision для класса "политика".

Определите класс, для которого вы хотите рассчитать точность. В данном примере это класс "политика".

Разделите все примеры на 4 категории: True Positive (TP), False Positive (FP), True Negative (TN) и False Negative (FN). В данном примере это:

TP: модель правильно определила статью по политике как статью по политике (120 статей).

FP: модель неправильно определила спортивную статью как статью по политике (10 статей).

TN: модель правильно определила спортивную статью как спортивную (40 статей). Значение TN не важно для расчета Precision, поскольку оно не учитывается в формуле.

FN: модель неправильно определила статью по политике как спортивную статью (30 статей). Значение FN также не важно для расчета Precision.

Рассчитайте точность как отношение TP к общему числу положительных ответов (TP + FP): Precision = TP / (TP + FP) = 120 / (120 + 10) = 120 / 130 = 0.923 = 92.3%

Таким образом, в данном примере модель правильно определила 120 из 130 статей, которые были классифицированы как статьи по политике. Точность модели для класса "политика" составляет 92.3%.

Метрика Recall (Полнота)

Метрика Recall (Полнота) – это одна из метрик качества работы алгоритма классификации, которая показывает, какую долю объектов положительного класса модель смогла правильно классифицировать. Recall фокусируется на правильно классифицированных положительных объектах и пропущенных срабатываниях (ложноотрицательные результаты).

Метрика Recall рассчитывается следующим образом:

Recall = TP / (TP + FN)

где:

TP (True Positives) – количество правильно классифицированных положительных объектов;

FN (False Negatives) – количество неправильно классифицированных положительных объектов (пропущенные срабатывания).

Recall принимает значения в диапазоне от 0 до 1 (или от 0% до 100%). Чем ближе значение Recall к 1 (или 100%), тем лучше модель справляется с задачей распознавания положительного класса.

Важно отметить, что метрика Recall не учитывает ложные срабатывания (False Positives). В некоторых случаях, когда ложные срабатывания могут иметь серьезные последствия, например, в задачах определения спам-писем, лучше использовать другие метрики, такие как Precision (точность) или F1-score, которые учитывают и ошибки первого, и второго рода.

Пример № 1:

Пример № 1: В задаче классификации писем на спам и не спам, модель должна максимизировать количество обнаруженных спам-писем. Если модель правильно определила 80 из 100 спам-писем, то полнота модели для класса "спам" будет 80%.

Давайте рассмотрим пошаговое решение для метрики Recall (Полнота) на примере № 1:

Определите класс, для которого вы хотите рассчитать полноту. В данном примере это класс "спам".

Разделите все примеры на 4 категории: True Positive (TP), False Positive (FP), True Negative (TN) и False Negative (FN). В данном примере это:

TP: модель правильно определила спам-письмо как спам (80 писем).

FP: модель неправильно определила не спам-письмо как спам (20 писем).

FN: модель неправильно определила спам-письмо как не спам (20 писем).

Рассчитайте полноту как отношение TP к общему числу положительных примеров (TP + FN):

Recall = TP / (TP + FN) = 80 / (80 + 20) = 0.8 = 80%

Таким образом, в данном примере модель правильно определила 80 из 100 спам-писем, что соответствует полноте в 80%.

Пример № 2: Представьте, что вы работаете аналитиком в интернет-магазине, который хочет улучшить свой алгоритм рекомендаций товаров пользователям. Вы хотите проверить, насколько хорошо работает текущий алгоритм и решаете посчитать метрику полноты для одной из категорий товаров – "электроника".

Для этого вы берете случайную выборку из 200 пользователей, которые просмотрели товары в категории "электроника" на вашем сайте за последний месяц. После того, как вы применили алгоритм рекомендаций, вы получили следующие результаты:

Из 200 пользователей 120 купили хотя бы один рекомендованный товар в категории "электроника" (TP).

Из 200 пользователей 80 не купили ни одного рекомендованного товара в категории "электроника" (FN).

Рассчитайте метрику полноты (recall) для категории "электроника".

Решение:

TP = 120 (пользователи, которые купили хотя бы один рекомендованный товар в категории "электроника") FN = 80 (пользователи, которые не купили ни одного рекомендованного товара в категории "электроника")

Recall = TP / (TP + FN) = 120 / (120 + 80) = 0.6 = 60%

Метрика полноты для категории "электроника" составляет 60%. Это означает, что ваш текущий алгоритм рекомендаций смог правильно найти 60% всех пользователей, которые купили товары в этой категории за последний месяц. Вам следует анализировать результаты и работать над улучшением алгоритма, чтобы повысить метрику полноты и увеличить долю пользователей, которым будут рекомендованы интересные товары в категории "электроника".

