Genomics and High Dimensional Data: Written Analysis

MITx 6.419x Data Analysis: Statistical Modeling and Computation in Applications

Ellick Hou - 3/24/21

table of contents

Problem 2: Larger unlabeled subset

Include your answers to all parts of Problem 2 in your written report.

Now we will work with the larger, unlabeled subset in `p2_unsupervised`. This dataset is has not been processed, so you should process using the same log transform as in Problem 1.

✦ Part 1: Visualization

✦ 1. Provide at least one visualization which clearly shows the existence of the three main brain cell types described by the scientist, and explain how it shows this. Your visualization should support the idea that cells from a different group (for example, excitatory vs inhibitory) can differ greatly.

➡ Ans:

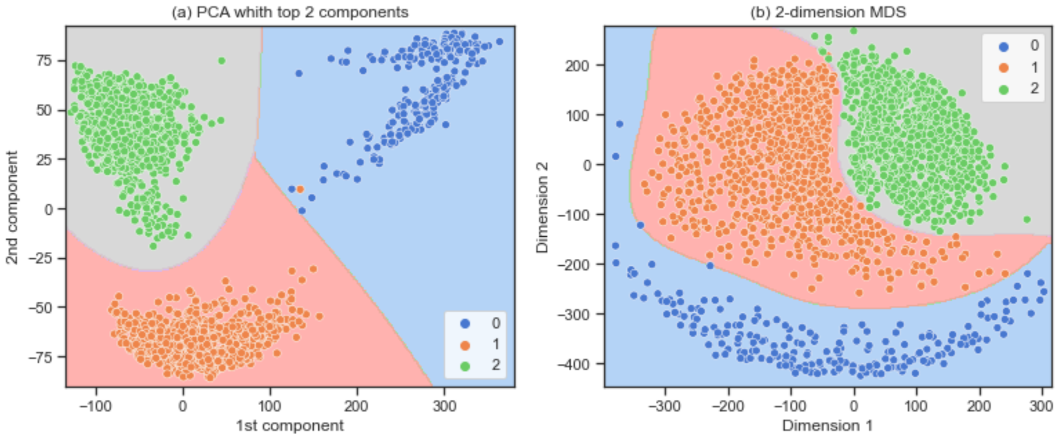

Figure 2.1.1: (a) Projection of ''p2_unsupervised" dataset in first and second PC subspace with decision boundary. **(b)** 2-dimensional MDS result of ''p2_unsupervised" dataset with decision boundary.

- Labels in Figure 2.1.1 are generated by given 3 clusters to Agglomerative Clustering algorithm with 'ward' linkage, then ran on ''p2_unsupervised" dataset projected onto top 509 PCs(60% cumulative explained variance). Cluster 0, 1, 2 are colored in blue, orange, green respectively.
- Figure 2.1.1 (a) shows the projected dataset on top-2 PC, Figure 2.1.1 (b) shows the 2-dimensional representation of dataset's dissimilarity matrix. Intuitively, top-2 PC seem to have a better performance in distinguishing the three clusters. To verify this intuition, these two transformed datasets were fed into two different support vector classifiers(SVC_pca and SVC_mds) with the pseudo labels mention above. Both

classifiers were been tune to the same error rate, 1/2169(sample size), with the lowest regularization parameter C, and be selected by stratified 5-folds cross-validation strategy. The resulting decision boundaries as shows in Figure 2.1.1 (a) and Figure 2.1.1 (b), SVC_pca and SVC_mds have C=0.4 and 10.6 respectively.

