Метрики качества

MSE (Mean Squared Error)

- **Определение**: Среднее значение квадратов ошибок:
  $MSE = \frac{1}{n} * \sum_i^n (ответ_i - предсказание_i)^2$
- **Задача**: Регрессия.
- **Когда использовать**: Когда нужно сильно штрафовать за большие ошибки.

MAE (Mean Absolute Error)

- **Определение**: Среднее значение абсолютных ошибок:
  $MAE = \frac{1}{n} * \sum_i^n |ответ_i - предсказание_i|$
- **Задача**: Регрессия.
- **Когда использовать**: Когда требуется одинаково штрафовать все ошибки независимо от их размера.

ROC-AUC (Receiver Operating Characteristic - Area Under Curve)

- **Определение**: Площадь под кривой зависимости истинно положительных от ложноположительных результатов.
- **Задача**: Классификация.
- **Когда использовать**: Для оценки качества модели при наличии несбалансированных классов.
- **Как считать**: делаем табличку $предсказание|ответ$, сортируем её по убыванию предсказаний, рисуем график. Шкалу X делим на количество 0, а Y - на количество 1. Далее идем по табличке ответов. Если 1 - идем вверх, если 0 - вправо. Получается фигура. Считаем её площадь.

F1 Score

- **Определение**: Гармоническое среднее между точностью (Precision) и полнотой (Recall): $F1Score = 2 * \frac{Precision * Recall}{Precision + Recall}$
- **Задача**: Классификация.
- **Когда использовать**: Когда важно учитывать как точность, так и полноту, особенно при несбалансированных классах.

Accuracy

- **Определение**: Доля правильно предсказанных объектов от общего числа объектов: $Accuracy = \frac{TP+TF}{TP+TN+FP+FN}$.
- **Задача**: Классификация.
- **Когда использовать**: Когда классы сбалансированы и все ошибки равнозначны.
- **Не использовать**: большая разница в количестве элементов классов

Precision

- **Определение**: Доля правильно предсказанных положительных объектов от всех предсказанных положительных объектов: $Precision = \frac{TP}{TP+FP}$
- **Задача**: Классификация.
- **Когда использовать**: Когда важнее минимизировать количество ложноположительных результатов.

Recall

- **Определение**: замеряет способность модели обнаруживать все положительные примеры: $Recall = \frac{TP}{TP+FN}$
- **Задача**: Классификация
- **Когда использовать**: Когда важно минимизировать пропуск положительных примеров

Алгоритмы кластеризации

K-Means

- **Определение**: Алгоритм, который разделяет данные на K кластеров, минимизируя сумму квадратов расстояний от точек до центров своих кластеров.
- **Когда использовать**: Когда кластеры имеют сферическую форму и примерно одинаковый размер. Подходит для больших наборов данных с четко разделенными кластерами.
- **Преимущества**: Простота, скорость.
- **Недостатки**: Требуется заранее задавать количество кластеров (K), чувствительность к начальной инициализации, плохо справляется с кластерами сложной формы и разного размера.

Иерархическая кластеризация

- **Определение**: Алгоритм, который строит иерархию кластеров путем последовательного объединения (агломеративная) или разбиения (дивизионная) кластеров.
- **Когда использовать**: Когда нужно понять структуру данных и их иерархию. Подходит для небольших наборов данных из-за вычислительной сложности.
- **Преимущества**: Не требует заранее задавать количество кластеров, хорошо визуализируется с помощью дендрограмм.
- **Недостатки**: Вычислительно затратен для больших наборов данных, может быть трудно определить оптимальное количество кластеров.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

- **Определение**: Алгоритм, который группирует точки, находящиеся в областях с высокой плотностью, и метит точки в областях с низкой плотностью как выбросы.
- **Когда использовать**: Когда кластеры имеют произвольную форму и плотность, и когда в данных присутствуют выбросы. Подходит для задач, где количество кластеров неизвестно.
- **Преимущества**: Не требует заранее задавать количество кластеров, хорошо справляется с кластерами сложной формы и выбросами.
- **Недостатки**: Чувствителен к выбору параметров ε (eps) и MinPts, может плохо работать с данными переменной плотности.

