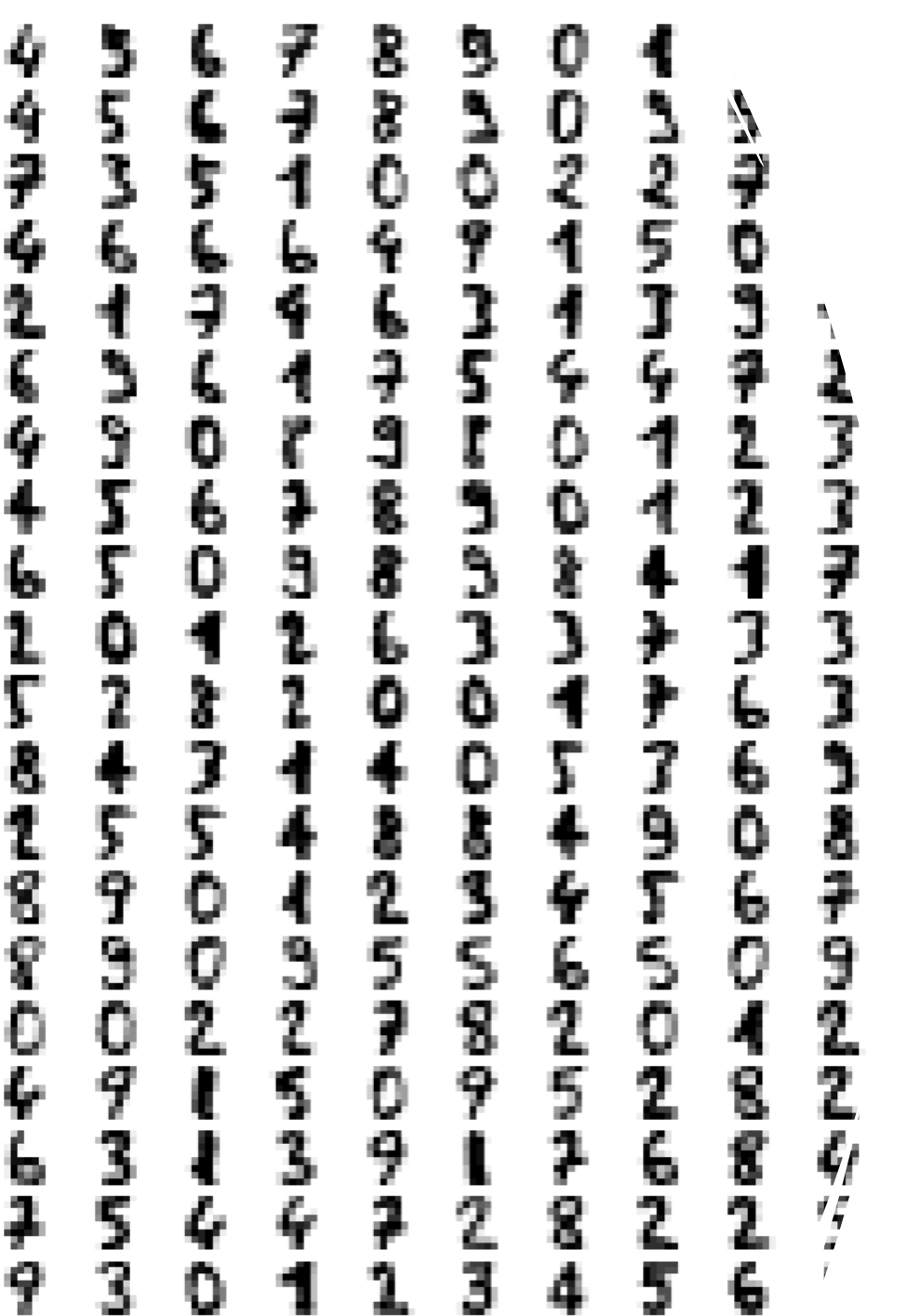




External Indices in Clustering



- Joseph Nelson Farrell
- DS 5230 Unsupervised Machine Learning
- Northeastern University
- Professor Steven Morin, PhD
- 03-30-24



Objective:

Perform dimensionality reduction on the digits dataset. Identify the best clustering algorithm using internal indices. Evaluate clustering algorithm performance using external indices.

MNIST Digits Data:

Image Size: 8 x 8

Data Shape:

- Rows: 1797 - digit observations
- Columns: 64 (float) – image pixel values

Preparation:

- To avoid repetition this presentation will assume the *midterm slide deck* has been reviewed.
- However, for a full explanation of the preparation and preprocessing please see the midterm slide deck.
- This slide deck begin with dimensionality reduction using **UMAP**.
- Nevertheless, some of the slides that follow will be repetitions from the *midterm slide deck*, please feel free to skip over these slides.

