Written Report - 6.419x Module 2

Name: kayal_t_vizhi

■ Problem 2: Larger unlabeled subset

Part 1: Visualization

A scientist tells you that cells in the brain are either excitatory neurons, inhibitory neurons, or non-neuronal cells. Cells from each of these three groups serve different functions within the brain. Within each of these three types, there are numerous distinct sub-types that a cell can be, and sub-types of the same larger class can serve similar functions. Your goal is to produce visualizations which show how the scientist's knowledge reflects in the data.

1. (3 points) Provide at least one visualization which clearly shows the existence of three main brain cell types as 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.

Solution:

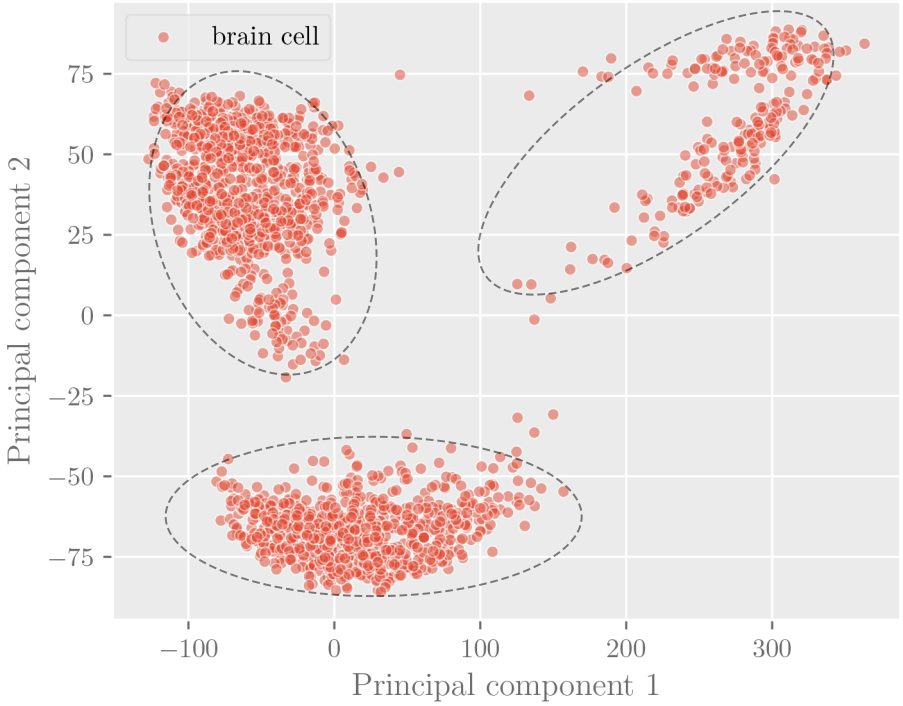

FIGURE 1. Visualization using PCA

Figure 1 is the projection of the data `p2_unsupervised` onto the first two principal components. In this representation, three visually distinct and non-overlapping clusters are observable (encircled in dotted lines). There appears to be considerable distance between these clusters in this representation therefore they are well separated in the high dimensional space. This supports the claim that cells in the brain are either excitatory neurons, inhibitory neurons, or non-neuronal cells. Since they serve different functions in the brain, their gene expressions vary significantly and can be seen in Figure 1.

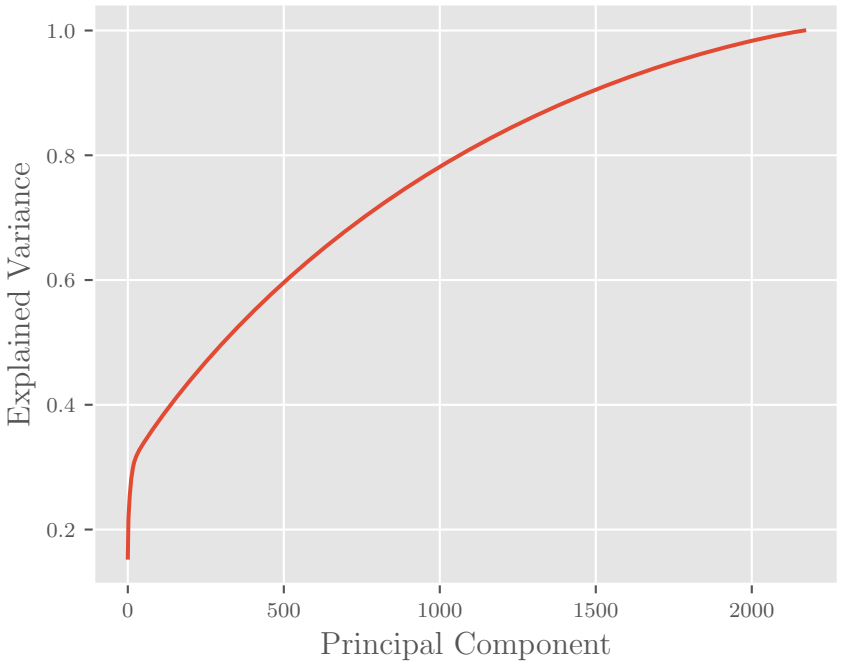

FIGURE 2. Explained Variance by Principal Components

2. (4 points) 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.

Solution:

I chose to visualise with T-Stochastic Neighbour Embedding since it can capture high dimensional local clusters effectively. From Figure 2, it is seen that the to explain 60% of the variance in data about 500 Principal components are required. Hence for this visualization, I chose 500 principal components so that more of the clustering information could be captured by the T-SNE algorithm. From the visualization shown in Figure 3 multiple sub-groups (encircled in blue dotted lines) within each main group (encircled in black dotted lines) can be observed. For two of the main groups the sub-groups are close to each other and for the third group the sub groups are spread apart. This visualization supports the claim that within each of these three types, there are numerous distinct sub-types that a cell can be, and sub-types of the same larger class can serve similar functions.

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 data set 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.