Метрика F1-score (F-мера)

Метрика F1-score (F-мера) – это совместная метрика для оценки качества алгоритма классификации, которая учитывает обе метрики Precision (Точность) и Recall (Полнота). F1-score является гармоническим средним между Precision и Recall, что делает эту метрику более сбалансированной, чем каждая из них по отдельности. F1-score особенно полезна в случаях, когда классы в данных несбалансированы или когда ошибки первого и второго рода имеют схожую важность.

Метрика F1-score рассчитывается следующим образом:

F1-score = 2 * (Precision * Recall) / (Precision + Recall)

где:

Precision = TP / (TP + FP) – точность;

Recall = TP / (TP + FN) – полнота;

TP (True Positives) – количество правильно классифицированных положительных объектов;

FP (False Positives) – количество неправильно классифицированных положительных объектов (ложные срабатывания);

FN (False Negatives) – количество неправильно классифицированных положительных объектов (пропущенные срабатывания).

F1-score принимает значения в диапазоне от 0 до 1 (или от 0% до 100%). Чем ближе значение F1-score к 1 (или 100%), тем лучше модель справляется с задачей классификации, учитывая обе метрики Precision и Recall. Если F1-score равен 0, это означает, что модель полностью не справляется с задачей классификации.

Пример № 1: В задаче определения, является ли человек носителем определенной генетической мутации, модель должна быть высоко точной и полной. Если точность модели равна 90%, а полнота – 80%, то F1-score будет равен 84%.

давайте распишем пошаговое решение для метрики F1-score (F-мера) на примере 1:

Рассчитайте точность и полноту модели, используя соответствующие формулы:

Precision = TP / (TP + FP) Recall = TP / (TP + FN)

В данном примере, точность = 0.9 (или 90%) и полнота = 0.8 (или 80%).

Рассчитайте F1-score как гармоническое среднее точности и полноты:

F1-score = 2 * (Precision * Recall) / (Precision + Recall)

F1-score = 2 * (0.9 * 0.8) / (0.9 + 0.8) = 0.84 (или 84%)

Таким образом, в данном примере F1-score равен 84%.

Мы получили F1-score равный 84%, что указывает на то, что модель демонстрирует неплохую производительность с учетом обеих метрик (точность и полнота). Это позволяет оценить модель с более сбалансированной точки зрения по сравнению с использованием только одной из метрик.

Пример № 2: В задаче определения, является ли новость фейковой или нет, модель должна быть высоко точной и полной. Если точность модели равна 85%, а полнота – 90%, то F1-score будет равен 87.5%.

давайте рассмотрим пошаговое решение для метрики F1-score (F-мера) на примере 2:

Рассчитайте точность и полноту модели, используя соответствующие формулы:

Precision = TP / (TP + FP) Recall = TP / (TP + FN)

В данном примере, точность = 0.85 (или 85%) и полнота = 0.9 (или 90%).

Рассчитайте F1-score как гармоническое среднее точности и полноты:

F1-score = 2 * (Precision * Recall) / (Precision + Recall)

F1-score = 2 * (0.85 * 0.9) / (0.85 + 0.9) = 0.875 (или 87.5%)

Таким образом, в данном примере F1-score равен 87.5%.

Метрика ROC AUC

Метрика ROC AUC (Receiver Operating Characteristic – Area Under the Curve) – это метрика качества алгоритма классификации, основанная на анализе ROC-кривой. ROC-кривая представляет собой графическое представление взаимосвязи между чувствительностью (True Positive Rate, TPR) и специфичностью (False Positive Rate, FPR) классификатора при различных пороговых значениях.

True Positive Rate (TPR) или Recall (Полнота) определяется как TP / (TP + FN);

False Positive Rate (FPR) определяется как FP / (FP + TN).

ROC AUC является численным значением, равным площади под ROC-кривой. Оно принимает значения в диапазоне от 0 до 1 (или от 0% до 100%). Чем ближе значение ROC AUC к 1 (или 100%), тем лучше модель справляется с задачей классификации. Значение ROC AUC, равное 0.5, означает, что модель работает на уровне случайного предсказания, а значение, меньше 0.5, указывает на то, что модель предсказывает хуже случайного предсказания.

Преимущества использования метрики ROC AUC заключаются в том, что она не зависит от порога классификации, устойчива к несбалансированным классам и может быть использована для сравнения различных моделей классификации.

Однако стоит отметить, что ROC AUC может давать оптимистичные оценки при наличии сильно несбалансированных классов. В таких случаях рекомендуется использовать другие метрики, такие как Precision-Recall AUC, которые учитывают ошибки первого и второго рода.

Конец ознакомительного фрагмента.