

СНИЖЕНИЕ РАЗМЕРНОСТИ И СОКРАЩЕНИЕ ДАННЫХ

ЛЕКЦИЯ ПО ДИСЦИПЛИНЕ

«ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ» (DATA MINING)

ПЛАН ЛЕКЦИИ

- Понятие снижения размерности и сокращения данных
- Проблема высокой размерности
- Отбор и извлечение признаков
- Методы PCA, LDA и t-SNE
- Сокращение количества записей и сжатие данных
- Практическая реализация на Python

ЦЕЛЬ И ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ

Цель лекции

Изучить методы снижения размерности и сокращения данных, понять принципы PCA, LDA, t-SNE и освоить их применение.

После лекции студент сможет

- Объяснять проблему высокой размерности
- Различать отбор и извлечение признаков
- Сравнивать PCA, LDA и t-SNE
- Выбирать метод под задачу
- Применять методы в Python

ПОНЯТИЕ СНИЖЕНИЯ РАЗМЕРНОСТИ И СОКРАЩЕНИЯ ДАННЫХ

При анализе поведения пользователей интернет-магазина могут использоваться следующие характеристики:

- возраст пользователя;
- пол;
- регион проживания;
- количество покупок;
- средняя стоимость заказа;
- частота посещения сайта;
- время пребывания на сайте;
- история просмотров товаров;
- количество возвратов;
- предпочтения пользователей.

ПОНЯТИЕ СНИЖЕНИЯ РАЗМЕРНОСТИ И СОКРАЩЕНИЯ ДАННЫХ

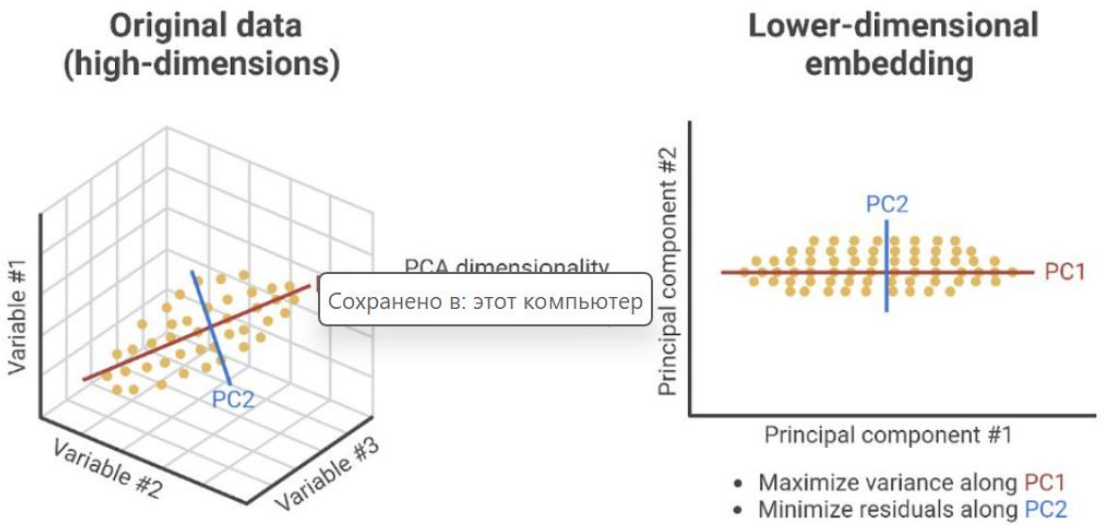
Большое количество признаков создаёт определённые трудности:

- увеличивается время обработки данных;
- возрастает потребность в вычислительных ресурсах;
- усложняется визуализация;
- появляются избыточные и коррелирующие признаки;
- может снижаться качество моделей машинного обучения;
- усложняется интерпретация результатов.

1. ОСНОВНЫЕ ПОНЯТИЯ

Снижение размерности

Умен ктному представлению
с сох



1. СНИЖЕНИЕ РАЗМЕРНОСТИ

Математически исходны представить матрицей:

$$X \in \mathbb{R}^{n \times p}$$

где:

- n — количество объектов;
- p — количество признаков.

После снижения размерности получаем:

$$Z \in \mathbb{R}^{n \times k}$$

где k<p — новое количество признаков.

1. СНИЖЕНИЕ РАЗМЕРНОСТИ

Например:

$$X \in \mathbb{R}^{1000 \times 50}$$

После преобразования:

$$Z \in \mathbb{R}^{1000 \times 5}$$

Таким образом, количество признаков уменьшается с 50 до 5.