- In this question, I think the statement “differ greatly” could be answered by this two following assumptions.
  1. If the embedded datasets can differ cell groups greatly, the labels observed/found in high-dimensional space should be classified correctly in in the low-dimensional embedded datasets.
     - If we are doing in supervised fashion, this assumption will be easy to confirm. But the labels I use here are generated by an unsupervised algorithm, if a data point is ‘’misclassified” is this data point an outlier in high-dimensional space(trust the low-d classifier) or the 2-dimensional representation is too simple to present the relationship(trust the pseudo label)? This question become hard to answer, so I limited this problem into the next assumption.
  2. If the embedded datasets can differ cell groups greatly, classifiers should be easily separate data points, in low-dimensional representation, into the pseudo groups.
     - One way to confirm this assumption, we can look into the objective function of the classifiers, in this case is SVC.

$$\min \frac{1}{2}||w||^2 + C \sum Loss_{hinge}(w, x, y) \qquad (1)$$

     - Function (1) is a simple version of the objective function of SVC. Regularization parameter C controls the weight between l2 penalty and hinge loss, and also implies how much effort classifiers should take to force data points to be/beyond support vectors, i.e. the margin boundaries. Furthermore, to make the comparison between two datasets more fairly, I want to set a minimum error rate of each classifier should reach. In Figure 2.1.1 (a), intuitively, I think 1 misclassified data is a standard of a good classifier should make, so the error rate baseline of SVC_pca and SVC_mds here is set to 1/2169(sample size). The minimum C of SVC_pca and SVC_mds reach the error rate baseline is 0.4 and 10.6 respectively(round to 1 decimal place). And we can tell by look into the objective function, SVC_mds seems to be more struggle to reach the baseline. Therefore we can verify our intuition now, top-2 PCs projection does differ cell groups better than 2-dimensional MDS result. For more generally speaking, if we use C=1 as a standard to referee the low-dimensional data ‘’differ” groups well or not. top-2 PCs projection still plays well.

✦ 2. Provide at least one visualization which supports the claim that within each of the three types, there are numerous possible sub-types for a cell. In your visualization, highlight which of the three main types these sub-types belong to. Again, explain how your visualization supports the claim.

➡ Ans:

❖

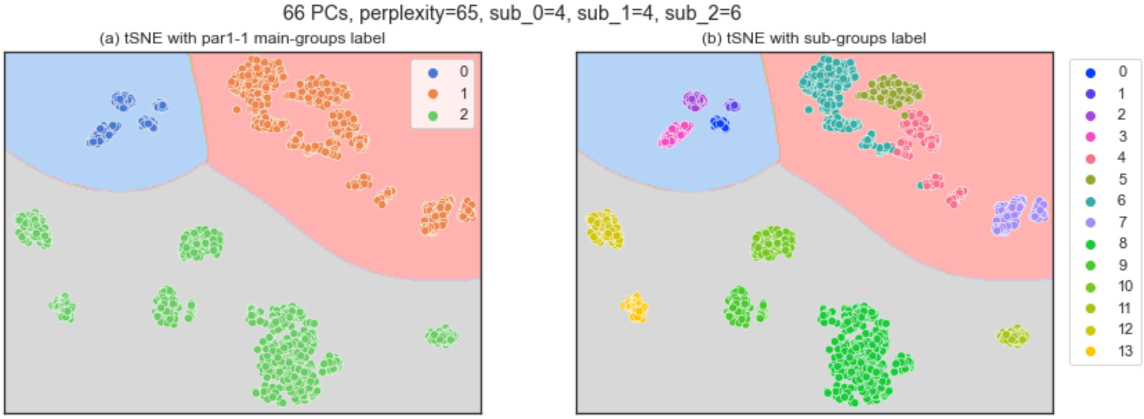

Figure 2.1.2: (a) Embedded dataset using tSNE with part1-1 main-groups label. **(b)** Embedded dataset using tSNE with sub-groups label.

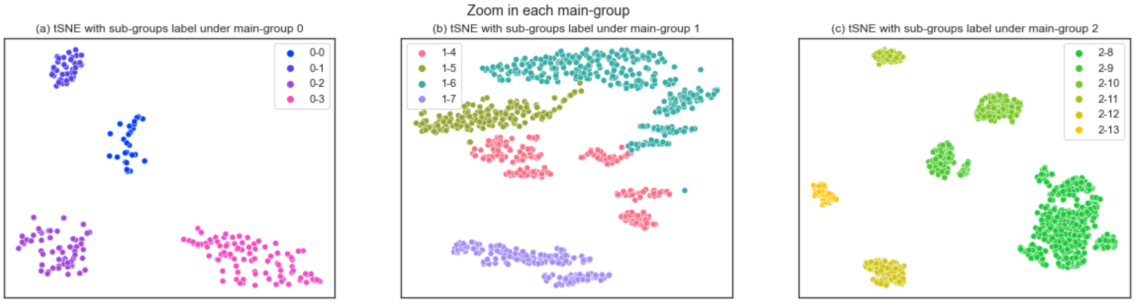

Figure 2.1.3: (a) Zoom in tSNE result on main-group 0. **(b)** Zoom in tSNE result on main-group 1. **(c)** Zoom in tSNE result on main-group 2.

