ЛЕКЦИЯ 6

МЕТОДЫ КЛАССИФИКАЦИИ И РЕГРЕССИИ

Основные понятия • определения • схемы • таблицы • графики • примеры

Преподаватель: проф.Ахмедиярова А.Т.

Структура и основные разделы

01 Классификация и регрессия как задачи обучения с учителем

02 Постановка задачи классификации

03 Логистическая регрессия

04 Метод k-ближайших соседей для классификации

05 Деревья решений для классификации

06 Метод опорных векторов

07 Оценка качества классификации

08 ROC-кривая и AUC

09 Постановка задачи регрессии

10 Линейная регрессия

11 kNN и деревья в задачах регрессии

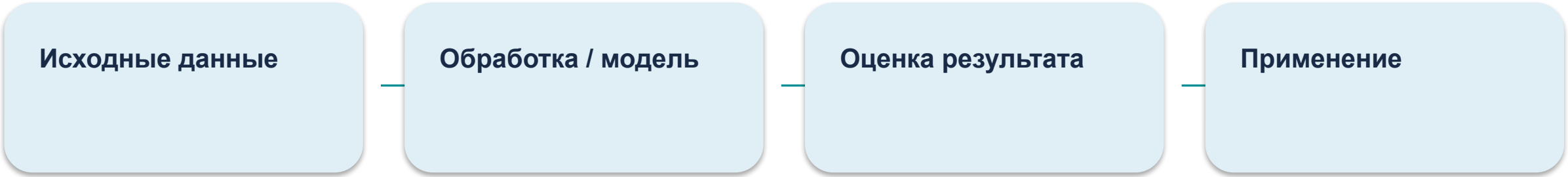
12 Метрики регрессии

Классификация и регрессия как задачи обучения с учителем

Классификация и регрессия относятся к обучению с учителем. В обоих случаях имеется набор объектов, для которых известны входные признаки и правильные ответы. Различие определяется типом целевой переменной. В классификации требуется определить класс объекта, а в регрессии — числовое значение. Рисунок 1 – Классификация и регрессия в обучении с учителем

Ключевая идея: Классификация и регрессия относятся к обучению с учителем. В обоих случаях имеется набор объектов, для которых известны входн

Схема: Классификация и регрессия как задачи обучения с учителем



Последовательность отражает основные этапы рассматриваемого процесса; конкретная реализация зависит от задачи.

Постановка задачи классификации

Пусть каждый объект описывается набором признаков x_1, x_2, …, x_n, а целевая переменная y принимает одно из конечного числа значений. Необходимо построить модель, которая по признакам нового объекта определит его класс. Пример. Для сетевого соединения используются признаки duration, packets, bytes, failed_logins и protocol. Целевая переменная принимает значения normal и attack.

Ключевая идея: Пусть каждый объект описывается набором признаков x_1, x_2, …, x_n, а целевая переменная y принимает одно из конечного числа зна

Постановка задачи классификации — продолжение

Модель должна классифицировать ранее не встречавшееся соединение. При двух классах говорят о бинарной классификации. При трёх и более классах — о многоклассовой классификации. Например, классификация сетевой активности может включать классы normal, DoS, brute force и malware.

Сравнительная таблица: Постановка задачи классификации

Постановка задачи классификации	Пусть каждый объект описывается набором признаков x_1, x_2, …, x_n, а целевая переменная y принимает одно из конечного
Логистическая регрессия	Пример. Для сетевого соединения используются признаки duration, packets, bytes, failed_logins и protocol. Целевая п
Метод k-ближайших соседей для классификации	При двух классах говорят о бинарной классификации. При трёх и более классах — о многоклассовой классификации. Напри

Логистическая регрессия

Логистическая регрессия является одним из базовых методов бинарной классификации. Несмотря на слово «регрессия» в названии, её результатом является оценка вероятности принадлежности объекта к определённому классу. Если целевая переменная кодируется значениями 0 и 1, модель должна выдавать значение в интервале от 0 до 1.

Ключевая идея: Логистическая регрессия является одним из базовых методов бинарной классификации. Несмотря на слово «регрессия» в названии, е

Логистическая регрессия — продолжение

Для этого используется логистическая, или сигмоидальная, функция: $p = 1 / (1 + e^{(-z)})$, где $z = b_0 + b_1x_1 + \ldots + b_nx_n$. При $z = 0$ вероятность равна 0,5. При увеличении z вероятность приближается к 1, а при уменьшении — к 0. Поэтому линейная комбинация признаков преобразуется в корректную вероятностную оценку. Пример. Пусть модель оценила вероятность подозрительного входа как 0,84.