1. ОСНОВНЫЕ ПОНЯТИЯ

Сокращение данных

это процесс уменьшения объёма данных при сохранении информации, необходимой для решения конкретной аналитической задачи.

Сокращение данных может осуществляться несколькими способами:

- уменьшение количества признаков;
- уменьшение количества записей;
- агрегирование данных;
- сжатие данных;
- построение компактного представления данных.

СНИЖЕНИЕ РАЗМЕРНОСТИ И СОКРАЩЕНИЕ ДАННЫХ

Критерий	Снижение размерности	Сокращение данных
Цель	Уменьшить число признаков	Уменьшить общий объём данных
Что меняется	Признаки или их представление	Записи, признаки, хранение
Методы	PCA, LDA, t-SNE, отбор	Выборка, агрегация, сжатие
Пример	50 признаков → 5 компонент	1 млн записей → 100 тыс.

2. ПРОБЛЕМА ВЫСОКОЙ РАЗМЕРНОСТИ

- Проклятие размерности — трудности анализа при большом числе признаков.
- Растут вычислительные затраты и требования к памяти.
- Избыточные признаки могут дублировать информацию.
- Шумовые признаки могут ухудшать работу моделей.
- Визуализация многомерных данных становится сложнее.

2.1. ПРОКЛЯТИЕ РАЗМЕРНОСТИ

Проклятие размерности — это совокупность проблем, возникающих при анализе данных с большим количеством признаков.
При увеличении размерности пространства его объём растёт очень быстро. Для равномерного покрытия такого пространства требуется значительно больше объектов.

2.1. ПРОКЛЯТИЕ РАЗМЕРНОСТИ

Пример

Пусть каждый признак принимает значения от 0 до 1.
Если у нас один признак, то пространство представляет собой отрезок:

$$V=1$$

Если два признака:

$$V=1x1=1$$

Если десять признаков:

$$V = 1^{10} = 1$$

Сам объём единичного гиперкуба не изменяется, но для его равномерного заполнения с заданной детализацией требуется экспоненциально больше точек.

2.1. ПРОКЛЯТИЕ РАЗМЕРНОСТИ

Если разделить каждый признак на 10 интервалов, то количество ячеек составит:

$$1^d$$

где d — количество признаков.

Количество признаков	Количество ячеек
1	10
2	100
3	1 000
5	100 000
10	10 000 000 000

Это показывает, почему при высокой размерности для качественного анализа требуется значительно больше данных.

2.2. ОСНОВНЫЕ ПРОБЛЕМЫ ВЫСОКОЙ РАЗМЕРНОСТИ

1. Увеличение вычислительной сложности.

Чем больше признаков, тем больше вычислений необходимо выполнить алгоритму.

2. Избыточность признаков.

Некоторые признаки могут содержать практически одинаковую информацию.

Например:

- возраст пользователя;
- год рождения пользователя.

Эти два признака сильно связаны между собой.

2.2. ОСНОВНЫЕ ПРОБЛЕМЫ ВЫСОКОЙ РАЗМЕРНОСТИ

3. **Шумовые признаки.**
Некоторые характеристики не имеют существенного отношения к прогнозируемому результату и могут ухудшать работу модели.
4. **Снижение эффективности алгоритмов**.
Алгоритмы, основанные на расстояниях, например KNN и K-means, могут работать менее эффективно в пространствах высокой размерности.
5. **Сложность визуализации.**
Человеку трудно непосредственно анализировать данные, содержащие десятки и сотни признаков.

3. ОСНОВНЫЕ ПОДХОДЫ К СНИЖЕНИЮ РАЗМЕРНОСТИ

Feature Selection — отбор признаков

Выбираются наиболее полезные исходные признаки.

Filter: корреляция, Chi-Square, Mutual Information

Wrapper: Forward Selection, RFE

Embedded: Lasso, Random Forest

Feature Extraction — извлечение признаков

Исходные признаки преобразуются в новое пространство.

PCA

LDA

SVD

t-SNE

3.1. ОТБОР ПРИЗНАКОВ (FEATURE SELECTION)

Отбор признаков — это процесс выбора наиболее информативных признаков из исходного набора без изменения их первоначального содержания.

Например, исходный набор содержит 20 признаков. После анализа выбираются 5 наиболее значимых признаков.

$$X = (x_1, x_2, \ldots, x_{20})$$

После отбора:

$$X' = (x_2, x_5, x_8, x_{12}, x_{17})$$

Оставшиеся признаки исключаются из анализа.