❖ Labels in Figure 2.1.2 and Figure 2.1.3 are generated by given 1000 euclidean distance threshold to Agglomerative Clustering algorithm with 'ward' linkage, then ran on dataset projected onto top-66 PCs(35% cumulative explained variance).

- ❖ All t-SNE embedded samples with corresponding three main-groups labels shows in Figure 2.1.2 (a). t-SNE was applied on projected top-66 PCs dataset with perplexity = 65. The same t-SNE result stay in the same position and re-color with new sub-groups labels mention above in Figure 2.1.2 (b). The decision boundaries in Figure 2.1.2 is corresponding to the three main-groups.
- ❖ Sub-groups under three main-groups were plot in Figure 2.1.3. There are 4, 4, 6 numbers of sub-groups in main-group 0, 1, 2 respectively. Each sub-group separate well except 1-6 slightly touch 1-4 and 1-5 as shows in Figure 2.1.3(b)

✦ Part 2: Unsupervised Feature Selection

Now we attempt to find informative genes which can help us differentiate between cells, using only unlabeled data. A genomics researcher would use specialized, domain-specific tools to select these genes. We will instead take a general approach using logistic regression in conjunction with clustering. Briefly speaking, we will use the `p2_unsupervised` dataset to cluster the data. Treating those cluster labels as ground truth, we will fit a logistic regression model and use its coefficients to select features. Finally, to evaluate the quality of these features, we will fit another logistic regression model on the training set in `p2_evaluation`, and run it on the test set in the same folder.

✦ 1. Using your clustering method(s) of choice, find a suitable clustering for the cells. Support your choice of clustering with appropriate visualizations and/or numerical findings. Be sure to briefly explain how you chose the number of clusters.

➡ Ans:

(1) I chose 3 different numbers of PCs as the t-SNE input, {17, 35, 66} (correspond to 30%, 32.5%, 35% cumulative explained variance) to save the searching time. Then greed search the perplexity of t-SNE as the 2-dimensional representation data.

(2) Use different number of clusters in K-means on {17, 35, 66} PCs, use WCSS elbow plot and average silhouette score to decide the cluster searching range. [4, 6], [7, 13], [6, 8] for main-group 0, main-group 1, main-group 2 respectively. Find out 4 clusters mach the number of sub-groups under main-group 0, 6 mach the number of sub-groups under main-group 2, But cannot find number of sub-groups mach the 2-dimensional representation of main-group 1. Show as Figure 2.2.1, Figure 2.2.2.

(3) Adjust euclidean distance threshold of Agglomerative Clustering algorithm to match 4, 6 of sub-groups under main-group 0, main-group 2. Show as Figure 2.2.3.

(4) Find the corresponding sub-groups with the same euclidean distance threshold under main-group 1.

(5) Match the 2-d patterns with the labels mention above. Result as Figure 2.1.2 (b), 4, 4, 6 numbers of sub-groups in main-group 0, 1, 2 respectively.