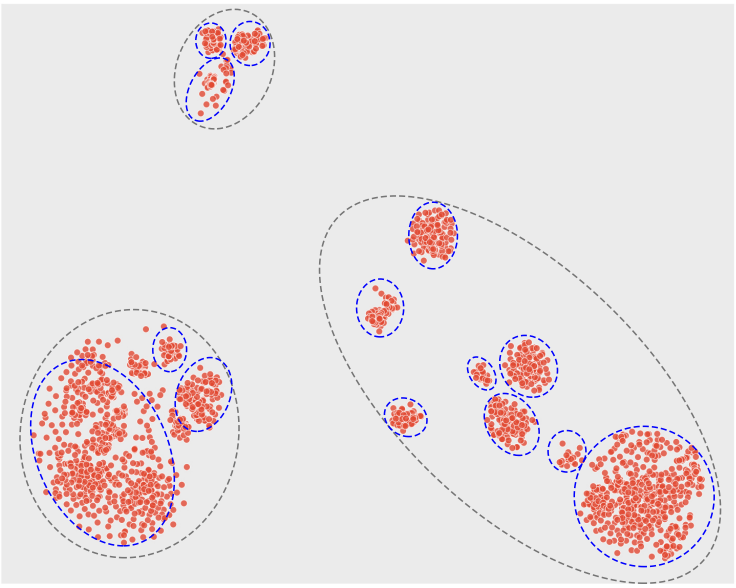

FIGURE 3. Visualization using TSNE (perplexity=40, early_exaggeration=12, learning_rate=200 PCs=500, n_iter=1000)

1. (4 points) Using your clustering method(s) of choice, find a suitable clustering for the cells. Briefly explain how you chose the number of clusters by appropriate visualizations and/or numerical findings.

Solution: **A suitable number of clusters for the cells is k=3**. For this analysis I used the `p2_unsupervised`,*educeddataset*. *singletonclustersformedinthelinkagearetheonlynon* − *leafnodesinthelinkage*1001[1].

```
Output:
ward linkage
k = 3, t=2000 silhouette score: 0.0883
k = 2, t=5000 silhouette score: 0.2957
```

The obtained dendrogram is shown in the left sub-figures of Figure 4 and Figure 5 with different thresholds. The largest distance between clusters are observed when the number of clusters `k` is either `2 or 3` hence are more robust clustering options compared to other choices of `k` To choose between these two choices of `k`, data with the cluster labels are projected on the principal components as shown in the right sub-figures of Figure 4 and Figure 5. It can be seen visually that the `k=3` is more suitable. Comparing the silhouette scores, `k=3` has a lower score which indicates better clustering.

2. (6 points) 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 l_1, l_2, 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.

Solution: My choice of Regularization hyper-parameter $C = 0.1$. For this analysis a `K-fold cross validation` was performed to choose the regularization hyper-parameters with `n_splits=10`. In order to avoid sampling bias, the data was first shuffled before splitting into 10 sub-samples. `'l1'` (lasso regularization) penalty was chosen to perform feature selection. To evaluate the performance on the validation set, `accuracy_score` metric was used. Since the data is well separated, the classifier performs well with a test accuracy of 0.9968 on the validation set for all values of C equal and above 0.1 as shown in the output of the code below.

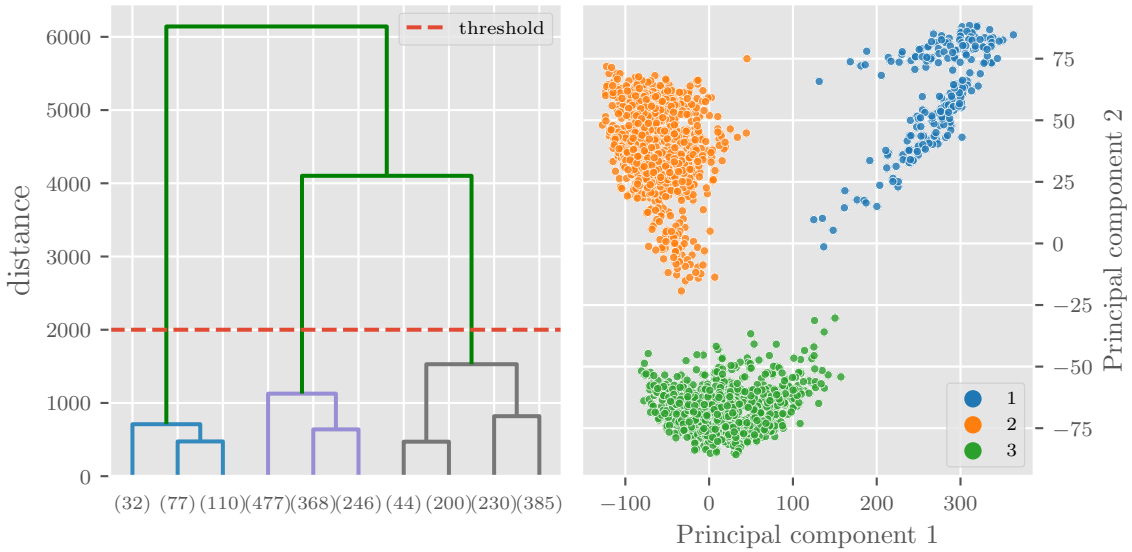

FIGURE 4. (left) Hierarchical clustering Dendogram(linkage=**'ward'**, metric='euclidean') (right) Projected data on first two principal components with labels generated with threshold distance **t=2000**

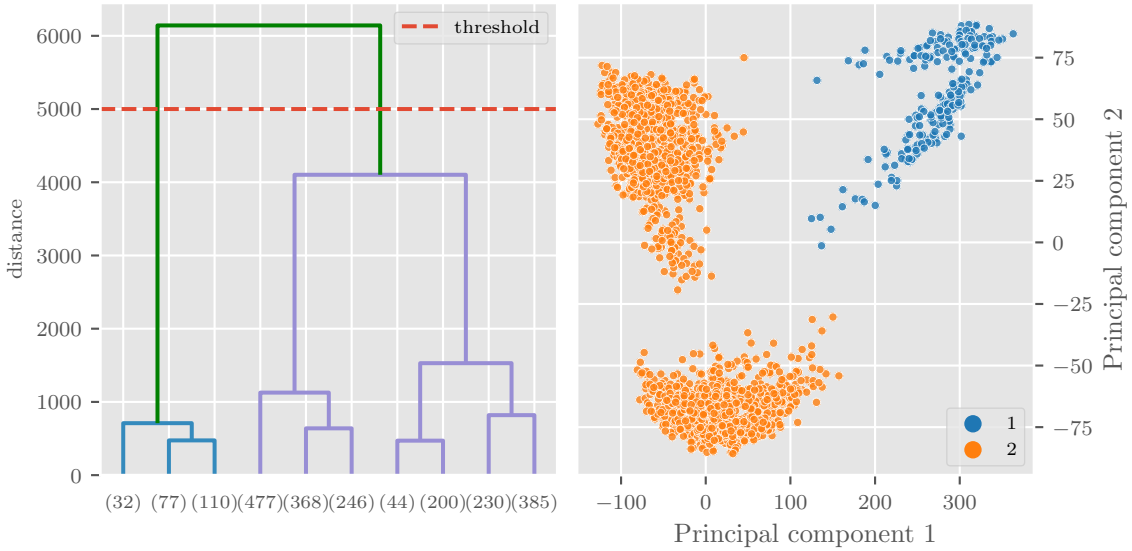

FIGURE 5. (left) Hierarchical clustering Dendogram(linkage=**'ward'**, metric='euclidean') (right) Projected data on first two principal components with labels generated with threshold distance **t=5000**

```
Output:
C  = 0.001
Accuracy: mean= 0.9479 std=0.0194

C  = 0.01
Accuracy: mean= 0.9963 std=0.0058
```