ОСНОВНЫЕ МЕТОДЫ ОТБОРА ПРИЗНАКОВ:

1. **Filter Methods — фильтрационные методы.**

Признаки оцениваются независимо от конкретной модели машинного обучения.
Примеры:

- корреляционный анализ;
- Chi-Square;
- Mutual Information;
- дисперсионный фильтр (Variance Threshold).

ОСНОВНЫЕ МЕТОДЫ ОТБОРА ПРИЗНАКОВ:

2. Wrapper Methods — методы обёртки.

Для оценки наборов признаков используется конкретная модель машинного обучения.

Примеры:
Forward Selection;
Backward Elimination;
Recursive Feature Elimination (RFE).

ОСНОВНЫЕ МЕТОДЫ ОТБОРА ПРИЗНАКОВ:

3. Embedded Methods — встроенные методы.

Отбор признаков выполняется непосредственно в процессе обучения модели.

Примеры:
Lasso Regression;
Decision Tree;
Random Forest.

3.2. ИЗВЛЕЧЕНИЕ ПРИЗНАКОВ (FEATURE EXTRACTION)

Извлечение признаков — это преобразование исходных признаков в новое пространство с меньшим количеством измерений.

В результате могут формироваться новые признаки, являющиеся комбинациями исходных.
Основные методы:
PCA — Principal Component Analysis;
LDA — Linear Discriminant Analysis;
SVD — Singular Value Decomposition;
t-SNE — t-distributed Stochastic Neighbor Embedding.

НАПРИМЕР, ИСХОДНЫЕ ПРИЗНАКИ:

$$X = (x_1, x_2, x_3, x_4)$$

Могут быть преобразованы в два новых признака:

$$Z = (z_1, z_2)$$

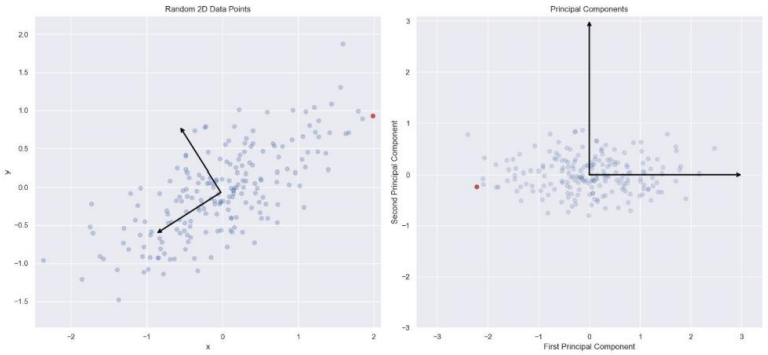
где z_1 и z_2 являются линейными комбинациями исходных признаков для PCA.

4. PCA — МЕТОД ГЛАВНЫХ КОМПОНЕНТ

PCA (Principal Component Analysis) — метод главных компонент, который используется для уменьшения размерности данных путём преобразования исходных признаков в новые взаимно ортогональные переменные — главные компоненты.

Основная идея PCA заключается в том, чтобы найти направления, вдоль которых данные имеют наибольшую дисперсию.

Первая главная компонента объясняет максимально возможную долю общей дисперсии данных. Вторая компонента объясняет максимально возможную долю оставшейся дисперсии при условии ортогональности первой компоненты.



4.1. ОСНОВНЫЕ ЭТАПЫ АЛГОРИТМА PCA

Шаг 1. Подготовка данных.

Перед применением PCA необходимо проверить пропущенные значения и подготовить признаки.
Часто используется стандартизация:

$$z = \frac{x - \mu}{\sigma}$$

где:

- x — исходное значение;
- μ — среднее значение признака;
- σ — стандартное отклонение.

4.1. ОСНОВНЫЕ ЭТАПЫ АЛГОРИТМА PCA

Шаг 3. Вычисление собственных значений и собственных векторов.

Для ковариационной матрицы определяются собственные значения и собственные векторы:

$$Cv = \lambda v$$

где:

- C — ковариационная матрица;
- v — собственный вектор;
- λ — собственное значение.

4.1. ОСНОВНЫЕ ЭТАПЫ АЛГОРИТМА PCA

Шаг 4. Выбор главных компонент.

Компоненты упорядочиваются по убыванию собственных значений. Выбираются первые k компонент, которые объясняют необходимую долю общей дисперсии.

4.1. ОСНОВНЫЕ ЭТАПЫ АЛГОРИТМА PCA

Шаг 5. Преобразование данных.