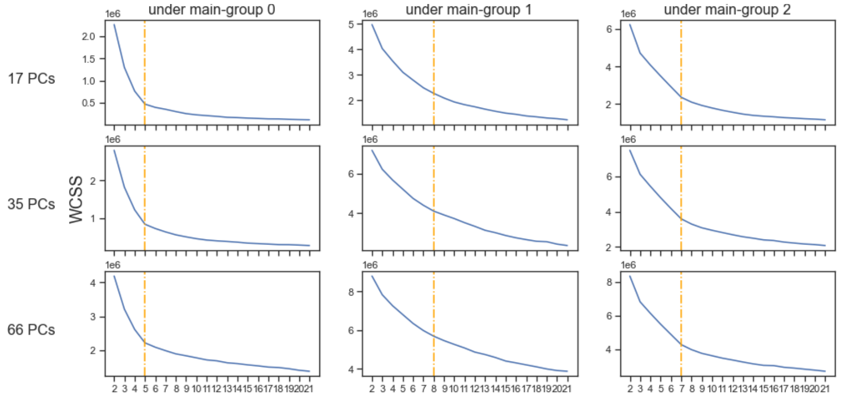

Figure 2.2.1: WCSS elbow plot with different number of cluster settings(column) and different PC projections(row).

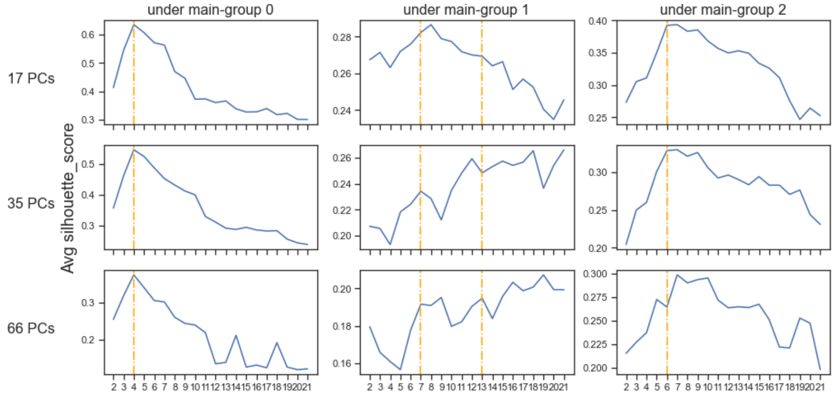

Figure 2.2.2: Average silhouette score plot with different number of cluster settings(column) and different PC projections(row).

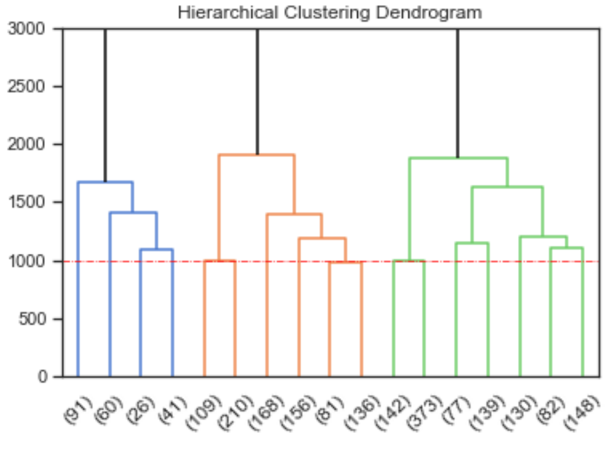

Figure 2.2.3: Hierarchical Clustering Dendrogram

✦ 2. We will now treat your cluster assignments as labels for supervised learning. Fit a logistic regression model to the original data (not principal components), with your clustering as the target labels. Since the data is high-dimensional, make sure to regularize your model using your choice of $\ell1$, $\ell2$, or elastic net, and separate the data into training and validation or use cross-validation to select your model. Report your choice of regularization parameter and validation performance.

➡ Ans:

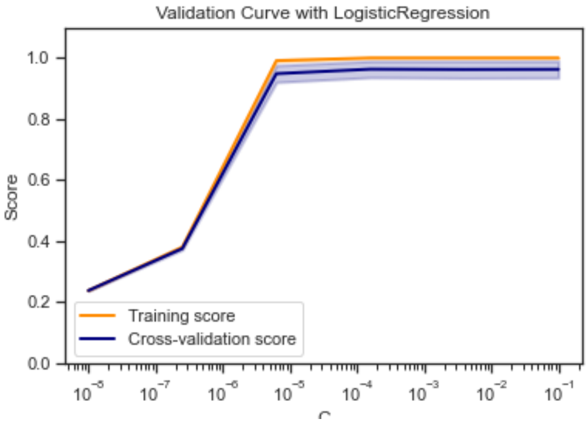

Figure 2.2.4: Validation Curve of log10(C) train on ''p2_unsupervised''

