STAT 5244 – Unsupervised Learning

Homework 1

Name: Chuyang Su UNI: cs4570

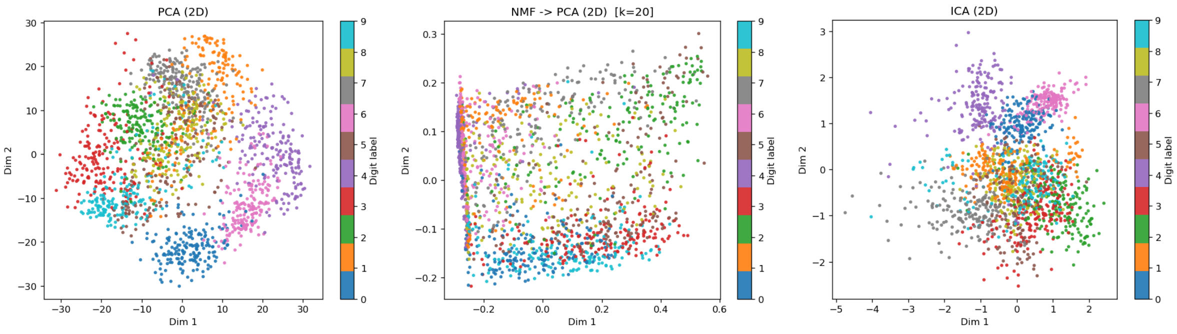
1 Dimension Reduction on Digits Data.

1.1 Apply linear dimension reduction techniques including PCA, NMF, and ICA.

In this experiment, I applied three linear dimension reduction methodsand compared their performance on the `scikit-learn` *Digits* dataset ($n = 1797$, $p = 64$).

Each method projects the data into a two-dimensional latent space, on which I visualized the results and quantitatively evaluated their ability to separate the ten digit classes.

The results of this experiment are summarized below.



(a) PCA (b) NMF→PCA (c) ICA

Figure 1: 2D embeddings of the digits data using PCA, NMF→PCA, and ICA. Colors denote true digit labels.

Method	**ARI**	**NMI**	**Silhouette**
PCA	**0.3614**	**0.5190**	0.3993
NMF → PCA	0.1588	0.2961	**0.4080**
ICA	0.3175	0.4567	0.3673

Table 1: Quantitative comparison of linear dimension reduction methods. The best scores for each metric are bolded.

PCA. For PCA, I retained the first two principal components and projected the samples into the 2D subspace they span, which preserves the primary directions of variance. As for hyperparameters, PCA has very few tunable parameters—the main one being the number of principal components. For ease of visualization, I set the number of components to 2 (PC=2) in this experiment. Figure 1a shows that the data are well dispersed, and digits such as 0, 3, 4, 6, and 9 form clearly separated clusters. Quantitatively, PCA achieved an ARI of 0.3614, an NMI of 0.5190, and a Silhouette score of 0.3993. Except for the Silhouette score, PCA obtained the highest values among the three methods. This indicates that PCA effectively separates the digits and maintains a high level of consistency with the true labels. Although its Silhouette value (approximately 0.4) is not the highest, it still suggests reasonably compact and well-separated clusters. This minor difference can be attributed to slight overlaps between neighboring clusters in the 2D embedding, even though the overall structure aligns well with the ground-truth classes.

In addition to the 2D embedding, I plotted the PCA scree plot and a bar chart of the top ten principal components' explained variance, as shown in Figure 2. Both plots reveal that the first three components contribute substantially more variance than the rest, with the third component explaining slightly less variance than the first two but significantly more than the fourth. This supports the observation that the 2D projection loses some discriminative information, which explains why the best ARI achieved by PCA remains moderate (0.3614).

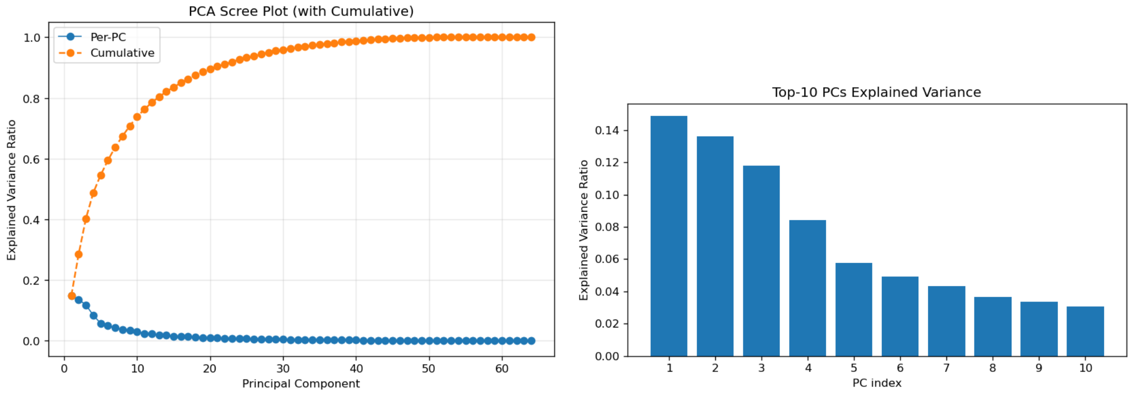


Figure 2: (Left) Scree plot showing the variance explained by each principal component; (Right) bar chart of the top-10 PCs' explained variance ratios.

Furthermore, I visualized the top ten PCA component images (Figure 3), which illustrate the principal modes of variation across digits.