Features	Selected	Highest Variance	Random
Test Accuracy	0.86282	0.90884	0.31588
Train Accuracy	0.95822	0.0.9805	0.40204
Test Error rate	0.13718	0.09116	0.68412

TABLE 1. Comparison of model performance

```
C  = 0.1
Accuracy: mean= 0.9968 std=0.0055

C  = 1
Accuracy: mean= 0.9968 std=0.0055

C  = 2
Accuracy: mean= 0.9968 std=0.0055

C  = 5
Accuracy: mean= 0.9968 std=0.0055
```

3. (9 points) 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 in p2_evaluation 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 the obtained score with two baselines: random features (take a random selection of 100 genes), and high-variance features (take the 100 genes with highest variance). Finally, 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.

Solution: After finalising the regularization parameter, a logistic regression model is trained on the entire `p2_unsupervised_reduced` data set and the 100 features with largest coefficients are selected by ranking the coefficients using the maximum absolute value over classes.

Comparison of model performance with baselines:
The model accuracy is compared with the two baselines as shown in the Figure 6 and Table 1 . It seen that the model trained with the selected features performs very close the model trained with the highest variance features. The model trained with random features performs poorly.
Comparison of the variance of selected features vs highest variance features :
From Figure 7 it can be seen that the selected features have comparatively lower variance with an average variance of 10.47 while the highest variance features have an average of 15.57. Yet the model trained with the selected features performs decently.

■ Problem 3: Influence of Hyper-parameters

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?

Solution:

For this analysis I chose `p1` dataset labelled using k-means with k=5. When data is projected onto fewer principal components, a lot of the embedding information is lost and most observations are closely spaced.

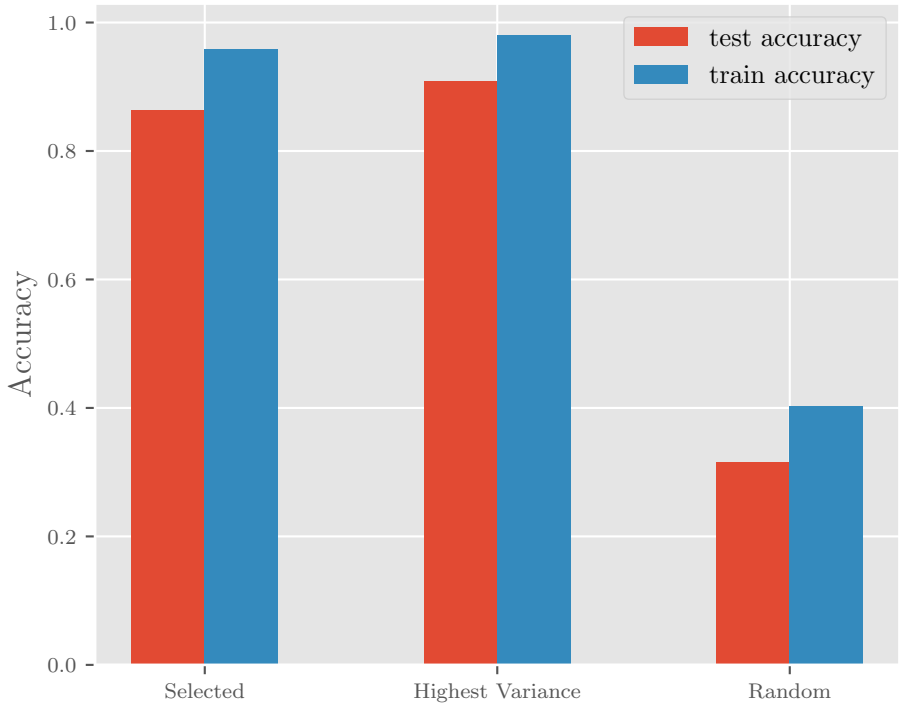

FIGURE 6. Comparison of performance of selected features with the baselines (Highest variance features and Random features)

As a trivial example it is seen in when we choose no of $PCs = 1$ all points are placed close except for a few points that are separated on this axis as shown in Figure 8. As the number of principal components is increased, more embedding information is used by the algorithm and it is able to form better localized clusters. For $PCs = 10$ 5 clusters are observed but they are still densely packed. For $PCs = 50$ and $PCs = 100$ the clusters start spreading out more and merging with one another. If the number of principal components are too large, **the curse of dimensionality** takes over. In high dimensional space the observations are so far spread out that they tend to be equally distant from one another. Hence T-SNE tends to find larger and more spread out clusters when we increase the number of principal components substantially. This effect can be seen in Figure 8.

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.

Solution:

1. CATEGORY A

The data chosen for this analysis is the `p2_unsupervised_reduced`. The projection of the data into the the first two principal components with labels learnt from hierarchical clustering algorithm (ward linkage,

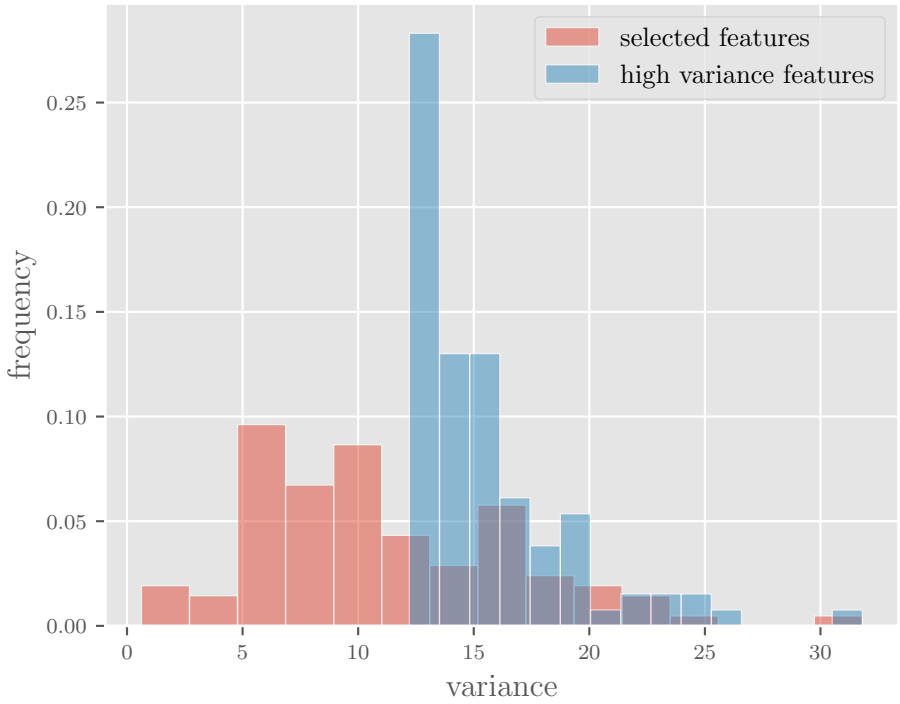

FIGURE 7. Histogram depicting the variance of selected features(100) vs the variance of highest variance features(100)

euclidean metric) are shown in Figure [9]. A suitable number of classes for this data is taken to be 3 corresponding to excitatory neurons, inhibitory neurons, and non-neural cells. These three classes are used as reference to see how the T-SNE plot clusters/groups these three classes and to see the effects of changing hyper-parameters.