Bias-Variance-Noise

Bias-Variance-Noise Decomposition — это концепция, используемая для понимания и анализа ошибок в моделях машинного обучения. Она делит общую ошибку на три компонента: смещение (bias), разброс (variance) и шум (noise).

Смещение (Bias)

- **Определение**: Смещение характеризует систематическую ошибку модели, возникающую из-за неправильных предположений о данных. Это разница между средним предсказанием модели и истинным значением.
- **Высокое смещение**: Указывает на недостаточную гибкость модели, которая не может захватить сложность данных (например, линейная модель для сильно

нелинейных данных).
- **Следствие**: Высокое смещение приводит к недообучению (underfitting).

Разброс (Variance)

- **Определение**: Разброс описывает, насколько сильно предсказания модели меняются при использовании различных обучающих наборов. Это чувствительность модели к изменению данных.
- **Высокий разброс**: Указывает на избыточную сложность модели, которая слишком подстраивается под обучающие данные, включая шум (например, полиномиальная модель высокой степени).
- **Следствие**: Высокий разброс приводит к переобучению (overfitting).

Шум (Noise)

- **Определение**: Шум — это неизбежная ошибка, вызванная неустранимыми случайными факторами, присущими самим данным. Шум невозможно уменьшить путем улучшения модели.
- **Источник**: Шум может возникать из-за измерительных ошибок, неполноты данных или других случайных факторов.

Формула общей ошибки (Mean Squared Error)

Общая ошибка модели может быть представлена следующим образом:
$MSE = Bias^2 + Variance + Noise$

Трейд-офф между смещением и разбросом

- **Трейд-офф**: Часто существует компромисс между смещением и разбросом. Увеличивая сложность модели, мы уменьшаем смещение, но увеличиваем разброс, и наоборот.
- **Оптимизация**: Цель заключается в нахождении баланса между смещением и разбросом для минимизации общей ошибки модели.

Примеры

1. **Модель с высоким смещением**: Простая линейная регрессия на сложных данных. Модель не способна захватить все зависимости в данных.
2. **Модель с высоким разбросом**: Полиномиальная регрессия высокой степени на тех же данных. Модель слишком сильно подстраивается под обучающие данные, включая шум.

Формула Байеса

Формула Байеса используется для обновления вероятности гипотезы на основе новых данных. Она выражается следующим образом: $P(A \mid B) = \frac{P(B|A)*P(A)}{P(B)}$

Формула полной вероятности

Формула полной вероятности используется для вычисления вероятности события на основе его возможных исходов. Она выражается следующим образом:
$P(B) = \sum_i P(B|A_i) * P(A_i)$

Пример

Допустим, есть тест на заболевание, который дает положительный результат с вероятностью 99% для больных и 5% для здоровых. Предположим, что 0.1% популяции больны. Какова вероятность того, что человек болен, если его тест положительный?

1. $P(B \mid A)$ вероятность положительного теста при наличии болезни = 0.99
2. $P(B \mid \neg A)$: вероятность положительного теста при отсутствии болезни = 0.05
3. $P(A)$: априорная вероятность болезни = 0.001
4. $P(\neg A)$: априорная вероятность отсутствия болезни = 0.999

Полная вероятность положительного теста:
$P(B) = P(B \mid A)P(A) + P(B \mid \neg A)P(\neg A)$
$P(B) = 0.99 \times 0.001 + 0.05 \times 0.999 = 0.00099 + 0.04995 = 0.05094$

Регулярезация

Регуляризация - способ борьбы с переобучением посредством штрафования за большие веса.

> ⚠ Warning
>
> НЕЛЬЗЯ РЕГУЛЯРИЗОВАТЬ СВОБОДНЫЙ КРЭФФИЦИЕНТ w0.

$L0 : [w1 \neq 0] + [w2 \neq 0]$
$L1 : |w_1| + |w_2|$
$L2 : {w_1}^2 + {w_2}^2$

L0 Регуляризация

L0 регуляризация, также известная как L0-норма регуляризация, заключается в добавлении к функции потерь штрафа, пропорционального количеству ненулевых коэффициентов модели. Формально это можно записать следующим образом:
$L(w) = L_0 + \lambda * \sum_i I(wi \neq 0)$

где:

- L_0 — исходная функция потерь.
- λ — коэффициент регуляризации (гиперпараметр).
- w_i — коэффициенты модели.
- $I(wi \neq 0)$— индикаторная функция, которая равна 1, если $w_i \neq 0$, и 0 в противном случае.