Логистическая регрессия — продолжение

При пороге 0,5 событие будет отнесено к классу suspicious. Если порог повысить до 0,9, то то же событие уже не будет отнесено к положительному классу. Следовательно, порог влияет на число ложных срабатываний и пропущенных событий.

Метод k-ближайших соседей для классификации

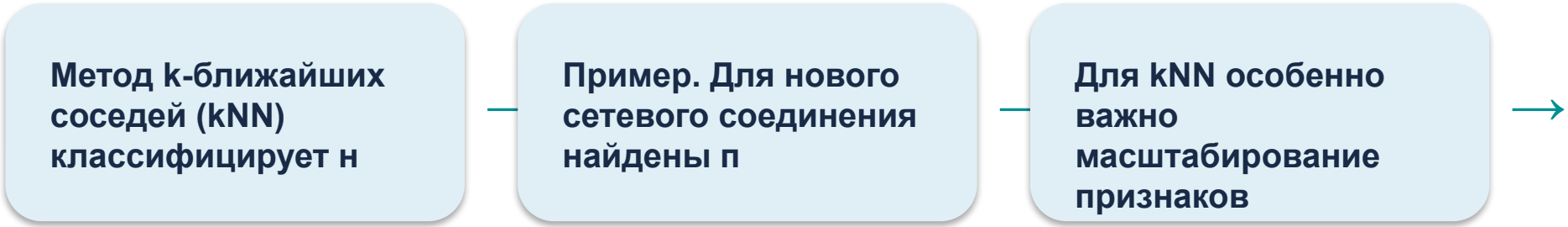
Метод k-ближайших соседей (kNN) классифицирует новый объект по классам наиболее близких к нему объектов обучающей выборки. В учебном пособии подчёркивается, что выбор k критичен: малое значение делает алгоритм чувствительным к шуму, а слишком большое может сгладить границы классов. Пример. Для нового сетевого соединения найдены пять ближайших объектов.

Ключевая идея: Метод k-ближайших соседей (kNN) классифицирует новый объект по классам наиболее близких к нему объектов обучающей выборки. В

Метод k-ближайших соседей для классификации — продолжение

Четыре из них имеют метку attack, один — normal. При k = 5 результатом голосования будет класс attack. Для kNN особенно важно масштабирование признаков. Если один признак измеряется тысячами, а другой — долями единицы, первый может доминировать при вычислении расстояния. В качестве расстояния могут использоваться евклидова, манхэттенская, Чебышёва и другие метрики.

Схема: Метод k-ближайших соседей для классификации



Последовательность отражает основные этапы рассматриваемого процесса; конкретная реализация зависит от задачи.

Деревья решений для классификации

Дерево решений представляет классификационные правила в иерархической структуре. Каждый внутренний узел проверяет некоторый признак, ветви соответствуют результатам проверки, а лист содержит итоговый класс. При построении дерева необходимо выбрать критерий разбиения. В алгоритме C4.5 используется прирост информации. Последовательное разбиение продолжается до выполнения правила остановки.

Ключевая идея: Дерево решений представляет классификационные правила в иерархической структуре. Каждый внутренний узел проверяет некоторый п

Деревья решений для классификации — продолжение

Для ограничения сложности дерева применяется отсечение ветвей — pruning. Преимущество дерева — наглядность. Можно проследить последовательность условий, приведшую к решению. Недостаток — склонность глубоких деревьев к переобучению.

Метод опорных векторов

Метод опорных векторов (Support Vector Machine, SVM) строит разделяющую границу между классами так, чтобы расстояние до ближайших обучающих объектов было максимально возможным. Эти ближайшие объекты называют опорными векторами. Для линейно разделимых данных граница может быть гиперплоскостью.

Ключевая идея: Метод опорных векторов (Support Vector Machine, SVM) строит разделяющую границу между классами так, чтобы расстояние до ближа

Метод опорных векторов — продолжение

Для более сложных зависимостей применяются ядровые функции, позволяющие строить нелинейные разделения в преобразованном пространстве признаков. Пример. При классификации текстовых описаний инцидентов после преобразования текста в числовые векторы SVM может отделять одну категорию событий от другой по комбинации большого числа признаков.