Grid Search Over UMAP Hypers:

- The grid search procedure was set up to iterate over the ranges of 4 UMAP hyperparameters
 - **min_dist:** [0.0, 0.01, 0.1, 0.25, 0.5, 0.75]
 - **n_neighbors:** [5, 10, 15]
 - **metric:** ['euclidean', 'manhattan']
 - **n_components:** [2, 3]
- The values of **n_components** were taken from:
 - *midterm (2)*
 - *assignment (3)*
- Note: In assignment 3 it was determined that the latent manifold of the digits dataset was 2. However, this value was replaced by 3 because **trustworthiness** does in fact increase beyond 2.

Model Selection

- At each iteration **cl.clustering** was called. This function implements **Kmeans** with the current embedding. Four intrinsic metrics were tracked.
 - **Silhouette score**
 - **Davies Bouldin score**
 - **Calinski Harabasz score**
 - **N_clusters found by the elbow method (intertia vs num_clusteres)**
- If all 4 of the metrics converged on the same number of clusters **Kmeans** was selected as a variable model for the current embedding.
- Alternatively, if **Silhouette score**, **Davies Bouldin score**, and **Calinski Harabasz** converged on the same number of clusters, **Kmeans** was also selected.
- If **Kmeans** was selected, the hyperparameters of both **Kmeans** and **UMAP** for the current embedding were stored as a row to be added to a **results** frame.

Model Selection conti...

- If the intrinsic metrics failed converge with **Kmeans**, **Dbscan** was executed.
- The hyperparameters **eps**, and **min_samples** were derived mathematically.
- A factor range around this derived **eps** was iterated over with **Dbscan** executed at each iteration.
- The model associated with the **max validity score** obtained during this iterative process was selected as the best model.
- The hyperparameters of both **DBscan** and **UMAP** for the current embedding were stored as row to be added to a **results** dataframe.

Results

- The process described in previous slides produces a 72 x 16 dataframe where each row contains a **UMAP** embedding, and the best clustering algorithm discovered for that embedding.

	true_number_of_clusters	algo	n_clusters_found	n_clusters_db_score_is_min	n_clusters_ch_score_is_max	n_clusters_silhouette_score_is_max	silhouette_score	hopkins_statistic
0	10	DBScan	18	NaN	NaN	NaN	NaN	0.092
1	10	DBScan	15	NaN	NaN	NaN	NaN	0.162
2	10	DBScan	17	NaN	NaN	NaN	NaN	0.101
3	10	DBScan	18	NaN	NaN	NaN	NaN	0.166
4	10	DBScan	16	NaN	NaN	NaN	NaN	0.102
...
67	10	DBScan	2	NaN	NaN	NaN	NaN	0.623
68	10	DBScan	3	NaN	NaN	NaN	NaN	0.333
69	10	DBScan	2	NaN	NaN	NaN	NaN	0.615
70	10	DBScan	3	NaN	NaN	NaN	NaN	0.389

Figure 1: Results Dataframe Abbreviated

Model Selection for Each Manifold Dimensionality

- The results frame displayed on the previous slide was filtered into 2 dataframes.
 - *n_components = 2*
 - *n_components = 3*
- The final selection of the latent manifold of the for each dataframe was determined by taking the maximum **silhouette score** or the **maximum validity index**.
- Next the external index ***adjusted rand score*** and the best ***contingency matrix*** were used to evaluate the clustering algorithms performance.
- These are external and thus require the ***true labels***

Adjusted Rand Score & Contingency Matrix

- The adjusted rand score is a measure of a clustering algorithm's performance.
 - It is bounded and produces a score of 1 for perfect clustering, and a score of close to 0 for random labeling.
 - It makes no assumptions about the structure of clusters.
- A contingency matrix is helpful in understanding where the clustering algorithm is making mistakes.
 - The rows of a contingency matrix are the true labels.
 - The columns are the predicted labels.
 - The diagonal elements are true labels that were predicted correctly.
 - By maximizing the trace, we can determine the best mapping of predicted labels to true labels.

Results: *Manifold Dimensionality = 3*

- On the right is a tabulation of latent manifold details for the *3-dimensional* manifold.
- This is the assignment 3 solution.
- This tabulation contains the **UMAP** and **DBscan** hyperparameter values for the best clustering algorithm.
- The external index **adjusted rand score** is also present.
- As can be observed, the *3-dimensional* latent manifold did not produce great clustering results.
- **Adjusted Rand Score = 0.106965**, where **1** is the best possible score and **0** is random random uniform.
- It can also be observed that **3** clusters were identified when **10** true labels exist.