1.1. **T-SNE early exaggeration.**

The T-SNE Early exaggeration temporarily increases the attractive forces between neighbours by a factor (default 12) chosen for a certain number of iterations (default 250) of the T-SNE Algorithm. Exaggeration can also be used during the normal optimization regime to form more densely packed clusters, making the separation between clusters more visible [2]. From the 3 sub-figures in Figure 10 the differences can be noted. Note that I've restricted the number of iterations to 500 to try to show the differences in clustering. Since the clusters are well separated in the high dimensional space, the algorithm converges for n_iter=1000 for any exaggeration. When the exaggeration=1 there is effectively no exaggeration and it can be seen that the algorithm converges at a local minima and is unable to group similar islands together.The blue islands is separated by a few islands of the orange class since the repulsive force between different clusters are strong. When exaggeration=12 the sub groups are clustered together . When exaggeration=50, the clusters are more densely packed.

1.2. **T-SNE number of iterations.**

For this analysis I applies an early exaggeration of a factor of 12 up to the 250^{th} iteration and stopped the algorithm at $250, 260, 270, 275, 300$ and 500 iterations and plotted the results as shown in Figure 11. It can be seen that at iteration 250 due to exaggeration the attractive forces are high resulting in the clusters being far apart and densely packed. As we increase the number of iterations the repulsive forces take charge and the clusters spread out more freely. Finally at iteration 500 the clusters balance the attractive and repulsive

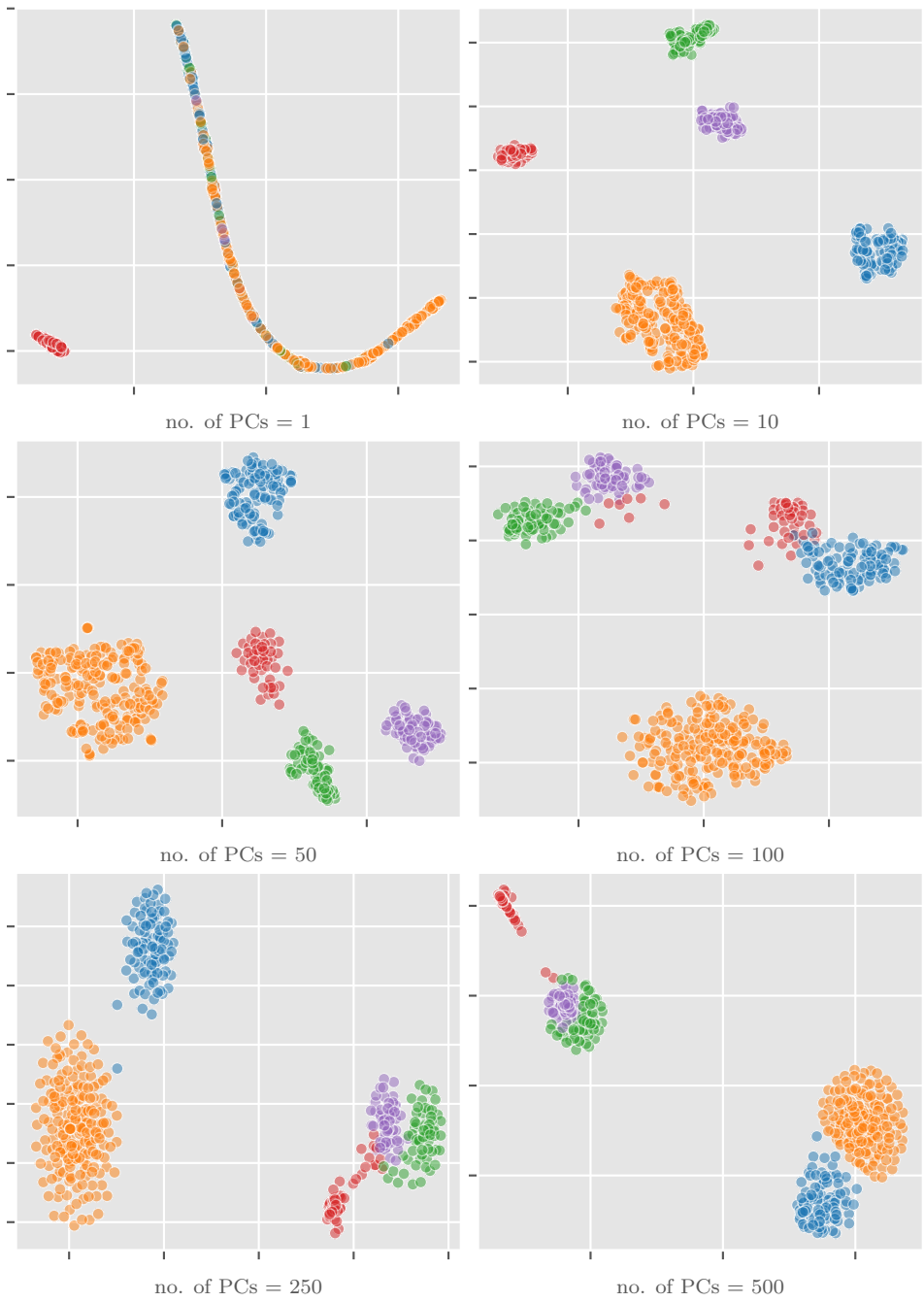

FIGURE 8. Effect of varying the number of Principal Components in TSNE visualization.

forces and assume more globular shape. Also, the islands are grouped together in accordance with the 3 presumed classes.

1.3. **Combined effect of number of iterations and early exaggeration.**
Figure 12 shows the combined effect of iteration and early exaggeration. With Low early exaggerations results is discovereing subgroups effectively but it is unable to group the subgroups together. It can be seen that with early_exaggeration=12 and n_iter=500 that a good clustered representation is achieved. And by increasing the number of iterations the subgroups or islands are more regular in shape and well separated.