Сравнительная таблица: Метод опорных векторов

Метод опорных векторов	Метод опорных векторов (Support Vector Machine, SVM) строит разделяющую границу между классами так, чтобы расстояни
Оценка качества классификации	Для линейно разделимых данных граница может быть гиперплоскостью. Для более сложных зависимостей применяются ядровы
ROC-кривая и AUC	Пример. При классификации текстовых описаний инцидентов после преобразования текста в числовые векторы SVM может от

Оценка качества классификации

Результат классификатора оценивается не только общей долей правильных ответов. Для бинарной классификации удобно использовать матрицу ошибок, которая показывает количество правильных и ошибочных решений для каждого класса. Рисунок 2 – Иллюстративная матрица ошибок: 1000 объектов, из них 50 атак Для показанной иллюстрации TP = 30, TN = 920, FP = 30, FN = 20.

Ключевая идея: Результат классификатора оценивается не только общей долей правильных ответов. Для бинарной классификации удобно использовать

Оценка качества классификации — продолжение

Из этих четырёх значений вычисляются основные метрики. Accuracy = (TP + TN) / (TP + TN + FP + FN). Precision = TP / (TP + FP). Recall = TP / (TP + FN). F1 = 2 × Precision × Recall / (Precision + Recall). В нашем иллюстративном примере accuracy = 95%, precision = 50%, recall = 60%. Это показывает, почему высокая accuracy сама по себе не гарантирует хорошего обнаружения редкого класса.

Схема: Оценка качества классификации

Результат классификатора оценивается не только о

Рисунок 2 – Иллюстративная матрица ошибок: 1000

Для показанной иллюстрации TP = 30, TN = 920, FP

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

Последовательность отражает основные этапы рассматриваемого процесса; конкретная реализация зависит от задачи.

ROC-кривая и AUC

Если классификатор выдаёт вероятность или оценку принадлежности к классу, его поведение можно исследовать при разных порогах. ROC-кривая показывает соотношение между долей правильно обнаруженных положительных объектов и долей ложноположительных решений при изменении порога. AUC — площадь под ROC-кривой. Чем выше AUC, тем лучше модель в среднем разделяет классы при различных порогах.

Ключевая идея: Если классификатор выдаёт вероятность или оценку принадлежности к классу, его поведение можно исследовать при разных порогах.

ROC-кривая и AUC — продолжение

Однако выбор конкретного порога всё равно должен учитывать цену ошибок в прикладной системе. В информационной безопасности нельзя выбирать порог только по максимальной общей точности. Для критичного сервиса может быть важнее высокий recall, тогда как для системы с дорогими ложными блокировками требуется контролировать false positive.

Постановка задачи регрессии

В регрессии целевая переменная является числовой. Требуется построить функцию, которая по входным признакам оценивает значение y. Примеры: прогноз времени ответа сервера; прогноз нагрузки; оценка объёма трафика; прогноз количества обращений; оценка времени устранения инцидента. Рисунок 3 – Иллюстрация линейной зависимости и линии регрессии

Ключевая идея: В регрессии целевая переменная является числовой. Требуется построить функцию, которая по входным признакам оценивает значени

Линейная регрессия

Линейная регрессия описывает зависимость целевой переменной от признаков линейной функцией. Для одного признака модель записывается как $\hat{y} = b_0 + b_1 x$. Для нескольких признаков используется сумма взвешенных признаков. Задача обучения состоит в подборе коэффициентов, при которых прогнозы максимально близки к фактическим значениям.

Ключевая идея: Линейная регрессия описывает зависимость целевой переменной от признаков линейной функцией. Для одного признака модель записы