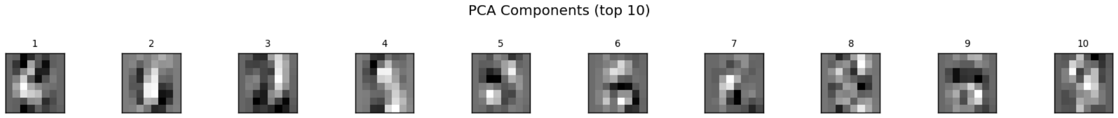


Figure 3: Top ten PCA components visualized as 8×8 basis images.

NMF→PCA. For the NMF method, I first determined the optimal number of components by minimizing the reconstruction error, which yielded $k = 20$. This means the data were decomposed into 20 non-negative basis vectors. The resulting coefficient matrix W was then compressed into a 2D embedding space using PCA for visualization and comparison purposes, and the final result is shown in Figure 1(b).

As observed in the plot, NMF fails to clearly separate the digits in the 2D space. This is consistent with the quantitative results, where NMF achieved the lowest ARI (0.1588) and NMI (0.2961) among all methods. These findings indicate that the NMF representation captures local, part-based features rather than global discriminative structures, and that the subsequent PCA compression may distort these part-based patterns, reducing the overall interpretability.

On the other hand, NMF obtained the highest Silhouette score (0.4080), slightly higher than PCA. The embedding shows that nearly all samples are concentrated in the positive subspace, with only a small portion extending below zero (no less than -0.2). Even after PCA compression, this non-negativity-induced structure is largely preserved, resulting in an asymmetric distribution that nevertheless exhibits slightly better cluster compactness than PCA. This suggests that some of the features extracted by NMF may be better suited to local clustering in this dataset. In particular, compared to PCA, NMF tends to focus on localized regions of variation rather than global variance directions, which likely contributes to the formation of tighter, more compact clusters.

Similar to the PCA case where excluding the third principal component led to a loss of explanatory power, it is plausible that the 2D projection of NMF also omits important structural information. Specifically, the visualization reveals that the first principal component successfully separates digit "4" along the left margin, while the second principal component distinguishes digits "9" (light blue) and "0" from the rest of the samples, forming a clear boundary near the bottom region of the plot. This indicates that the first PC primarily captures the unique pattern of the digit "4", whereas the second PC isolates the shapes shared by "0" and "9". A potential 3D embedding including an additional principal component might further clarify the remaining overlapping clusters visible in the upper-right portion of the 2D space.

Finally, I visualized the NMF basis components, as shown in Figure 4. These components represent localized stroke-like patterns, providing an interpretable decomposition of the digits into additive parts.

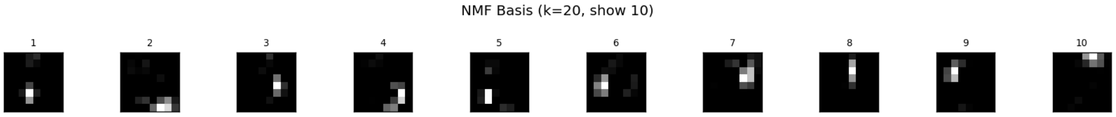


Figure 4: Top ten PCA components visualized as 8 × 8 basis images.

ICA. For the ICA method, I applied FastICA with two components after standardizing the data to ensure consistent feature scaling. As shown in Figure 1(c) and the summary table, ICA exhibits the most balanced overall performance among the three linear methods. Some clusters—notably digits 4, 6, and 0—are well separated and relatively compact, outperforming both PCA and NMF in local structure preservation. However, the remaining digits appear more mixed than in PCA, though less dispersed than in NMF. Visually, the embedding forms two major groups: one consisting primarily of digits 4, 6, and 0, and another containing the other seven digits.

Quantitatively, ICA's ARI (0.3175) and NMI (0.4567) values lie between those of PCA and NMF, while its Silhouette score (0.3673) is the lowest among the three. This aligns with the visual interpretation: ICA achieves moderate global separability but weaker overall cluster compactness.

Regarding hyperparameters, ICA provides limited tuning options. Here I set `components`=2, reducing the data to a 2D space, and standardized all features prior to decomposition. Standardization is a conventional preprocessing step for ICA, as its underlying assumption relies on statistical independence between components. Consequently, the resulting embedding appears nearly spherical and centered around the origin—a distribution consistent with ICA's model assumptions but inconsistent with the inherent non-spherical structure of handwritten digits. This mismatch explains the weaker performance of ICA in this task.

Finally, the ICA component images (Figure 5) reveal contrast-like and edge-detecting patterns that emphasize stroke boundaries, differing from the smoother, global variations captured by PCA and the localized parts extracted by NMF.

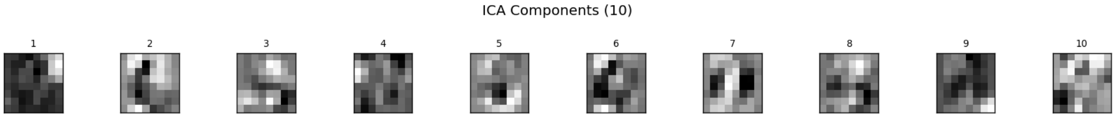


Figure 5: Top ten PCA components visualized as 8 × 8 basis images.

Overall Discussion. In summary, among the three linear dimension reduction techniques, PCA stands out as the most effective approach. It achieved the highest ARI (0.3614) and NMI (0.5190), and a Silhouette score (0.3993) close to the best. Visually, PCA produced the clearest and most interpretable clusters in the 2D embedding, separating several digits (such as 0, 3, 4, 6, and 9) distinctly. Given that PCA captures global variance directions, its performance would likely improve further in higher-dimensional embeddings, where additional components could better preserve discriminative variance.

A Appendix: Code Implementation