Особенности и преимущества:

1. **Сжатие и отбор признаков**:
    - L0 регуляризация стремится минимизировать количество ненулевых коэффициентов, эффективно выбирая подмножество важных признаков.
2. **Модель с минимальным количеством параметров**:
    - Это приводит к более простой и интерпретируемой модели, так как большинство коэффициентов становятся равными нулю.

L1 Регуляризация (Lasso)

L1 регуляризация добавляет сумму абсолютных значений коэффициентов (весов) к функции потерь. Формула регуляризованной функции потерь: $L(w) = L_0 + \lambda \sum_i | w_i |$
где:

- L_0 — исходная функция потерь.
- λ — коэффициент регуляризации (гиперпараметр).
- w_i — коэффициенты модели.

Особенности:

- Способствует разреженности модели (многие коэффициенты становятся равными нулю).

- Может использоваться для отбора признаков.

L2 Регуляризация (Ridge)

L2 регуляризация добавляет сумму квадратов коэффициентов (весов) к функции потерь. Формула регуляризованной функции потерь: $L(w) = L_0 + \lambda \sum_i w_i^2$

где:

- L_0 — исходная функция потерь.
- λ — коэффициент регуляризации (гиперпараметр).
- w_i — коэффициенты модели.

Особенности:

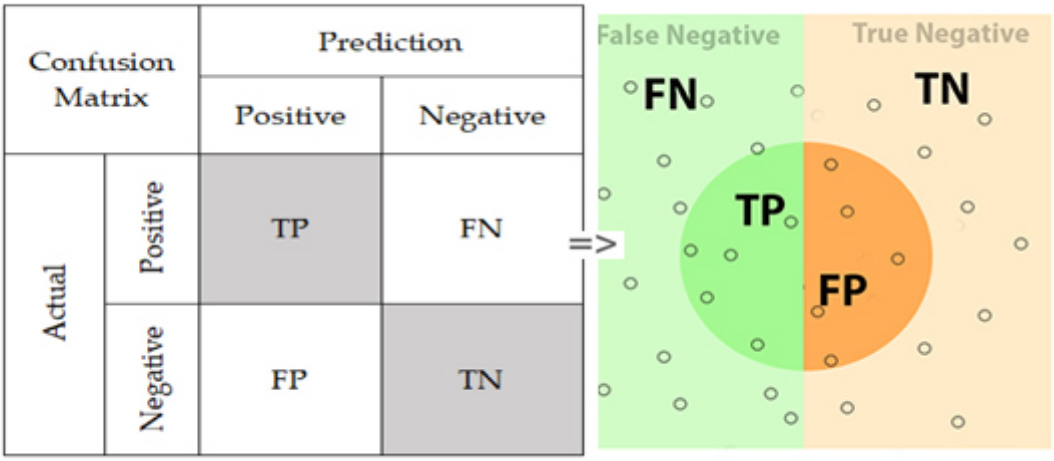
- Способствует уменьшению значений коэффициентов, но редко делает их равными нулю.
- Уменьшает мультиколлинеарность и улучшает устойчивость модели.

Работа с текстом

Стемминг — отрезание окончаний (каждый -> кажд).
Лемматизация — приведение слов к начальной форме.
Токенизация — разбиение текста на кусочки (слова).
Векторизация — перевод текста/картинки в числа.

Матрица ошибок

1 буква - реальный ответ
2 буква - предсказание



4 вопрос

- Коэффициент корреляции Пирсона тем больше, чем ближе распределение к линии
- Даже если величины зависимы, при обучении их не нужно удалять потому что, к примеру, линейной регрессия не сможет обнаружить нелинейную зависимость.