Исходные данные проецируются на выбранные компоненты:

$$Z = XW$$

где:

- X — центрированная матрица исходных данных;
- W — матрица выбранных собственных векторов;
- Z — преобразованные данные.

4.2. ДОЛЯ ОБЪЯСНЁННОЙ ДИСПЕРСИИ

Для каждой главной компоненты рассчитывается доля объяснённой дисперсии:

$$EVR_i = \frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j}$$

Накопленная доля объяснённой дисперсии:

$$EVR_{cum}(k) = \sum_{i=1}^{k} EVR_i$$

НАПРИМЕР, ПОЛУЧЕНЫ СЛЕДУЮЩИЕ ЗНАЧЕНИЯ:

Главная компонента	Доля объяснённой дисперсии
PC1	45%
PC2	30%
PC3	15%
PC4	7%
PC5	3%

Если выбрать первые три компоненты, то сохранённая доля дисперсии составит:
45%+30%+15%=90%

Таким образом, вместо пяти признаков можно использовать три компоненты, сохраняющие 90% общей дисперсии.

4.3. ПРЕИМУЩЕСТВА И НЕДОСТАТКИ PCA

Преимущества:

- уменьшает количество признаков;
- позволяет снизить вычислительные затраты;
- помогает устранить линейную корреляцию между новыми компонентами;
- позволяет визуализировать многомерные данные;
- может уменьшать влияние избыточных признаков.

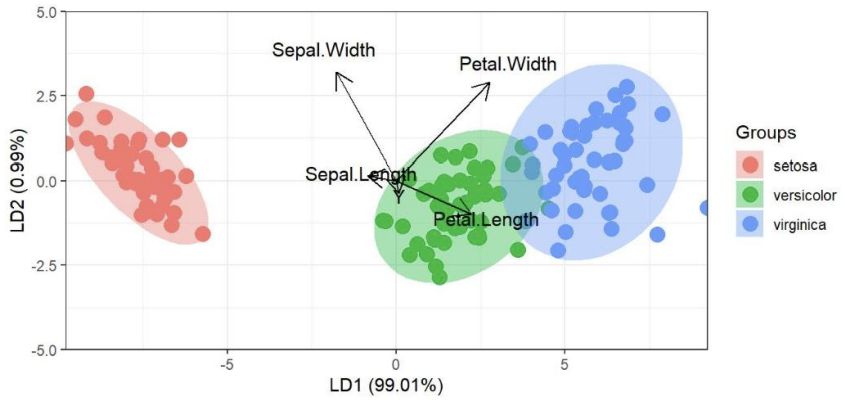
Недостатки:

- новые компоненты сложнее интерпретировать;
- PCA не использует целевую переменную;
- чувствителен к масштабам признаков;
- линейный метод может не выявлять сложные нелинейные структуры.

5. LDA — ЛИНЕЙНЫЙ ДИСКРИМИНАНТНЫЙ АНАЛИЗ

LDA (Linear Discriminant Analysis) — линейный метод снижения размерности, который использует информацию о классах и стремится найти такие направления проекции, при которых классы максимально разделены.

В отличие от PCA, который ищет направления максимальной общей дисперсии, LDA учитывает принадлежность объектов к известным классам.



Цель: максимизировать отношение межклассового разброса к внутриклассовому.

5.1. ОСНОВНАЯ ИДЕЯ LDA

Предположим, необходимо классифицировать пользователей на три категории:

- активные;
- умеренно активные;
- неактивные.

Каждый пользователь описывается десятью признаками.

LDA ищет такое преобразование данных, при котором:

- объекты одного класса располагаются близко друг к другу;
- объекты разных классов располагаются как можно дальше друг от друга.

5.2. МАТЕМАТИЧЕСКАЯ ОСНОВА LDA

LDA использует две матрицы:

1. Внутриклассовая матрица разброса (Within-Class Scatter Matrix):

$$S_W = \sum_{c=1}^{C} \sum_{x \in D_c} (x - \mu_c)(x - \mu_c)^T$$

где:

- C — количество классов;
- D_c — множество объектов класса c;
- μ_c — среднее значение признаков класса.

5.2. МАТЕМАТИЧЕСКАЯ ОСНОВА LDA

LDA использует две матрицы:

2. Межклассовая матрица разброса (Between-Class Scatter Matrix):

$$S_B = \sum_{c=1}^{C} N_c(\mu_c - \mu)(\mu_c - \mu)^T$$

где:

- N_c — количество объектов класса;
- μ_c — среднее значение признаков класса;
- μ — общее среднее значение признаков.