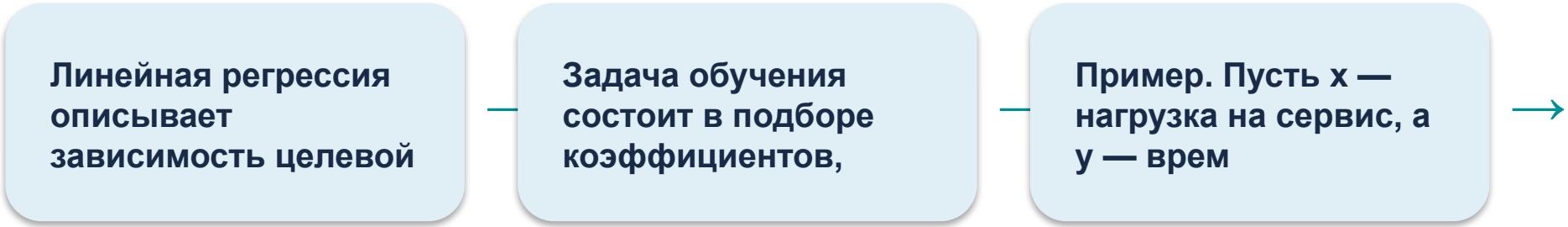
Линейная регрессия — продолжение

В загруженном пособии эта задача рассматривается как задача оптимизации; для простой линейной регрессии используется метод наименьших квадратов и среднеквадратическая ошибка MSE. Пример. Пусть x — нагрузка на сервис, а y — время ответа. После обучения модель может показать, как ожидаемое время ответа изменяется при увеличении нагрузки.

Линейная регрессия — продолжение

Если зависимость существенно нелинейна, простая линейная модель может быть недостаточной.

Схема: Линейная регрессия



Последовательность отражает основные этапы рассматриваемого процесса; конкретная реализация зависит от задачи.

Сравнительная таблица: Линейная регрессия

Линейная регрессия	Линейная регрессия описывает зависимость целевой переменной от признаков линейной функцией. Для одного признака мод
kNN и деревья в задачах регрессии	Задача обучения состоит в подборе коэффициентов, при которых прогнозы максимально близки к фактическим значениям. В
Метрики регрессии	Пример. Пусть x — нагрузка на сервис, а y — время ответа. После обучения модель может показать, как ожидаемое время

kNN и деревья в задачах регрессии

Идея kNN может применяться и к регрессии. Вместо голосования по классам берётся среднее или медианное значение целевой переменной у ближайших соседей. Например, время обработки нового запроса можно оценить по времени обработки наиболее похожих запросов. Деревья также могут решать задачи регрессии.

Ключевая идея: Идея kNN может применяться и к регрессии. Вместо голосования по классам берётся среднее или медианное значение целевой переме

kNN и деревья в задачах регрессии — продолжение

Пространство признаков разделяется условиями, а в листе формируется числовой прогноз, например среднее значение целевой переменной для объектов соответствующей области.

Метрики регрессии

Пример. Истинное время обработки трёх запросов равно 10, 20 и 30 мс, а прогнозы — 12, 18 и 33 мс. Абсолютные ошибки равны 2, 2 и 3 мс, поэтому MAE = (2 + 2 + 3) / 3 ≈ 2,33 мс.

Дисбаланс классов

В реальных задачах безопасности положительный класс часто встречается значительно реже нормального. Например, из 100 000 событий подавляющее большинство может быть нормальным. В такой ситуации модель способна получить высокую accuracy, почти всегда выбирая большинство, но плохо обнаруживать атаки.

Ключевая идея: В реальных задачах безопасности положительный класс часто встречается значительно реже нормального. Например, из 100 000 собы

Дисбаланс классов — продолжение

Поэтому необходимо анализировать распределение классов, confusion matrix, precision, recall и F1-score. При разделении выборки полезно сохранять пропорции классов. При необходимости применяются методы балансировки, изменение весов классов и настройка порога решения.

Сравнительная таблица: Дисбаланс классов

Дисбаланс классов	В реальных задачах безопасности положительный класс часто встречается значительно реже нормального. Например, из 10
Ошибки первого и второго типа в прикладной системе	Поэтому необходимо анализировать распределение классов, confusion matrix, precision, recall и F1-score. При разделе

Ошибки первого и второго типа в прикладной системе

Нельзя утверждать, что одна из ошибок всегда важнее другой. Их цена определяется контекстом. Например, для системы блокировки критичных транзакций большое число false positive может нарушить работу пользователей, а для раннего обнаружения опасной атаки особенно критичен false negative.

Выбор метода классификации или регрессии

Выбор должен подтверждаться экспериментом. Несколько моделей обучают на одинаковой обучающей выборке и сравнивают на одинаковых независимых данных с использованием подходящих метрик.