Градиентный спуск

Градиентный спуск — это метод оптимизации, используемый для минимизации функции потерь путем итеративного обновления параметров модели в направлении антиградиента функции потерь.

$$w_{k+1} = w_k - a * \nabla L(w_k)$$

- w_k - k-тый шаг градиентного спуска
- a - параметр быстроты спуска
- $\nabla L(w_k)$ - градиент

> ⚠ Warning
>
> Большой шаг градиентного спуска может привести к расходимости.

Стохастический градиентный спуск (SGD)

Стохастический градиентный спуск (SGD) обновляет параметры модели на основе градиента, вычисленного для одного случайного примера из обучающей выборки.

Сравнение:

- **Обычный градиентный спуск** использует всю выборку для вычисления градиента, что делает его стабильным, но медленным.
- **Стохастический градиентный спуск** использует один пример, что делает его быстрым, но шумным.

Случайный лес (Random Forest)

Главная особенность:

- Множество решающих деревьев, обученных на разных подмножествах данных, комбинируются для улучшения прогноза.

Когда используется:

- Для задач классификации и регрессии, особенно с высокоразмерными данными.

Преимущества:

- Высокая точность.
- Устойчивость к переобучению.
- Обработка пропущенных значений и оценка важности признаков.

Недостатки:

- Медленная предсказательная способность на больших наборах данных.
- Трудность интерпретации результатов.

Бэггинг (Bagging)

Главная особенность:

- Множественные модели обучаются на разных бутстрап-выборках и их предсказания усредняются.

Когда используется:

- Для уменьшения вариативности и улучшения стабильности моделей.

Преимущества:

- Снижение переобучения.
- Увеличение точности моделей.

Недостатки:

- Высокие вычислительные затраты.
- Менее эффективен для сильно шумных данных.

Бустинг (Boosting)

Главная особенность:

- Последовательное обучение слабых моделей, каждая из которых пытается исправить ошибки предыдущих.

Когда используется:

- Для повышения точности моделей, особенно когда базовые модели слабо обучаются.

Преимущества:

- Высокая точность.
- Эффективность на сложных задачах.

Недостатки:

- Высокие вычислительные затраты.
- Риск переобучения при неправильной настройке гиперпараметров.

Модификации бустинга

CatBoost

Главная особенность:

- Эффективная работа с категориальными признаками.

Когда используется:

- Для задач с большим количеством категориальных признаков.

Преимущества:

- Высокая производительность на категориальных данных.
- Минимальная необходимость в предобработке данных.

Недостатки:

- Относительно медленная тренировка на больших наборах данных.

XGBoost

Главная особенность:

- Оптимизация скорости и производительности с использованием регуляризации.

Когда используется:

- Для соревнований по анализу данных и задач, требующих высокой точности.

Преимущества:

- Высокая точность и производительность.
- Многочисленные параметры для настройки.

Недостатки:

- Сложность настройки гиперпараметров.
- Высокие вычислительные затраты.

LightGBM

Главная особенность:

- Быстрое обучение с использованием метода градиентного бустинга.

Когда используется:

- Для больших наборов данных, требующих быстрой обработки.

Преимущества:

- Высокая скорость и низкие требования к памяти.
- Хорошая масштабируемость.

Недостатки:

- Может быть менее точным на небольших наборах данных.
- Чувствительность к качеству данных.

Бутстрап (Bootstrap)

Bootstrap — это статистический метод, который используется для оценки характеристик распределения выборки, путем многократного перепоиска данных. Основная идея

заключается в том, чтобы создать множество подвыборок (бутстрап-выборок) из исходной выборки и использовать эти подвыборки для оценки точности, вычисления доверительных интервалов и других статистических характеристик.

Процесс бутстрапирования включает несколько шагов:

1. **Создание бутстрап-выборок**:
   - Из исходной выборки данных создаются подвыборки путем случайной выборки с возвращением. Это означает, что один и тот же элемент может быть выбран несколько раз в одну бутстрап-выборку.
   - Обычно создается большое количество бутстрап-выборок (например, 1000 или 10000).
2. **Вычисление статистик**:
   - Для каждой бутстрап-выборки вычисляется интересующая статистика (например, среднее, медиана, коэффициент регрессии и т.д.).
3. **Анализ распределения статистик**:
   - Полученные значения статистик из всех бутстрап-выборок используются для построения эмпирического распределения этой статистики.
4. **Оценка характеристик**:
   - Из эмпирического распределения можно вычислить стандартные ошибки, доверительные интервалы и другие характеристики интересующей статистики.