5.2. МАТЕМАТИЧЕСКАЯ ОСНОВА LDA

Цель LDA — найти направления, максимизирующие отношение межклассового разброса к внутриклассовому:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}$$

5.3. ОСОБЕННОСТИ LDA

Количество компонент, которые можно получить с помощью классического LDA, не превышает:

$$k \leq C - 1$$

где C— количество классов.

Например:

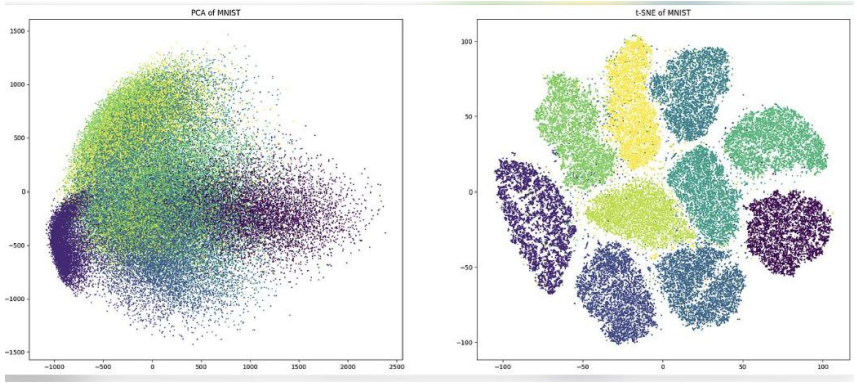
- для двух классов можно получить максимум одну компоненту;
- для трёх классов — максимум две компоненты;
- для четырёх классов — максимум три компоненты.

PCA И LDA: СРАВНЕНИЕ

Критерий	PCA	LDA
Обучение	Без учителя	С учителем
Метки классов	Не использует	Использует
Цель	Сохранить общую дисперсию	Разделить классы
Критерий	Максимум дисперсии	Межклассовый / внутриклассовый разброс
Компоненты	Не больше исходной размерности	Не больше C – 1
Применение	Сжатие, визуализация	Разделение классов

6. T-SNE — НЕЛИНЕЙНАЯ ВИЗУАЛИЗАЦИЯ

t-SNE (t-distributed Stochastic Neighbor Embedding) — нелинейный метод снижения размерности, предназначенный преимущественно для визуализации многомерных данных в двумерном или трёхмерном пространстве.
Основная идея заключается в сохранении локальных соседств: объекты, близкие друг к другу в исходном пространстве, должны с высокой вероятностью располагаться рядом в пространстве меньшей размерности.



6.1. ПРИНЦИП РАБОТЫ T-SNE

Алгоритм выполняет следующие действия:

1. Вычисляет сходство между объектами в исходном многомерном пространстве.
2. Преобразует расстояния между объектами в вероятности соседства.
3. Создаёт начальное расположение объектов в пространстве меньшей размерности.
4. Вычисляет сходство между объектами в новом пространстве.
5. Минимизирует различие между распределениями сходства в исходном и новом пространствах.

6.1. ПРИНЦИП РАБОТЫ T-SNE

Для измерения различия используется дивергенция Кульбака — Лейблера:

$$KL(P\|Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

где:

- P — распределение сходства объектов в исходном пространстве;
- Q — распределение сходства объектов в пространстве меньшей размерности.

6.2. ПАРАМЕТР PERPLEXITY

Одним из важных параметров t-SNE является perplexity.
Он связан с эффективным количеством соседей, учитываемых при оценке локальной структуры данных.
Типичные значения параметра могут находиться в диапазоне от 5 до 50, однако оптимальное значение зависит от объёма и структуры данных.

T-SNE: ПРЕИМУЩЕСТВА И ОГРАНИЧЕНИЯ

Преимущества

- Выявляет локальные группы объектов
- Отображает нелинейные структуры
- Полезен для визуального исследования данных

Ограничения

- Вычислительно затратен
- Зависит от параметров и инициализации
- Глобальные расстояния могут искажаться
- Не является универсальным методом прогнозирования

Визуально разделённые группы не доказывают высокую точность классификации.

7. МЕТОДЫ СОКРАЩЕНИЯ ДАННЫХ

Sampling — выборка

Выбор части объектов.

- Случайная
- Стратифицированная

Aggregation — агрегация

Объединение записей.

Например: ежедневные продажи → недельная сумма.