	0
true_number_of_clusters	10
algo	DBScan
n_clusters_found	3
n_clusters_db_score_ls_min	NaN
n_clusters_ch_score_ls_max	NaN
n_clusters_silhouette_score_ls_max	NaN
silhouette_score	NaN
hopkins_statistic	0.385
umap_n_neighbors	15
umap_min_dist	0.25
umap_metric	euclidean
umap_n_components	3
trustworthiness	0.980747
eps	1.048814
dbscan_min_samples	7.0
validity_index	0.855885
cluster_labels	[0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, ...
adj_rand_score	0.106965

Figure 2: Transpose Manifold & Clustering Details Row

Results: *Manifold Dimensionality = 2*

- On the right is a tabulation of latent manifold details for the *2-dimensional* manifold.
- This is the midterm value.
- It contains the same information as the tabulation on the previous slide
- **Adjusted Rand Score = 0.840194.**
- This model performed considerably better than the best model on the *3-dimensional* manifold.
- As we can observe, this model found 13 clusters.
- This is still more than the true number of clusters.
- Next, we will evaluate where the clustering algorithm is making mistakes using the contingency matrixes.

	1
true_number_of_clusters	10
algo	DBScan
n_clusters_found	13
n_clusters_db_score_ls_min	NaN
n_clusters_ch_score_ls_max	NaN
n_clusters_silhouette_score_ls_max	NaN
silhouette_score	NaN
hopkins_statistic	0.112
umap_n_neighbors	15
umap_min_dist	0.0
umap_metric	manhattan
umap_n_components	2
trustworthiness	0.97445
eps	0.330699
dbscan_min_samples	5.0
validity_index	0.856719
cluster_labels	[0, 1, 2, 3, 4, 5, 6, 4, 6, 4, 7, 8, 9, 10, 9,...
adj_rand_score	0.840194

Figure 3: Transpose Manifold & Clustering Details Row

Finalized Tabulation of Results

	true_number_of_clusters	umap_n_components	umap_min_dist	umap_n_neighbors	umap_metric	trustworthiness	algo	n_clusters_found	validity_index	adj_rand_score
0	10	3	0.25	15	euclidean	0.980747	DBScan	3	0.855885	0.106955
1	10	2	0.0	15	manhattan	0.97445	DBScan	13	0.856719	0.840194

Figure 4: Finalized Tabulation of Manifold and Clustering Results

- Above is finalized tabulation of the manifold and clustering results for the latent manifold dimensionality from assignment 3 (**3**) and the midterm (**2**)
- As can be observed, the best clustering on the 3-dimensional manifold did not perform particularly well, achieving an **adjusted rand score** of only **0.106955**.
- The best clustering on the 2-dimensional manifold performed considerably better, achieving an **adjusted rand score** of **0.840194**.
- Next, we'll examine the best contingency matrices to gain insights into where the clustering algorithm is making mistakes.

Best Contingency Matrix 3D Manifold: Assignment 3

- To the right is best contingency matrix from the clustering performed on the 3D manifold.
- The rows represent the true labels.
- The columns represent the predicted labels.
- As previously noted, the performance is rather poor, identifying only **3** clusters when **10** true classes exist.
- Interestingly, while overall, the clustering algorithm did not perform well, it was able to identify 2 of the true classes and achieve one pure cluster (**0**) and one nearly pure cluster (**6**).
- Despite the poor performance, next, we'll examine some of the mislabeled images.

```
[[178  0  0]
 [ 0 182  0]
 [ 0 177  0]
 [ 0 183  0]
 [ 0 181  0]
 [ 0 181  1]
 [ 0  1 180]
 [ 0 179  0]
 [ 0 174  0]
 [ 0 179  1]]
```

Figure 5: Contingency Matrix 3D Manifold

Mislabeled Image Examples 3D Embedding

- Here can see a sample of true label 1 elements and their predicted label.
- Because a *mode* method was used to determine the true labels, the majority of mislabeled elements in this embedding received *predicted label 3*.
- This is a result of the limited number of clusters found (3) and mode of the cluster containing the majority of the elements, namely *mode = true label 3*.
- A similar pattern is observed for other true labels with this embedding.
- We'll display this next, with more true label examples.

