非监督学习：降维 (Dimensionality Reduction)

汪小圈

2025-04-21

降维技术简介

- 在处理现实世界的数据时，我们经常会遇到特征维度非常高的情况（即有很多列）。
- 高维度数据不仅会增加模型的计算复杂度、延长训练时间，还可能引入噪声。
- **维度灾难 (Curse of Dimensionality)**：随着维度增加，数据变得稀疏，模型难以学习有效模式，性能下降。
- **降维 (Dimensionality Reduction)**：将高维数据转换为低维表示的过程，同时保留大部分有用信息。

为什么需要降维?

- **降低计算复杂度:** 特征越少，模型训练和预测所需的时间和内存就越少。
- **缓解维度灾难:** 在高维空间中，数据点变得稀疏，距离度量失去意义，模型更难找到有效的模式。
- **去除冗余和噪声:** 并非所有特征都是有用的，有些特征可能高度相关（冗余），有些可能是噪声。
- **提高模型可解释性:** 使用更少的关键特征更容易理解模型的决策过程。
- **数据可视化:** 将高维数据降到 2 维或 3 维，方便我们进行可视化探索。

降维的分类

降维技术主要可以分为两大类：

1. **线性降维方法**：假设数据位于线性子空间中
    - 主成分分析 (PCA)
    - 线性判别分析 (LDA)，监督式
2. **非线性降维方法**：处理位于非线性流形上的数据
    - t-分布随机邻域嵌入 (t-SNE)
    - UMAP (Uniform Manifold Approximation and Projection)
    - 核主成分分析 (Kernel PCA)

主成分分析 (PCA) 概述

PCA 是一种非常流行的**无监督线性降维**技术，属于**特征提取 (Feature Extraction)** 的范畴：

- 不是简单地选择一部分原始特征，而是将原始特征**线性组合**成一组新的、不相关的**主成分**。
- 这些主成分能最大程度地保留原始数据的**方差 (Variance)**。
- **第一主成分:** 数据投影后方差最大的那个方向。
- **第二主成分:** 与第一主成分正交，并且是剩余方差最大的方向。
- **以此类推**

PCA 的数学原理

从线性代数角度，PCA 可以通过以下步骤实现：

1. **数据中心化**：将每个特征减去其均值，使得每个特征的均值为 0
2. **计算协方差矩阵**：$\Sigma = \frac{1}{n-1}X^TX$，其中 X 是中心化后的数据矩阵
3. **计算协方差矩阵的特征值和特征向量**：求解 $\Sigma v = \lambda v$
4. **特征向量排序**：根据特征值大小降序排列特征向量
5. **选择前 k 个特征向量**：构建投影矩阵 W
6. **数据投影**：$Z = XW$，得到降维后的数据

PCA 的几何解释

从几何角度看，PCA 寻找的是数据中的主要变化方向：

- 想象一个三维空间中的扁平椭球体数据云：
  - 第一主成分是椭球体最长的轴
  - 第二主成分是次长的轴
  - 第三主成分是最短的轴
- 通过保留变化最大的方向，PCA 能够用较少的维度捕捉数据的主要结构。
- 主成分是相互正交（垂直）的，形成一个新的坐标系。

方差解释率与主成分选择

方差解释率 (Explained Variance Ratio):

- 衡量每个主成分能够解释原始数据**方差的比例**。
- 第一个主成分解释的方差比例最高，第二个次之，以此类推。
- 所有主成分解释的方差比例之和为 1 (或 100%)。

如何选择主成分数量 (n_components):

- **累积方差解释率:** 保留能够解释 95% 或 99% 方差的主成分。
- **肘部法则:** 绘制主成分数量与累积方差解释率的关系图，找到曲线拐点。
- **可视化需求:** 如果是为了可视化，通常选择 2 或 3 个主成分。
- **作为超参数:** 通过交叉验证来选择最佳值。

PCA 实现与特征缩放

警告

特征缩放的重要性
PCA 对特征的尺度非常敏感。在应用 PCA 之前，**必须对数据进行特征缩放** (通常使用 `StandardScaler`)。

使用 Scikit-learn 实现 PCA:

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

# 特征缩放
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# 应用 PCA
```

主成分的解释

了解主成分的物理意义，对理解数据结构非常重要：

- 每个主成分是原始特征的线性组合。
- 可以检查主成分的系数（即特征向量）来理解原始特征的贡献：python # 查看主成分系数 `print(pca.components_)`
- 对于图像数据，可以将主成分形象化：
  - 将主成分系数重塑为原始图像形状
  - 可视化为”特征脸”或”特征数字”等

PCA 应用与局限性

应用:

- **数据压缩:** 用更少的维度存储数据，减少存储空间和计算时间。
- **噪声去除:** 保留方差较大的主成分通常能过滤掉部分噪声。
- **可视化:** 将高维数据降到 2D 或 3D 进行可视化。
- **作为预处理步骤:** 降维后可输入到其他机器学习模型中。

局限性:

- **线性假设:** PCA 假设数据的主要结构是线性的，对于高度非线性的数据效果可能不佳。
- **可解释性差:** 主成分是原始特征的线性组合，其物理意义不如原始特征直观。
- **对特征缩放敏感:** 必须进行特征缩放。

t-SNE 简介

t-SNE (t-Distributed Stochastic Neighbor Embedding) 是一种非常流行的非线性降维方法：

- 特别适合数据可视化。
- 尝试在低维空间中保留高维空间中点的局部结构。
- 相似的点在降维后仍然靠近，不相似的点保持分离。
- 比 PCA 更好地保留局部结构，适合发现聚类。

t-SNE 的核心原理

t-SNE 与 PCA 的根本区别在于其目标函数:

1. **高维相似度计算**:在原始高维空间中,使用高斯分布计算点对之间的条件概率作为相似度。
2. **低维映射**:在低维空间中,使用 t 分布(而非高斯分布)计算点对之间的相似度。
3. **优化目标**:最小化高维空间和低维空间中相似度分布的 KL 散度。

perplexity 参数:

- 困惑度控制考虑每个点的局部邻域大小。
- 可理解为"有效邻居数量",通常在 5-50 之间。
- 关键超参数,需要尝试不同值。

t-SNE 的特点

- **使用 t 分布的原因**：在低维空间使用 t 分布（重尾分布）而非高斯分布，可以缓解"拥挤问题"（高维空间中适度远距离的点在低维投影中过于靠近）。
- **随机性**：结果依赖于随机初始化，每次运行可能得到不同结果。
- **注重局部结构**：t-SNE 特别擅长保留局部结构，但可能扭曲全局关系。
- **计算开销**：比 PCA 计算密集，不适合非常大的数据集。
- **参数敏感**：perplexity、迭代次数等参数需要调整。

UMAP 简介

UMAP (Uniform Manifold Approximation and Projection) 是一种较新的非线性降维技术：

- 基于黎曼几何和代数拓扑学理论。
- 在保持数据全局结构的同时，维持局部结构。
- 比 t-SNE 更快，且能更好地保留全局结构。
- 支持监督、半监督和无监督学习。

UMAP 的原理与特点

理论基础：

- **黎曼几何和流形理论**：假设高维数据位于低维流形上。
- **代数拓扑**：使用简化拓扑表示数据。

UMAP 的关键参数：

- **n_neighbors**：控制局部邻域大小，类似于 t-SNE 的 perplexity。
- **min_dist**：控制点的紧密程度，值越小，点越聚集。

UMAP 与 t-SNE 的关键区别：

- **全局结构**：UMAP 通常更好地保留全局结构。
- **计算效率**：UMAP 比 t-SNE 更快，尤其是对大型数据集。
- **新数据处理**：UMAP 支持新数据点的 transform，t-SNE 不支持。

降维方法比较

方法	数据规模	局部结构保持	全局结构保持	计算速度	可视化效果	新数据处理
PCA	任何规模	一般	好	非常快	一般	支持
t-SNE	中小规模	非常好	一般	慢	优秀	不支持
UMAP	各种规模	非常好	好	快	优秀	支持

如何选择合适的降维方法?

选择降维方法时应考虑以下因素:

1. **数据规模**: 大数据集可能更适合 PCA 或 UMAP, 而不是计算密集的 t-SNE
2. **任务目标**:
   - 可视化: t-SNE 或 UMAP 通常效果更好
   - 降噪: PCA
   - 分类预处理: LDA 或 PCA
3. **数据结构**:
   - 线性结构: PCA 或 LDA
   - 非线性流形: t-SNE、UMAP 或其他流形学习方法
4. **可解释性需求**: PCA 的主成分有明确的数学解释, 而非线性方法通常解释性较弱
5. **计算资源**: PCA 快速且高效, 非线性方法计算密集

降维实践建议

- **总是从 PCA 开始**：先尝试简单的线性方法，再逐步尝试复杂的非线性方法
- **特征缩放非常重要**：大多数降维方法对特征尺度敏感
- **可视化降维效果**：通过可视化了解数据的内在结构
- **调整参数**：每种方法都有关键参数需要调整（如 t-SNE 的 perplexity，UMAP 的 n_neighbors）
- **结合领域知识**：利用对数据的领域理解来评估降维结果
- **对比多种方法**：不同方法揭示数据的不同方面，综合考虑多种降维结果

总结

- 降维是处理高维数据的重要技术，能够减少计算复杂度、缓解维度灾难、去除冗余和噪声。
- **线性降维方法**（如 PCA）假设数据位于线性子空间，计算高效但对非线性结构效果有限。
- **非线性降维方法**（如 t-SNE 和 UMAP）能捕捉复杂的非线性关系，特别适合数据可视化，但计算更复杂。
- 选择合适的降维方法需考虑数据规模、结构特性、任务目标和计算资源。
- 降维是探索性数据分析、特征工程和机器学习流程中的重要环节，掌握这些技术将帮助你更有效地处理高维数据。