❖ Final model, regularization parameter C=6.30957344e-06, penalty='l2', select by stratified 5-folds cross-validation with
- Validation accuracy = [0.95391705, 0.96774194, 1, 0.98156682, 0.92147806]
    - Mean validation accuracy = 0.9649407732995605
- Training accuracy = 0.9944674965421854

✦ 3. Select the features with the top 100 corresponding coefficient values (since this is a multi-class model, you can rank the coefficients using the maximum absolute value over classes, or the sum of absolute values). Take the evaluation training data and use a subset of the genes, consisting of the features you selected. Train a logistic regression classifier on this training data, and evaluate its performance on the evaluation test data. Report your score. Compare with two baselines: random features (take a random selection of 100 genes), and high-variance features (take the 100 genes with highest variance). Compare the variances of the features you selected with the highest variance features by plotting a histogram of the variances of features selected by both methods.

➡ Ans:

- Model accuracy of each feature selection
  - Top 100 coefficient model, C=0.01128838
    - Validation accuracy= [0.958 0.944 0.958 0.953 0.944]
    - Training accuracy = 0.989
    - Test accuracy = 0.9296
  - Top 100 variance model, C=0.01128838
    - Validation accuracy= [0.953 0.958 0.953 0.949 0.949]
    - Training accuracy = 0.990
    - Test accuracy = 0.9386
  - Random features model, C=0.1
    - Validation accuracy = [0.35648148 0.38888889 0.46046512 0.44651163 0.43255814]
    - Training accuracy = 0.7
    - Test accuracy = 0.3925992779783393
- Top 100 coefficient model has a similar performance as Top 100 variance model, but Top 100 variance model has better Test accuracy. Random features model is worst than guessing the predictions.
- The top 100 coefficient selection captures the 19 highest variance features in dataset. When the true variance decrease, coefficient selection has different choices of features. It might be due to the pseudo labels were 14 classes but the real labels were 36 classes. Therefore the logistic regression model in Part 2.2 can not capture the most important feature in dataset. The lack of ground truth problem also responds in Top 100 variance model has a slightly better validation and test accuracy.

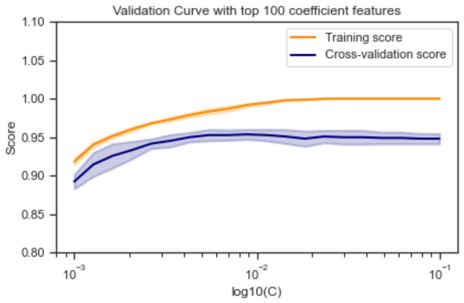

Figure 2.2.5: Validation Curve of log10(C) on top 100 coefficient features of ''p2_evaluation/X_train''.

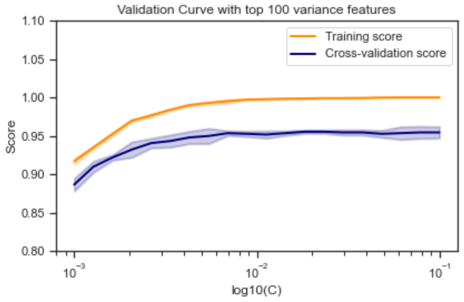

Figure 2.2.6: Validation Curve of log10(C) on top 100 variance features of ''p2_evaluation/X_train''.

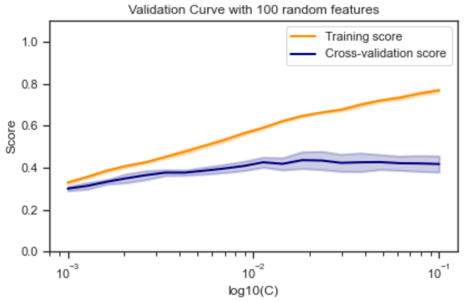

Figure 2.2.7: Validation Curve of log10(C) on 100 random features of ''p2_evaluation/X_train''.