Mislabeled Image Examples

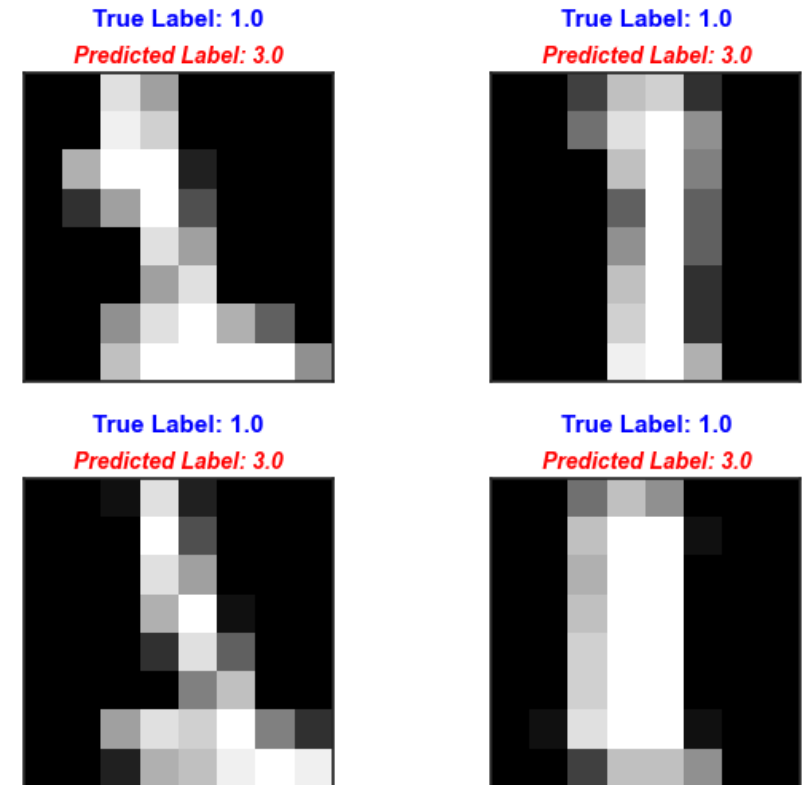


Figure 6: Examples of Mislabeled Element

Mislabeled Image Examples 3D Embedding

Mislabeled Image Examples

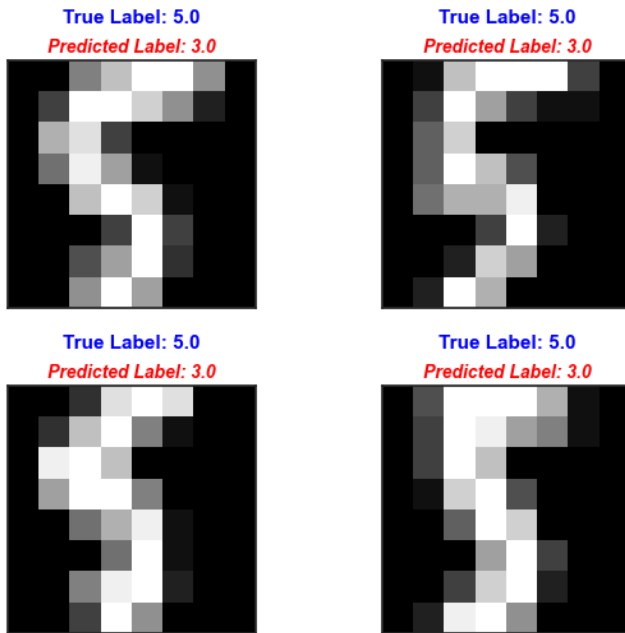


Figure 6: Examples of Mislabeled Elements

Mislabeled Image Examples

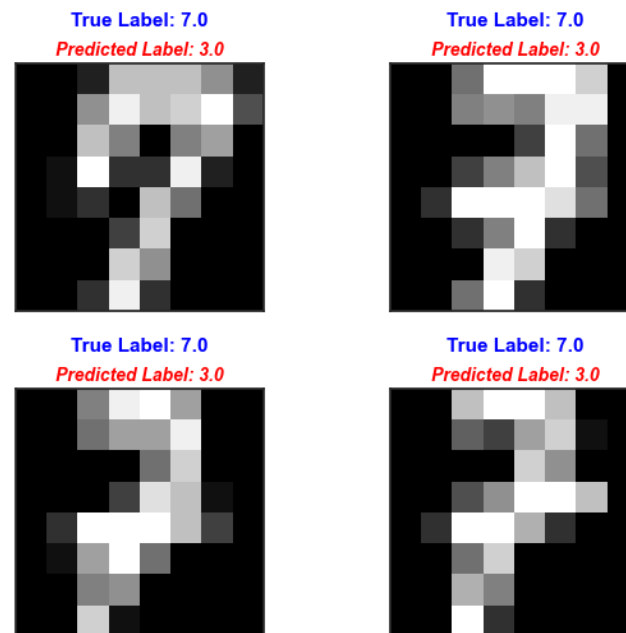


Figure 7: Examples of Mislabeled Elements

Mislabeled Image Examples

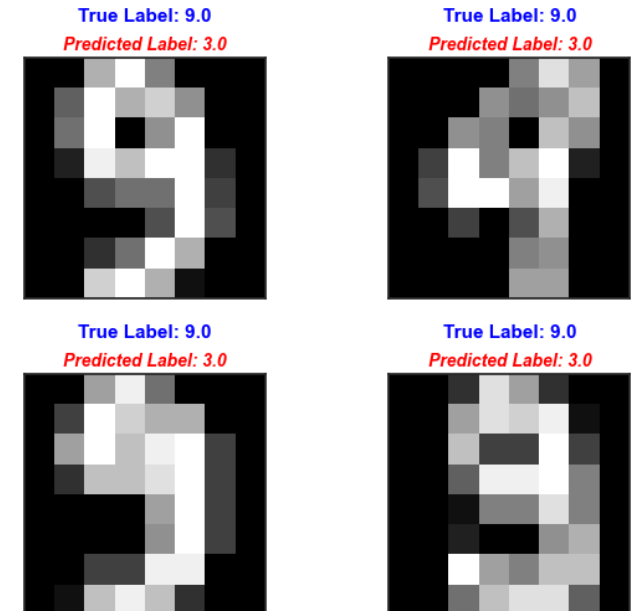
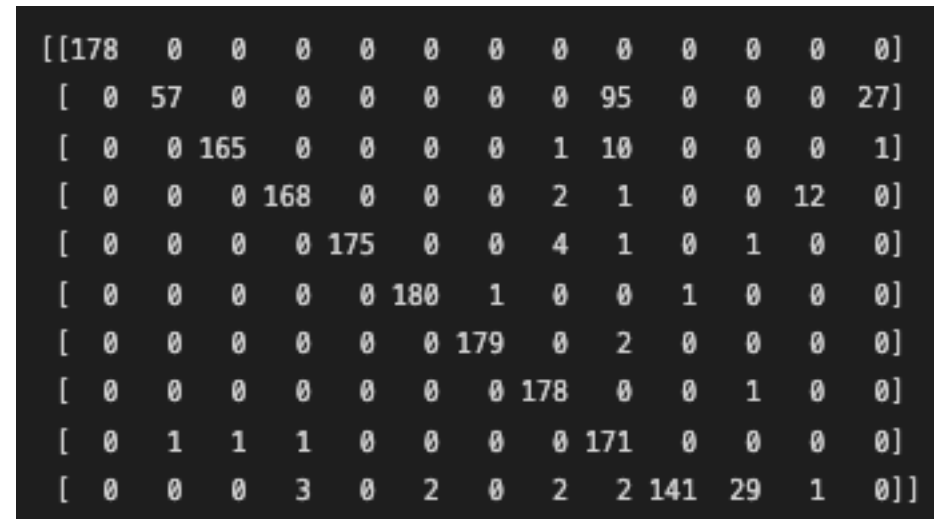


Figure 8: Examples of Mislabeled Elements

- Because of the poor performance of this clustering solution, we'll move to an examination of the 2D embedding.

Best Contingency Matrix 2D Manifold: Midterm

- To the right is best contingency matrix from the clustering performed on the 2D manifold.
- Clustering on the 2D manifold performed considerably better.
- Many of the true labels have a high degree of purity.
- True labels where the algorithm made most mistakes are:
 - **True Label: 1**