Ключевая идея: Выбор должен подтверждаться экспериментом. Несколько моделей обучают на одинаковой обучающей выборке и сравнивают на одинаков

Выбор метода классификации или регрессии — продолжение

Пример построения классификационной модели в Python from sklearn.model_selection import train_test_split from sklearn.preprocessing import StandardScaler from sklearn.linear_model import LogisticRegression from sklearn.metrics import classification_report, confusion_matrix X = data[['duration', 'packets', 'bytes', 'failed_logins']] y = data['label'] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y) scaler = StandardScaler() X_train = scaler.fit_transform(X_train) X_test = scaler.transform(X_test) model = LogisticRegression() model.fit(X_train, y_train) y_pred = model.predict(X_test) print(confusion_matrix(y_test, y_pred)) print(classification_report(y_test, y_pred)) Пример построения регрессионной модели в Python from sklearn.model_selection import train_test_split from sklearn.linear_model import LinearRegression from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score X = data[['users', 'traffic', 'cpu_load']] y = data['response_time'] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42) model = LinearRegression() model.fit(X_train, y_train) y_pred = model.predict(X_test) print('MAE:', mean_absolute_error(y_test, y_pred)) print('MSE:', mean_squared_error(y_test, y_pred)) print('R2:', r2_score(y_test, y_pred))

Схема: Выбор метода классификации или регрессии

Выбор должен подтверждаться экспериментом. Неско

Пример построения классификационной модели в Pyt

from sklearn.model_selectio n import train_test_s

Пример построения регрессионной модели в Python

Последовательность отражает основные этапы рассматриваемого процесса; конкретная реализация зависит от задачи.

Развёрнутый пример: классификация сетевого трафика

Рассмотрим набор сетевых соединений с признаками duration, protocol, packets, bytes и failed_logins. Цель — определить normal или attack. Сначала проверяется качество данных и распределение классов. Затем данные разделяются на обучающую и тестовую части. Для методов, чувствительных к масштабу, выполняется стандартизация. В качестве первой модели можно использовать логистическую регрессию.

Ключевая идея: Рассмотрим набор сетевых соединений с признаками duration, protocol, packets, bytes и failed_logins. Цель — определить normal

Развёрнутый пример: классификация сетевого трафика — продолжение

Она формирует вероятностную оценку и удобна как интерпретируемая базовая модель. Затем результат можно сравнить с kNN, деревом решений или SVM. Сравнение проводится на одной тестовой выборке. Если логистическая регрессия имеет высокий precision, но недостаточный recall, часть атак остаётся необнаруженной. Можно исследовать изменение порога или другую модель.

Развёрнутый пример: классификация сетевого трафика — продолжение

Если дерево показывает почти идеальный результат на train, но значительно худший на test, вероятно переобучение. Для kNN необходимо проверить масштабирование и подобрать k. Итоговая модель выбирается не по одному числу, а по совокупности требований: качество обнаружения, число ложных тревог, скорость работы, интерпретируемость и устойчивость на новых данных.

Развёрнутый пример: классификация сетевого трафика — продолжение

Основные выводы: Классификация прогнозирует дискретный класс, а регрессия — числовое значение. Логистическая регрессия оценивает вероятность класса и широко применяется в бинарной классификации. kNN использует близость объектов и требует внимательного выбора k и масштабирования признаков.

Развёрнутый пример: классификация сетевого трафика — продолжение

Деревья решений формируют последовательность интерпретируемых правил, но требуют контроля переобучения. SVM строит разделяющую границу с максимальным зазором и эффективен в пространствах с большим числом признаков. Для классификации необходимо анализировать confusion matrix, precision, recall, F1 и при необходимости ROC-AUC. Для регрессии применяются MAE, MSE, MAPE и R^2.

График: иллюстрация зависимости



Иллюстративные значения (не результаты эксперимента).

Итоги и вопросы для обсуждения

1. Объясните понятие «Классификация и регрессия как задачи обучения с учителем» и приведите пример.
2. Объясните понятие «Постановка задачи классификации» и приведите пример.
3. Объясните понятие «Логистическая регрессия» и приведите пример.
4. Объясните понятие «Метод k-ближайших соседей для классификации» и приведите пример.
5. Объясните понятие «Деревья решений для классификации» и приведите пример.
6. Объясните понятие «Метод опорных векторов» и приведите пример.
7. Объясните понятие «Оценка качества классификации» и приведите пример.