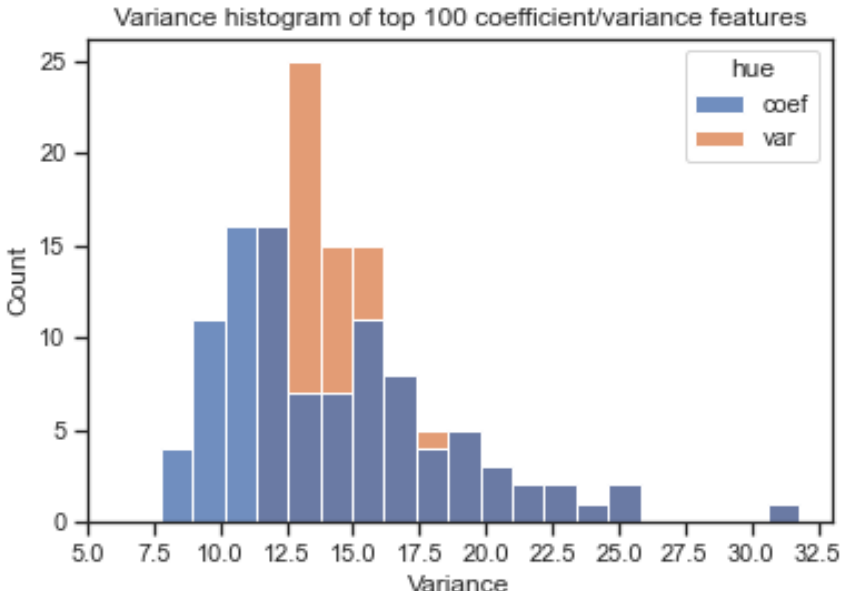

Figure 2.2.8: Variance histogram of top 100 coefficient/variance features.

Problem 3: Influence of Hyper-parameters

✦ Problem 3.1

(3 points) When we created the T-SNE plot in Problem 1, we ran T-SNE on the top 50 PC's of the data. But we could have easily chosen a different number of PC's to represent the data. Run T-SNE using 10, 50, 100, 250, and 500 PC's, and plot the resulting visualization for each. What do you observe as you increase the number of PC's used?

➡ Ans

- ❖ The number of outliers become more when the number of PCs increase. It's because T-SNE algorithm only try to maintain the small distance. When the outlier is far away from other point, the higher the dimension is the better chance T-SNE make them more far away. And the in group middle distance also has a better chance to become far distance, in this case, is especially obvious in main-group 0 as shows in Figure 3.1.1.