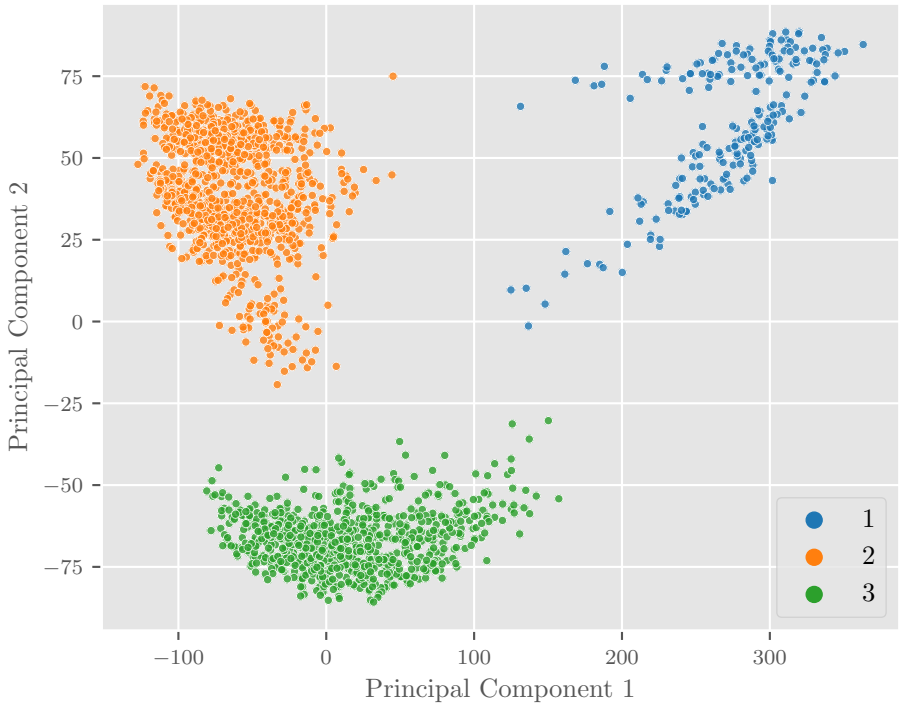

FIGURE 9. Data projected on first two principal components with labels learnt from hierarchical clustering algorithm.

2. CATEGORY B

2.1. **Type of clustering criterion used in hierarchical clustering.**

The differences in the effects of three types of linkages namely - single linkage, complete linkage and ward linkage - on visualization and feature selection is analysed in this section. For this exercise I used the `p2_unsupervised_reduced` data set. The data is projected onto the first two principal components and I chose this representation to input into the agglomerate clustering algorithm.
Dendrograms for each type of linkage were generated. By computing the silhouette for generating cluster labels were chosen as shown in figures [13], [14], [15]. Also the dendrogram has been truncated at level 10.

2.1.1. *'ward' linkage.*

From the dendogram in Figure 13 appropriate number of clusters are likely 2 or 3 as the distance between clusters are relatively large for these numbers and presumably results in robust clustering. I chose to generate labels using distance threshold t=5000 and t=2000 respectively and computed the silhouette scores. Since number of clusters=3 minimizes the silhouette scores I chose it for further analysis.

```
Output:
ward linkage
n_clusters = 3, t=2000 silhouette score: 0.0883
n_clusters = 2, t=5000 silhouette score: 0.2957
```

2.1.2. *'complete' linkage.*

From the dendogram in Figure 14 appropriate number of clusters are likely 2 or 3 as the distance between clusters are relatively large for these numbers and presumably results in robust clustering. I chose to generate labels using distance threshold t=250 and t=400 respectively and computed the silhouette scores. Since number of clusters=3 minimizes the silhouette scores I chose it for further analysis.

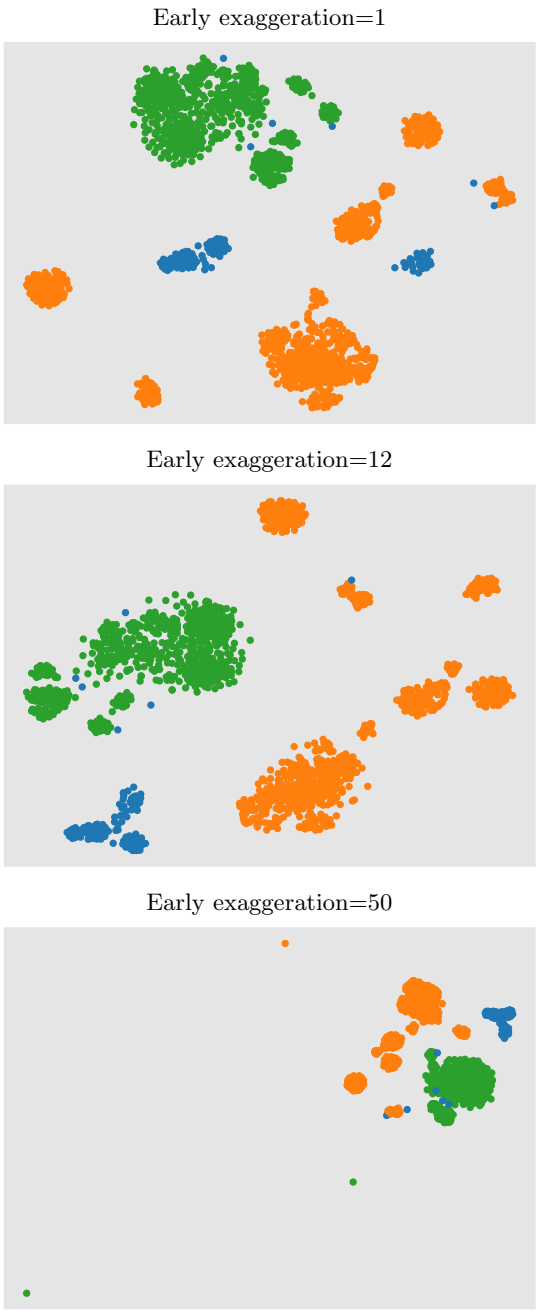

FIGURE 10. Effect of varying the early exaggeration in TSNE visualization (PCs=500, learning_rate='200',n_iter=500, perplexity=40).

```
Output:
complete linkage
n_clusters = 3, t=250 silhouette score: 0.1246
n_clusters = 2, t=400 silhouette score: 0.2983
```