 - **True Label: 2**
 - **True Label: 9**
- The algorithm also identified 3 clusters that are not present in the true labels.
- Next, we'll examine some example images that the algorithm mislabeled.
- Because this is a 2D embedding we can also examine the clusters with respect to the embedding space.



[178	0	0	0	0	0	0	0	0	0	0	0	0]
[0	57	0	0	0	0	0	0	95	0	0	0	27]
[0	0	165	0	0	0	0	1	10	0	0	0	1]
[0	0	0	168	0	0	0	2	1	0	0	12	0]
[0	0	0	0	175	0	0	4	1	0	1	0	0]
[0	0	0	0	0	180	1	0	0	1	0	0	0]
[0	0	0	0	0	0	179	0	2	0	0	0	0]
[0	0	0	0	0	0	0	178	0	0	1	0	0]
[0	1	1	1	0	0	0	0	171	0	0	0	0]
[0	0	0	3	0	2	0	2	2	141	29	1	0]]

Figure 5: Contingency Matrix 2D Manifold

Mislabeled Image Examples 2D Manifold

Mislabeled Image Examples

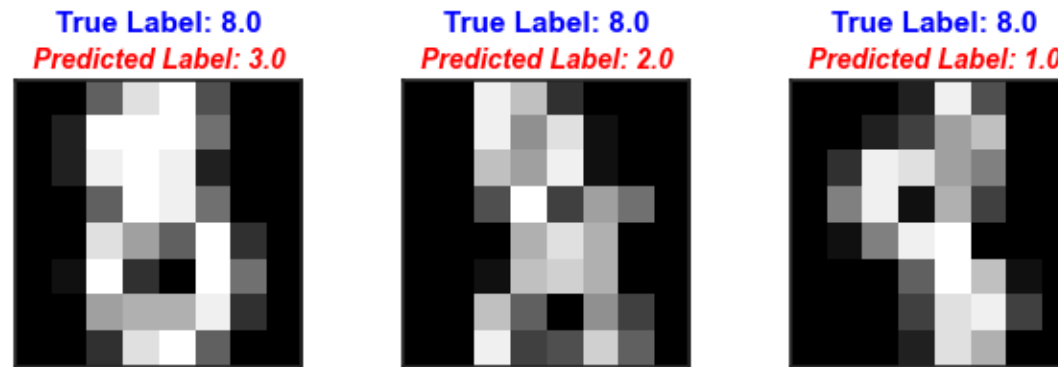


Figure 8: True Label 8 – Mislabeled Images

- Above are the 3 images with **true label 8** that were mislabeled.
- There does not appear to be pattern to the mislabeling.
- Next, we'll look at true label 4.

