数据降维及可视化

文章对PCA算法的基本原理进行了解释说明，并采用了PCA算法和t-SNE算法对部分GTSRB数据集样本进行数据降维，并绘制了降维后的数据点空间分布图像，同时对PCA与t-SNE算法进行对比分析。此外还对数据集中随机一张图片，进行不同精度条件下的降维，并展示其对比重构效果。

PCA算法是一种常用的降维技术，其主要思想是将高维数据映射到低维空间中，以便于数据处理和可视化。其原理可以从不同的角度来解释，以下是三种常见的解释：

PCA算法的基本原理如下：

1. 最小化重构误差原理：在保持数据的低维表示的前提下，最小化数据在原始空间中的重构误差。
2. 最大化表示方差原理：在保持数据的低维表示的前提下，最大化数据在低维子空间中的方差，从而保留尽可能多的信息。
3. 概率视角原理：将数据看作是从一个高维空间中的概率分布中采样得到的，通过最大化数据的似然函数来找到最佳的低维表示。

1. 最小化重构误差原理

PCA的第一种解释是最小化重构误差原理，即将高维数据映射到低维空间中，使得映射后的数据在重构回原始高维空间时误差最小。这个误差通常用重构误差或平方重构误差来度量，即原始数据与其在低维空间中的投影之间的距离平方和。

其主要思想是将数据投影到一个低维的子空间，使得在这个子空间中投影误差最小。具体来说，PCA通过对原始数据的协方差矩阵进行特征值分解，得到一组正交的基，这些基可以被用来表示原始数据的主要方向。将数据投影到这些基所构成的低维空间中，就可以得到一个新的低维表示，它最小化了重构误差，即从原始数据到低维表示再回到原始数据时的误差。

2. 最大化表示方差原理

PCA的第二种解释是最大化表示方差原理，即在低维空间中找到一个方向，使得在该方向上投影的数据方差最大。这个方向被称为第一主成分，它是数据方差最大的方向。我们可以将数据投影到这个方向上，从而得到一个一维的低维表示。接下来，我们可以找到第二主成分，即在与第一主成分正交的方向上投影的数据方差最大的方向。我们可以将数据投影到这个方向上，从而得到一个二维的低维表示。这个过程可以继续进行下去，直到我们得到所需的低维表示。

最大化表示方差原理的基本思想是，保留数据中最重要的方向，即方差最大的方向，以尽可能多地保留数据的信息。

3. 概率视角原理

PCA的第三种解释是概率视角原理，即将数据看作是由一个高维的潜在变量和一个高斯噪声组成的模型。PCA算法的目标是找到一个低维的表示，使得在这个低维表示下，潜在变量和噪声可以被分离开来。具体来说，我们可以假设潜在变量是低维的，且它们之间是独立的，而噪声是高斯分布的。通过对数据的协方差矩阵进行特征值分解，我们可以得到一组正交基，这些基可以被用来表示原始数据的主要方向，同时也对应着潜在变量的方向。因此，通过投影数据到这些基所构成的低维空间中，我们可以得到一个低维的表示，使得潜在变量可以被分离出来。

从概率视角看，PCA算法的优化目标是最大化数据的似然函数。具体来说，我们可以假设数据是由潜在变量和噪声组成的高斯混合模型，然后通过最大化数据的对数似然函数，即最大化观察到的数据在模型中的概率，来找到最佳的低维表示。在实际应用中，我们通常使用EM算法来估计模型参数，并得到最佳的低维表示。

总的来说，PCA算法是一种非常常用的降维技术，它可以从不同的角度解释和理解。无论是最小化重构误差、最大化表示方差还是概率视角，都可以帮助我们更好地理解PCA算法的本质和优化目标。

分别将原始数据降维至2维、3维情形，绘制降维后的数据点在空间中的分布

1.原始数据的选择

使用t-SNE方法对GTSRB数据集中的10个类别进行降维可视化，与PCA进行对比分析。选择了与之前相同的00000，00001，00002，00003，00004，00005，00006，00007，00008，00009这10个类别，每个类别选取200张图片进行降维处理，保留前两个和前三个主成分，得到了2维和3维的降维结果。下图展示了t-SNE和PCA在2维和3维下的降维可视化结果。

选择GTSRB数据集中训练集部分前10个类别，每个类别中选择200个图像样本。

selected\_classes = ["00000", "00001", "00002", "00003", "00004", "00005", "00006", "00007", "00008", "00009"]

num\_images\_per\_class = 200

2.提取图片特征向量

3.使用PCA、t-SNE算法将特征向量降维至2维和3维

*# 定义PCA模型并进行拟合*

pca\_model = PCA(n\_components=n\_components)

img\_pca = pca\_model.fit\_transform(img\_tensor.cpu().numpy())

*# 定义t-SNE模型并进行拟合*

tsne\_model = TSNE(n\_components=n\_components,

                  perplexity=30, learning\_rate=200,

                  n\_iter=1000, random\_state=42)

img\_tsne = tsne\_model.fit\_transform(img\_tensor.cpu().numpy())

其中t-SNE模型的超参数设置了一些常见值。

4.数据点空间分布可视化

在这个例子中，我们首先将一张图像转换为向量形式，并使用不同的精度进行PCA和t-SNE降维。然后，我们将降维后的图像重新构建回原始形状，并将其可视化展示出来。通过对比不同精度下的降维结果，我们可以发现随着精度的提高，降维后的图像的质量也越来越高。

通过对比分析，可以得出以下结论：

1.t-SNE在降维后的数据点分布更加分散，不同类别之间的区分度更高，而PCA在降维后的数据点分布更加集中，不同类别之间的区分度较低。

2.t-SNE在降维时更加关注数据点之间的相似度，更加擅长发现高维空间中的局部结构，而PCA在降维时更加关注数据点的全局结构，更加擅长发现高维空间中的主要方向。

3.t-SNE在降维时计算复杂度较高，需要耗费较长的时间和计算资源，而PCA计算速度较快，适用于大规模数据的降维处理。

综上所述，t-SNE和PCA都是常用的降维方法，具有各自的优缺点和适用场景。在实际应用中，需要根据具体的数据集和任务需求来选择合适的降维方法。

第二次作业是关于交通信号灯数据的聚类。实验数据采用GTSRB的数据，请从中选择至少5个类别，每个类别至少20张图片，以完成实验报告。

具体来说，此次报告需要完成以下两个视角的实验分析，

- 将图片处理成同样大小（比如，HxWxC），一张图片看作一个数据点，直接进行聚类（高维聚类，可考虑降维/特征提取之后再聚类）

- 将一张图片看作RGB空间中的一个点集，对该点集进行聚类（基于聚类的图像分割）

备注

- 可选的聚类方法，如K-means及其变体，Mean Shift，Spectral Clustering, Subspace Clustering, Density-based Clustering, etc.

- 至少选择两种不同的聚类方法，简述原理（不超过半页）

- 展示聚类效果（适量）并进行定量和定性对比分析

- 独立完成报告，不超过三页，请勿抄袭

- 建议使用tex排版，注意参考文献引用

- 报告以pdf格式发送至changsheng\_zhou@gzhu.edu.cn

- 截至日期2023.6.6