2.1.3. *'single' linkage.*

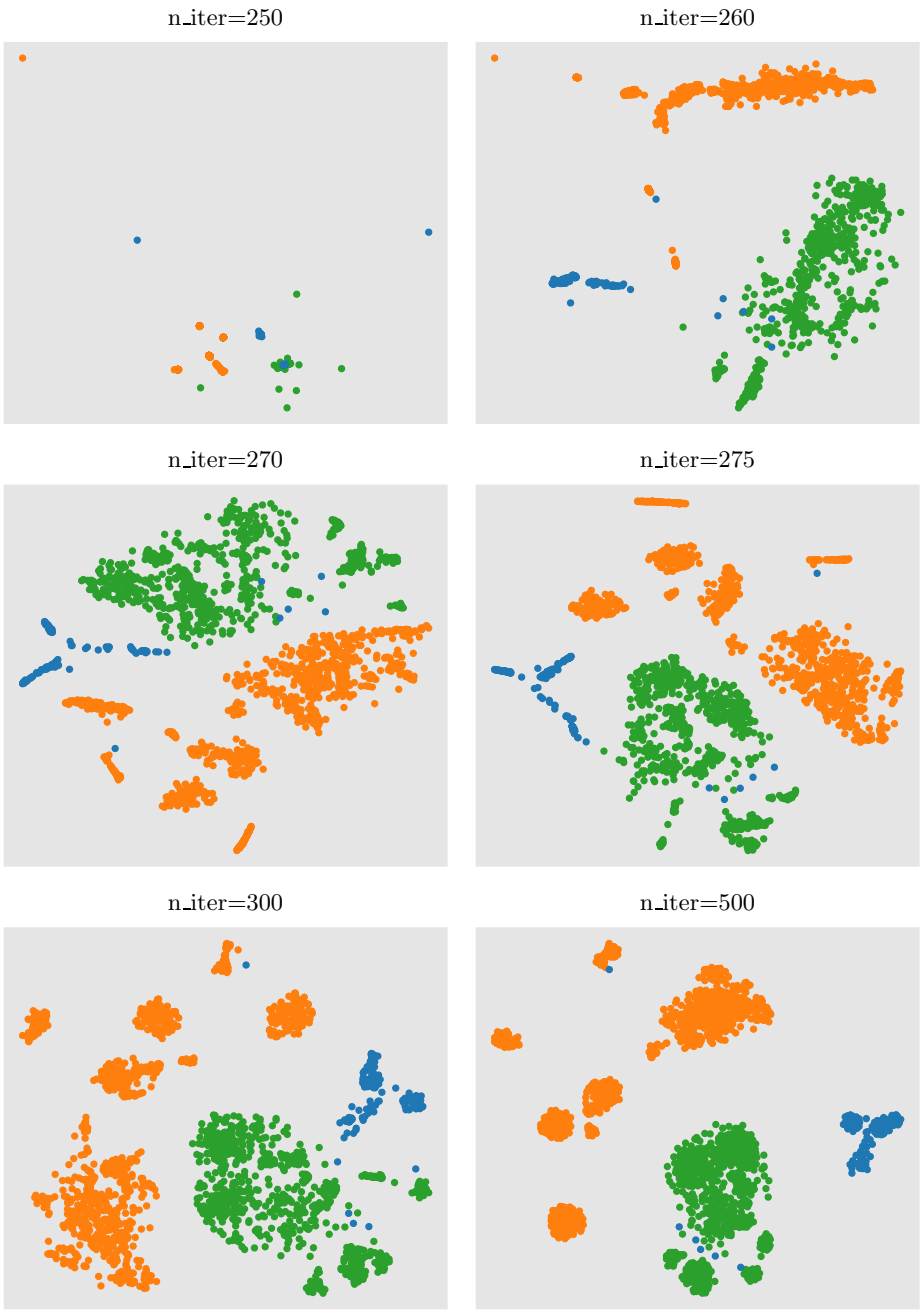

FIGURE 11. Effect of varying the number of iterations in TSNE visualization (PCs=500, learning_rate='200', perplexity=40, early_exaggeration=12).

From the dendogram in Figure 15 appropriate number of clusters are likely 2 or 5 as the distance between clusters are relatively large for these numbers presumably results in robust clustering. I chose to generate labels using distance threshold t=25 and t=35 respectively and computed the silhouette scores. Since number of clusters=5 minimizes the silhouette scores I chose it for further analysis.

```
Output:
single linkage
n_clusters = 5, t=25 silhouette score: 0.0802
```

FIGURE 12. Effect of varying the early exaggeration and n_iter in TSNE visualization (PCs=500, learning_rate='200', perplexity=40).

```
n_clusters = 2, t=35 silhouette score: 0.1606
```

Judging by the way the data is distributed, there are three visually distinct clusters that are not globular in shape (more ellipsoidal) and the three different linkage metrics produce different numbered/shaped clusters. The clusters produced by complete linkage are more potato shaped and therefore unable to produce clusters efficiently for this dataset. Ward and single linkage however produce elongated clusters that match expectation. But single linkage is very sensitive to the two outlier points that are slightly apart from cluster 1 and 2 and are labelled their own cluster 4 and 5.The effect of these points on single linkage can be clearly seen in Figure [15].

2.1.4. *Comparing Silhouette scores.*

```
Output:
n_clusters = 3 (ward linkage) silhouette score: 0.0883
```

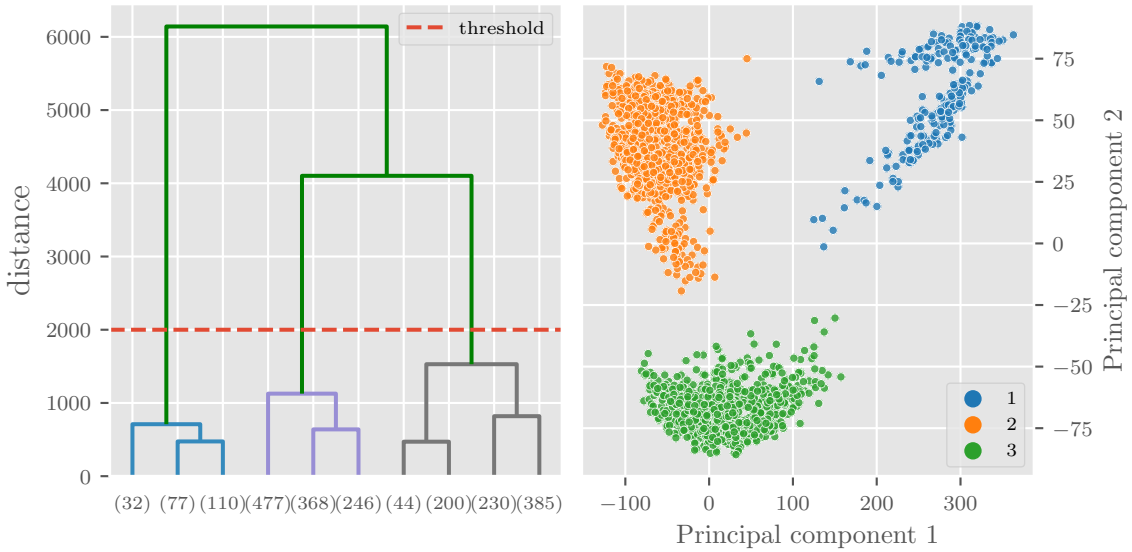

FIGURE 13. (left) Hierarchical clustering Dendogram(linkage=**'ward'**, metric='euclidean') (right) Projected data on first two principal components with labels generated with threshold distance t=2000