Mislabeled Image Examples 2D Manifold

- On the right is a sample of **true label 4** images that were mislabeled.
- These mislabeled images do display a pattern, they are all in the same mislabeled cluster, namely **7**.
- This is not true for all the mislabeled 4's, but interesting, nevertheless.
- Now we'll shift, and rather than further examining mislabeled elements by true label, we'll examine mislabeled elements by looking at predicted labels that are not present in the true label set.

Mislabeled Image Examples

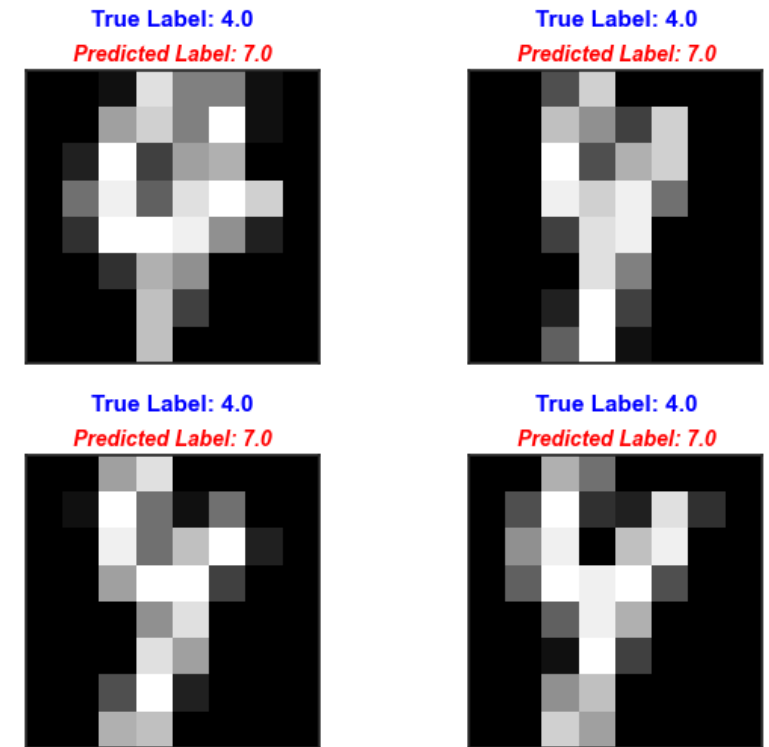
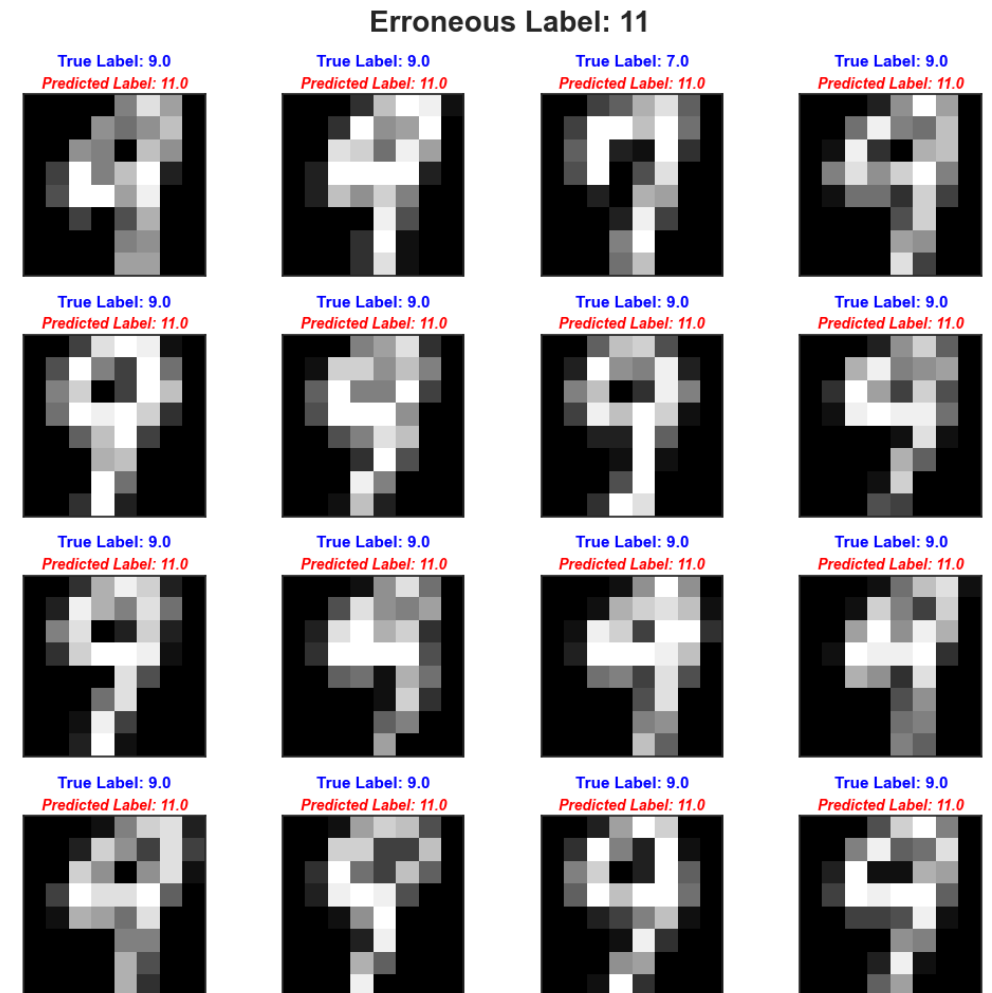