Сравнительная таблица моделей с деревьями:

Модель	Главная особенность	Когда используется	Преимущества	Недостатки
Random Forest	Комбинация решающих деревьев	Классификация и регрессия	Высокая точность, устойчивость	Медленная предсказательна способность
Bagging	Усреднение моделей на бутстрап-выборках	Уменьшение вариативности моделей	Снижение переобучения, увеличение точности	Высокие вычислительные затраты
Boosting	Последовательное обучение моделей	Повышение точности моделей	Высокая точность, эффективность	Высокие вычислительные затраты, риск переобучения
Bootstrap	Статистический перепоиск	Оценка точности и	Простота реализации,	Не улучшает результаты для больших выборо

Модель	Главная особенность	Когда используется	Преимущества	Недостатки
		доверительных интервалов	применимость к малым выборкам	
CatBoost	Эффективность на категориальных признаках	Категориальные данные	Высокая производительность, минимальная предобработка	Относительно медленная тренировка
XGBoost	Оптимизация скорости и производительности	Высокая точность, соревнования	Высокая точность, многочисленные параметры	Сложность настройки, высокие вычислительные затраты
LightGBM	Быстрое обучение	Большие наборы данных	Высокая скорость, хорошая масштабируемость	Менее точен на небольших данных, чувствителен к качеству данных

Кросс-валидация

Кросс-валидация — это метод, предназначенный для оценки качества работы модели, широко применяемый в машинном обучении. Он помогает сравнить между собой различные модели и выбрать наилучшую для конкретной задачи.

Hold-out

Метод **hold-out** представляет из себя простое разделение на train и test.

k-Fold

Метод **k-Fold** чаще всего имеют в виду, когда говорят о кросс-валидации. Он является обобщением метода hold-out и представляет из себя следующий алгоритм:

1. Фиксируется некоторое целое число k (обычно от 5 до 10), меньшее числа семплов в датасете.
2. Датасет разбивается на k одинаковых частей (в последней части может быть меньше семплов, чем в остальных). Эти части называются *фолдами*.
3. Далее происходит k итераций, во время каждой из которых один фолд выступает в роли тестового множества, а объединение остальных — в роли тренировочного. Модель учится на k −1 фолде и тестируется на оставшемся.

4. Финальный скор модели получается либо усреднением k получившихся тестовых результатов, либо измеряется на отложенном тестовом множестве, не участвовавшем в кросс-валидации.

Leave-one-out

Метод **leave-one-out (LOO)** является частным случаем метода k-Fold: в нём каждый фолд состоит ровно из одного семпла. Этот метод может понадобиться в случае, если у вас очень мало данных.

Расстояние между плоскостями

$d = \frac{|D_2 - D_1|}{\sqrt{A^2+B^2+C^2}}$

Ширина между прямыми

$d = \frac{2}{||w||}$

Плотность и функция распределения

Ошибки первого и второго рода

Ошибка первого рода состоит в том, что гипотеза H_0 будет отвергнута, хотя на самом деле она правильная. Вероятность допустить такую ошибку называют **уровнем значимости** и обозначают буквой α («альфа»).

Ошибка второго рода состоит в том, что гипотеза H_0 будет принята, но на самом деле она неправильная. Вероятность совершить эту ошибку обозначают буквой β («бета»). Значение $1-\beta$ называют **мощностью критерия** – это вероятность отвержения неправильной гипотезы.

Дисперсия

Дисперсия σ^2 дискретной случайной величины X:

$$E[X] = \frac{-3+7}{2} = 2$$

$$E[X^2] = \frac{(-3)^2+7^2}{2} = \frac{9+49}{2} = 29$$

$$\sigma^2 = E[X^2] - (E[X])^2 = 29 - 2^2 = 25$$

Ответ: **25**

Кореляция Спирмена

Ниже приведены примеры вычисления корреляций Кенделла и Спирмена. Значения коэффициентов указаны над каждым изображением в виде (x,y) - x - корреляция Кенделла, y - корреляция Спирмена.