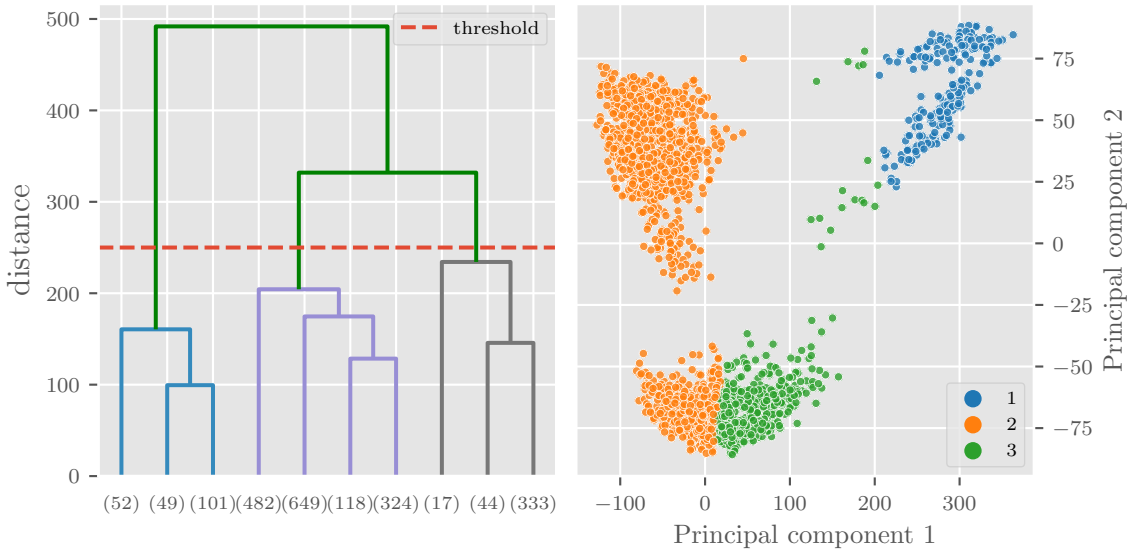

FIGURE 14. (left) Hierarchical clustering Dendogram(linkage=**'complete'**, metric='euclidean') (right) Projected data on first two principal components with labels generated with threshold distance t=250

```
n_clusters = 3 (complete linkage) silhouette score: 0.1246
n_clusters = 5 (single linkage) silhouette score: 0.0802
```

The average silhouette scores of the observations show that the clustering with complete linkage performs the worst. Single linkage has the best silhouette score. However as seen from the Figure [15] this is over-fitting.

2.1.5. *Evaluation accuracy.*

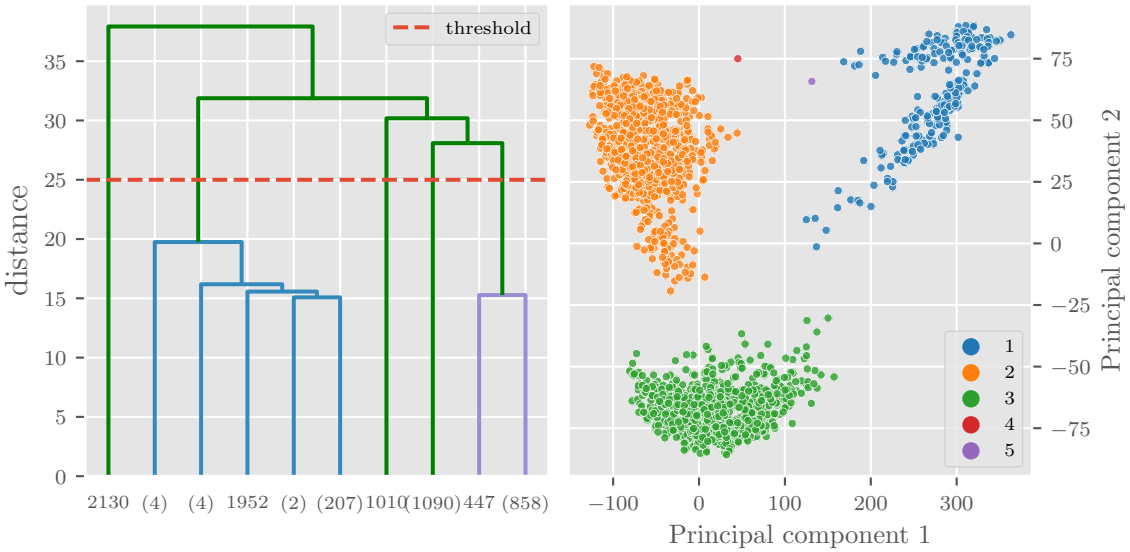

FIGURE 15. (left) Hierarchical clustering Dendogram(linkage=**'single'**, metric='euclidean') (right) Projected data on first two principal components with labels generated with threshold distance t=25