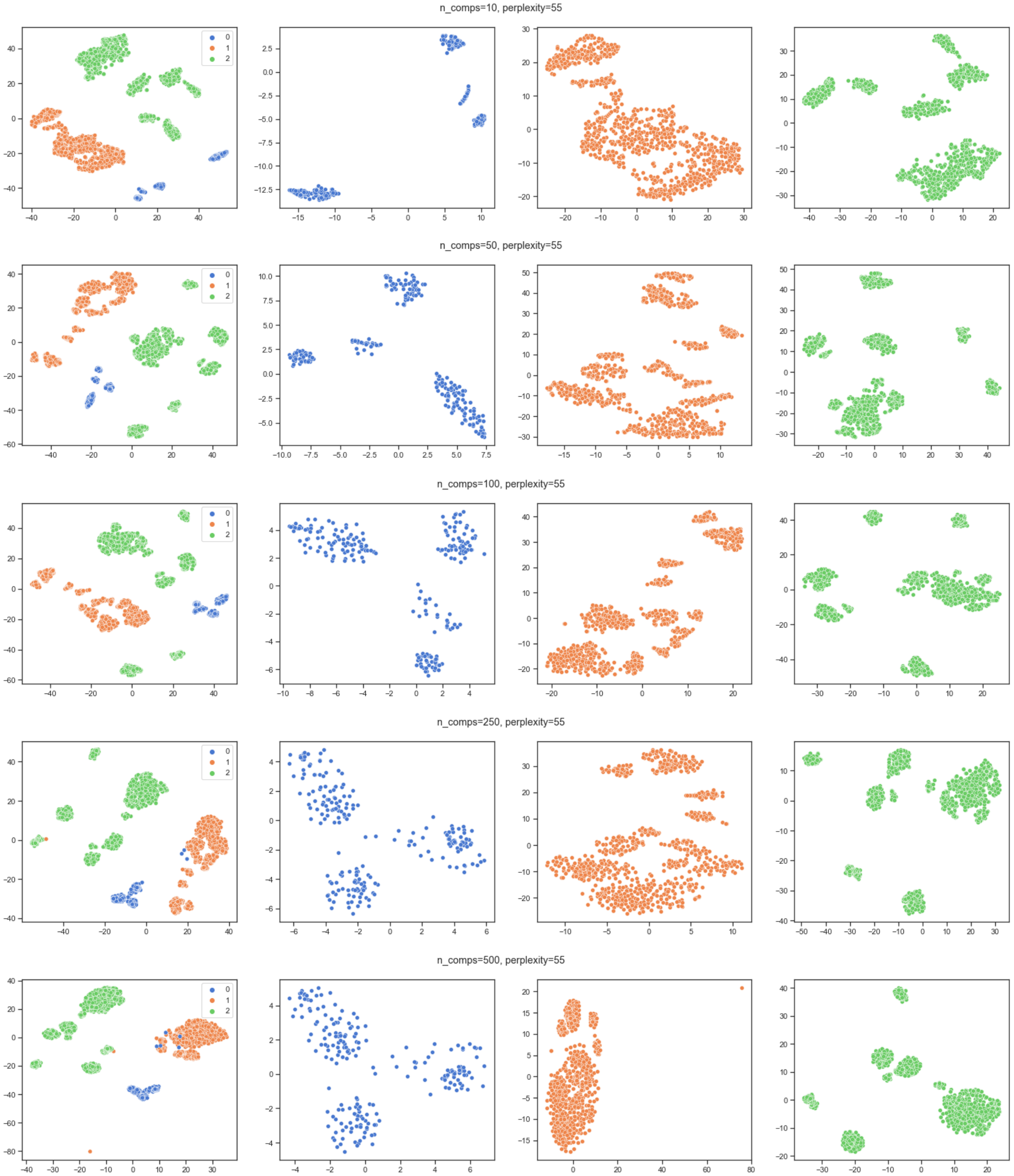

Figure 3.1.1: T-SNE results run on 10, 50, 100, 250, and 500 PCs projections.

✦ Problem 3.2

(13 points) Pick three hyper-parameters below and analyze how changing the hyper-parameters affect the conclusions that can be drawn from the data. Please choose at least one hyper-parameter from each of the two categories (visualization and clustering/feature selection). At minimum, evaluate the hyper-parameters individually, but you may also evaluate how joint changes in the hyper-parameters affect the results. You may use any of the datasets we have given you in this project. For visualization hyper-parameters, you may find it productive to augment your analysis with experiments on synthetic data, though we request that you use real data in at least one demonstration.

Some possible choices of hyper-parameters are:

Category A (visualization)	Category B (clustering/feature selection)
T-SNE perplexity	Effect of number of PC's chosen on clustering
T-SNE learning rate	Type of clustering criterion used in hierarchical clustering (single linkage vs ward, for example)
T-SNE early exaggeration	Number of clusters chosen for use in unsupervised feature selection and how it affects the quality of the chosen features
T-SNE initialization	Magnitude of regularization and its relation to your feature selection (for example, does under or over-regularizing the model lead to bad features being selected?)
T-SNE number of iterations/convergence tolerance	Type of regularization ($L1$, $L2$, elastic net) in the logistic regression step and how the resulting features selected differ

For visualization hyper-parameters, provide substantial visualizations and explanation on how the parameter affects the image.

For clustering/feature selection, provide visualizations and/or numerical results which demonstrate how different choices affect the downstream visualizations and feature selection quality.

Provide adequate explanations in words for each of these visualizations and numerical results.

➡ Ans

- T-SNE perplexity
  - The perplexity is is effectively the number of nearest neighbors.
  - When the perplexity is small, T-SNE considers a smaller number of neighbors. So the embedding will focus on the small local pattern and ignores the global information
  - Larger perplexity lead to more nearest neighbors and less sensitive to small structure, the small local pattern close to each other might have chance glued together.