Figure 9: True Label 4 – Mislabeled Images

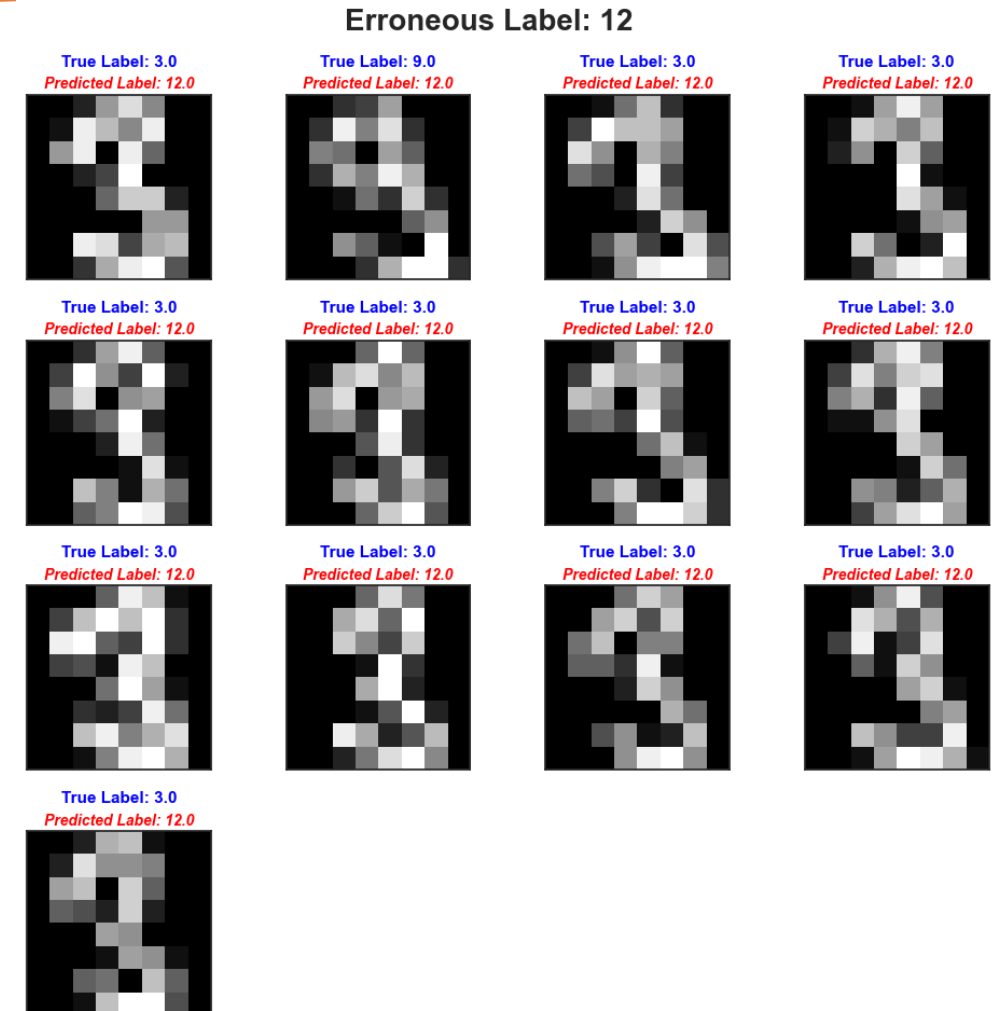
Mislabeled Image Examples 2D Manifold

- This is perhaps a more informative exploration.
- In this sample we can observe that many of the elements that were *labeled 11* are in fact a *9*.
- This information could potentially be used to make corrections to the algorithmic output.
- Next, we'll examine one more of the erroneous labels.