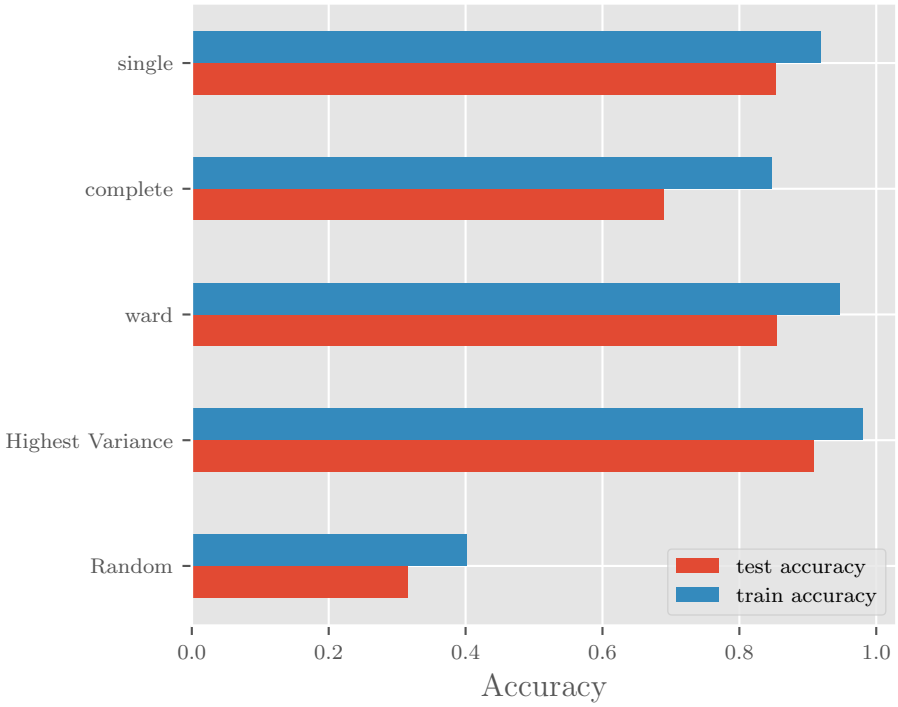

FIGURE 16. Comparison of Selected features performance using ward/complete/single linkages with baselines on the evaluation dataset

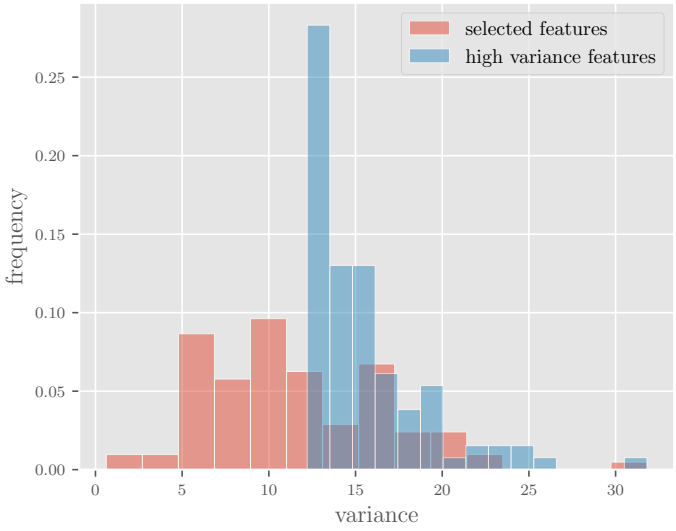

FIGURE 17. Histogram of variance of selected features(ward linkage) vs highest variance features(100

100 features were selected for each of the 3 cases by ranking the coefficients using the maximum absolute value over classes. A logistic regression model was trained on the dataset `p2_evaluation_reduced` with these 100 features for each case and the test and training accuracy were recorded. These accuracy's are compared to the baseline models trained with random features (take a random selection of 100 genes), and high-variance features (take the 100 genes with highest variance) in Figure 16. The features selected with ward linkage performs the best on the evaluation data. The features selected with single linkage comes close second and the effect of verfitting to those outliers results in a slightly poor performance but much better than complete linkage. Therefore the choice of linkages has a significant effect on clustering.

```
Output:
features (ward linkage): test accuracy: 0.86101, train accuracy 0.948
features(complete linkage): test accuracy: 0.68863, train accuracy 0.84773
features (single linkage): test accuracy: 0.8583, train accuracy 0.94058
features (high variance): test accuracy: 0.90884, train accuracy 0.9805
features (random): test accuracy: 0.31588, train accuracy 0.40204
```

2.1.6. *Feature variance.*
From Figure 17, Figure 18, Figure 19 It can be that the features selected by clustering using complete linkage has lower average variance compared to the other three cases. the features selected by clustering using ward linkage is the most comparable to the high variance features. And single linkage case performs second best.

2.1.7. *Conclusion.*
Complete linkage is not suitable for this data as the clusters in the data are inherently irregular. Hence ward/single can be used. Even though clustering with single linkage produced lower silhouette scores, the features selected it did not perform better that the features selected by clustering with ward linkage. Therefore ward linkage performs best for this dataset.

REFERENCES

[1] Eads, Damian, "Scipy.cluster.hierarchy.dendrogram," 2007.
[2] D. Kobak and P. Berens, "The art of using t-SNE for single-cell transcriptomics," *Nat Commun*, vol. 10, p. 5416, Nov. 2019.

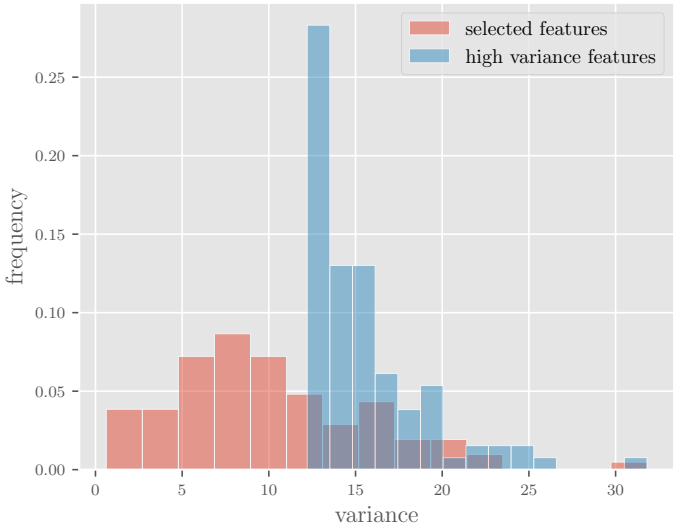

FIGURE 18. Histogram of variance of selected features(single linkage) vs highest variance features(100

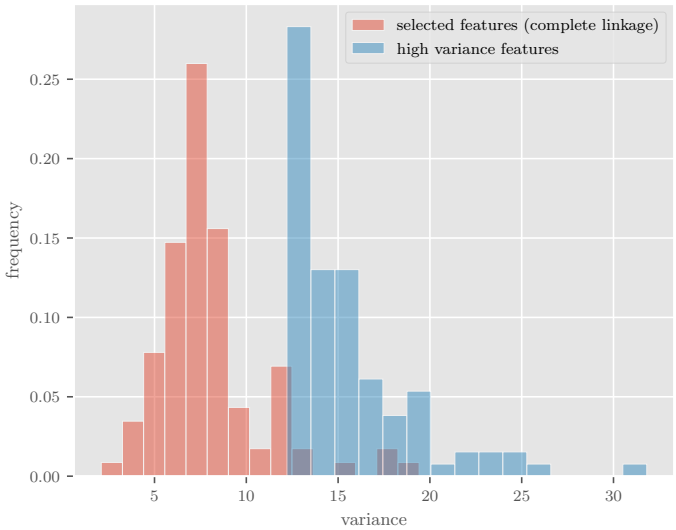

FIGURE 19. Histogram of variance of selected features(complete linkage) vs highest variance features(100