Compression — сжатие

Без потерь: ZIP, GZIP

С потерями: JPEG, MP3

Дискретизация преобразует непрерывные значения в интервалы или категории.

8. СРАВНЕНИЕ МЕТОДОВ

Метод	Основная идея	Метки классов	Задача
PCA	Максимизация дисперсии	Нет	Сжатие признаков
LDA	Разделение классов	Да	Дискриминантный анализ
t-SNE	Локальное сходство	Нет	Визуализация
Feature Selection	Выбор исходных признаков	Зависит	Удаление лишних признаков
SVD	Разложение матрицы	Нет	Компактное представление
Sampling	Выбор части объектов	Не обязательно	Сокращение записей
Aggregation	Объединение записей	Не обязательно	Уменьшение детализации

9. ПРАКТИКА PYTHON: PCA

```
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA

iris = load_iris()
X, y = iris.data, iris.target
X_scaled = StandardScaler().fit_transform(X)

pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

print(pca.explained_variance_ratio_)
print(sum(pca.explained_variance_ratio_))
```

Что делает код?

1. Загружает Iris.
2. Стандартизирует признаки.
3. Преобразует 4 признака в 2 компоненты.
4. Выводит объяснённую дисперсию.

ПРАКТИКА PYTHON: LDA И T-SNE

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.manifold import TSNE

lda = LinearDiscriminantAnalysis(n_components=2)
X_lda = lda.fit_transform(X_scaled, y)

tsne = TSNE(n_components=2, perplexity=30, random_state=42)
X_tsne = tsne.fit_transform(X_scaled)
```

Важно

- LDA использует y — метки классов.
- t-SNE предназначен прежде всего для визуализации.
- Сравните графики трёх методов.

10. ТИПИЧНЫЕ ОШИБКИ

- Не стандартизировать признаки перед PCA при разных масштабах.
- Слишком сильно уменьшать размерность и терять полезную информацию.
- Допускать утечку данных при отборе признаков до разделения выборки.
- Считать расстояния между группами t-SNE точными исходными расстояниями.
- Оценивать результат только по графику, не проверяя основную задачу.
- Считать, что максимальная дисперсия гарантирует лучшую точность прогноза.

11. ПРАКТИЧЕСКИЕ ОБЛАСТИ ПРИМЕНЕНИЯ

Медицина

Медицинские показатели, генетические данные, медицинские изображения.

Финансы

Транзакции, выявление мошенничества, поведение клиентов.

Тексты и IoT

TF-IDF, датчики, температура, вибрация, энергопотребление.

ПРАКТИЧЕСКОЕ ЗАДАНИЕ

- Загрузить набор данных Iris и определить исходную размерность.
- Выполнить стандартизацию признаков.
- Применить PCA к двум компонентам и вычислить объяснённую дисперсию.
- Применить LDA и t-SNE.
- Построить три графика и сравнить результаты.
- Сформулировать вывод о применимости методов.

Отчёт

Цель, описание данных, код, графики, сравнение и выводы.

ЗАКЛЮЧЕНИЕ

- Снижение размерности упрощает анализ многомерных данных.
- PCA сохраняет направления максимальной общей дисперсии.
- LDA использует метки классов для их разделения.
- t-SNE преимущественно применяется для визуализации локальной структуры.
- Выборка, агрегация и сжатие уменьшают общий объём данных.
- Метод выбирают с учётом цели анализа и проверяют качество результата.

КОНТРОЛЬНЫЕ ВОПРОСЫ

- 1. Что такое снижение размерности и сокращение данных?
- 2. Что означает «проклятие размерности»?
- 3. Чем Feature Selection отличается от Feature Extraction?
- 4. Какова основная идея PCA?
- 5. Для чего нужна стандартизация перед PCA?
- 6. Чем LDA отличается от PCA?
- 7. Для чего применяется t-SNE и что означает Perplexity?
- 8. Чем случайная выборка отличается от стратифицированной?
- 9. Что такое агрегация и сжатие данных?
- 10. Какие ошибки возникают при снижении размерности?

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

- Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques. 3rd ed., 2011.
- James G. et al. An Introduction to Statistical Learning. 2023.
- Bishop C. M. Pattern Recognition and Machine Learning. Springer, 2006.
- Jolliffe I. T., Cadima J. Principal Component Analysis: A Review and Recent Developments. 2016.
- Scikit-learn: Dimensionality Reduction — scikit-learn.org/stable/modules/unsupervised_reduction.html
- Scikit-learn: PCA, LDA и t-SNE — официальная документация scikit-learn.