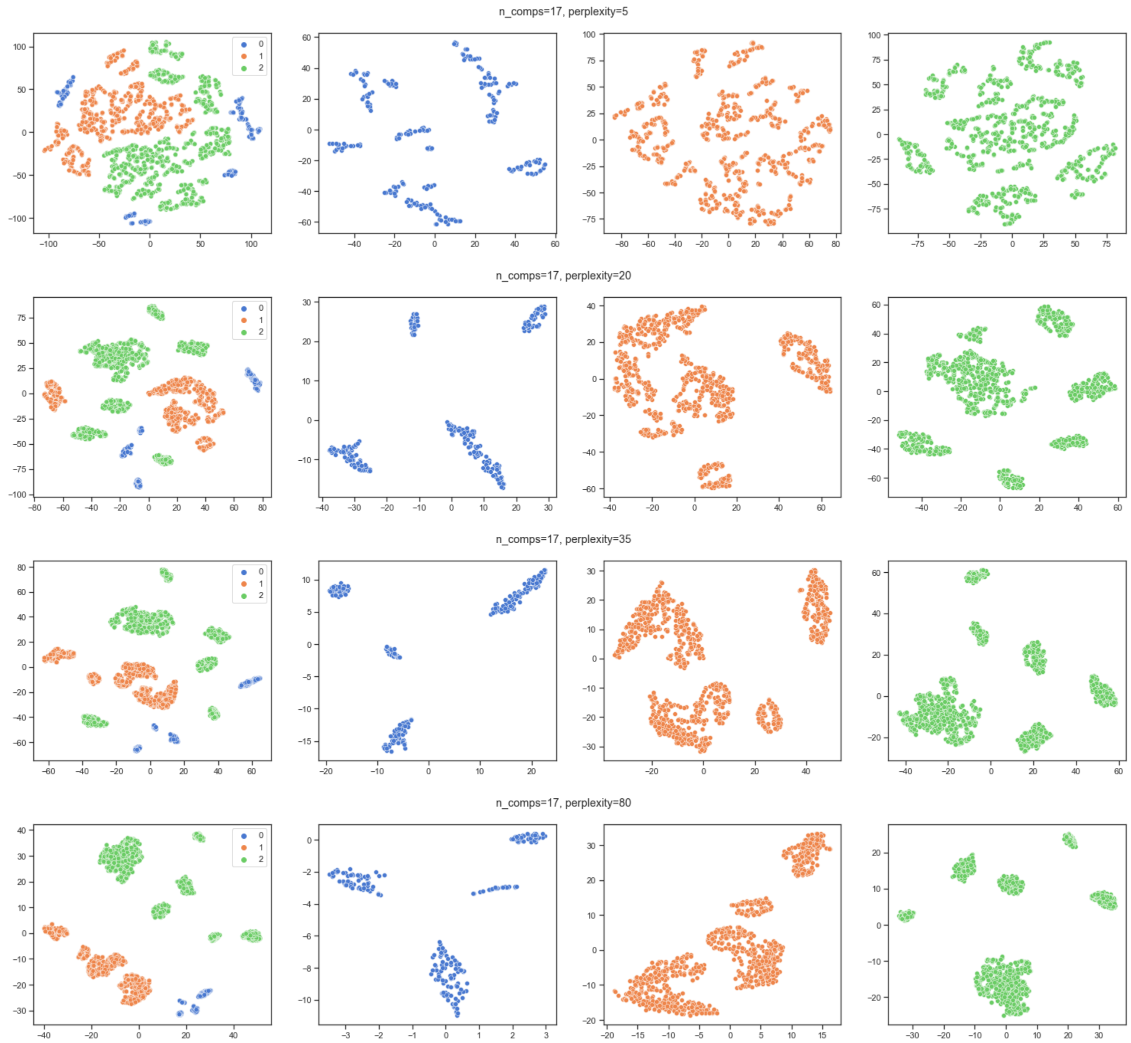

Figure 3.1.1: T-SNE results run on perplexity = 5, 20, 35, 80.