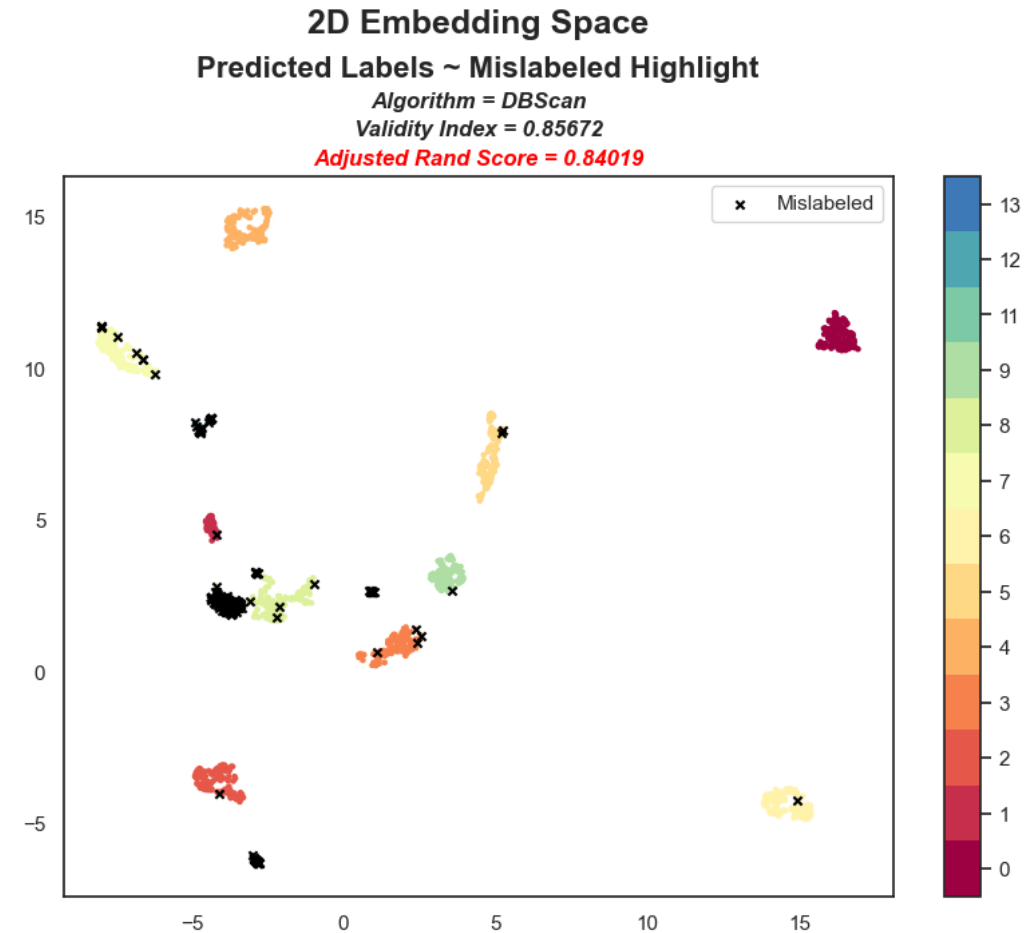
Mislabeled Image Examples 2D Manifold

- This example shows the entire cluster for erroneous **label 12**.
- Again, we can observe that many of the elements that received this label are of the same true label class, **3**.
- This information could potentially be used make corrections to the algorithmic output.
- Because this is a 2D embedding we can further explore cluster performance with respect to the embedding space.



Clustering Outcome with Respect to the Embedding Space

- Here we can see data points on the 2D embedding with their cluster labels.
- This graphic highlights all the data points that were mislabeled.
- What immediately jumps out is the large cluster of mislabeled data points around (-4, 1).
- Two smaller clusters of mislabeled data points are visible around:
 - (-2.5, -6)
 - (-4, 1.5)
- Many of the other mislabeled data points appear to be in proximity to true clusters.
- This information could potentially be used to improve data point labeling.



Summary

- Clustering performance on the 3D embedding was poor, only identifying 3 clusters
- Clustering performance on the 2D embedding faired better, identifying 13 clusters.
- Potential patterns identified with the mislabeled images could improve element labeling in the 2D embedding.

Acknowledgements.

- Morin, S. (2024). DS 5330 Class Materials
- Scikit-learn.org
- Umap-learn.readthedocs.io