Assignment 4 – Emil Collén

Exercise 3

The datasets I chose to go with are digits, wine and breast cancer. These are all from sklearn and are imported using load.digits() for example. During this exercise, not the entire digits dataset is used, only the 500 first samples are in play. This is partly because of the time to run, but mainly because the result is easier to read with “only” 500 samples. Classes, dimensions and samples are displayed in the table below. The wine is a rather small dataset, but I chose to go with it anyway, since it is otherwise suited for the presentation of these tasks.

	Digits	**Wine**	**Breast cancer**
Classes	10	3	2
Dimensions	64	13	30
Samples	1797 (only 500 used)	178	569

3.1 Comparison of DR techniques

One downside to having the wine dataset, with only 178 samples, is that each time the program is being run, the starting point and therefore time can vary relatively much. The threshold value and learning rate are individual for each dataset and is pinpointed by tracking the stress progress for each iteration. When the reduction is barely noticeable, the “elbow” is basically found – this is how these two parameters were set for each dataset.

In figure 1 below, we see the result of the 3.1 task. In it, we can see that the three techniques all work in different ways as they separate classes and present the results in very different ways. In my opinion, the technique that works the best is **highly** dependent on the dataset.

Digits: In the digits dataset, I would say that t-SNE is the obvious best technique as it clearly separates and distinguishes the different classes from each other. Both Sammon mapping and PCA shows one fairly compact cluster, where different parts of the cluster are somewhat separated. Between these two, I would say that Sammon mapping outperforms PCA.

Wine: In the wine dataset, I would say that Sammon mapping shows the best separation of classes. With a few exceptions, simple lines could almost be drawn to separate the three classes in this scatterplot. For this dataset, I would say that t-SNE shows the worst outcome as the classes are very difficult to distinguish from one another.

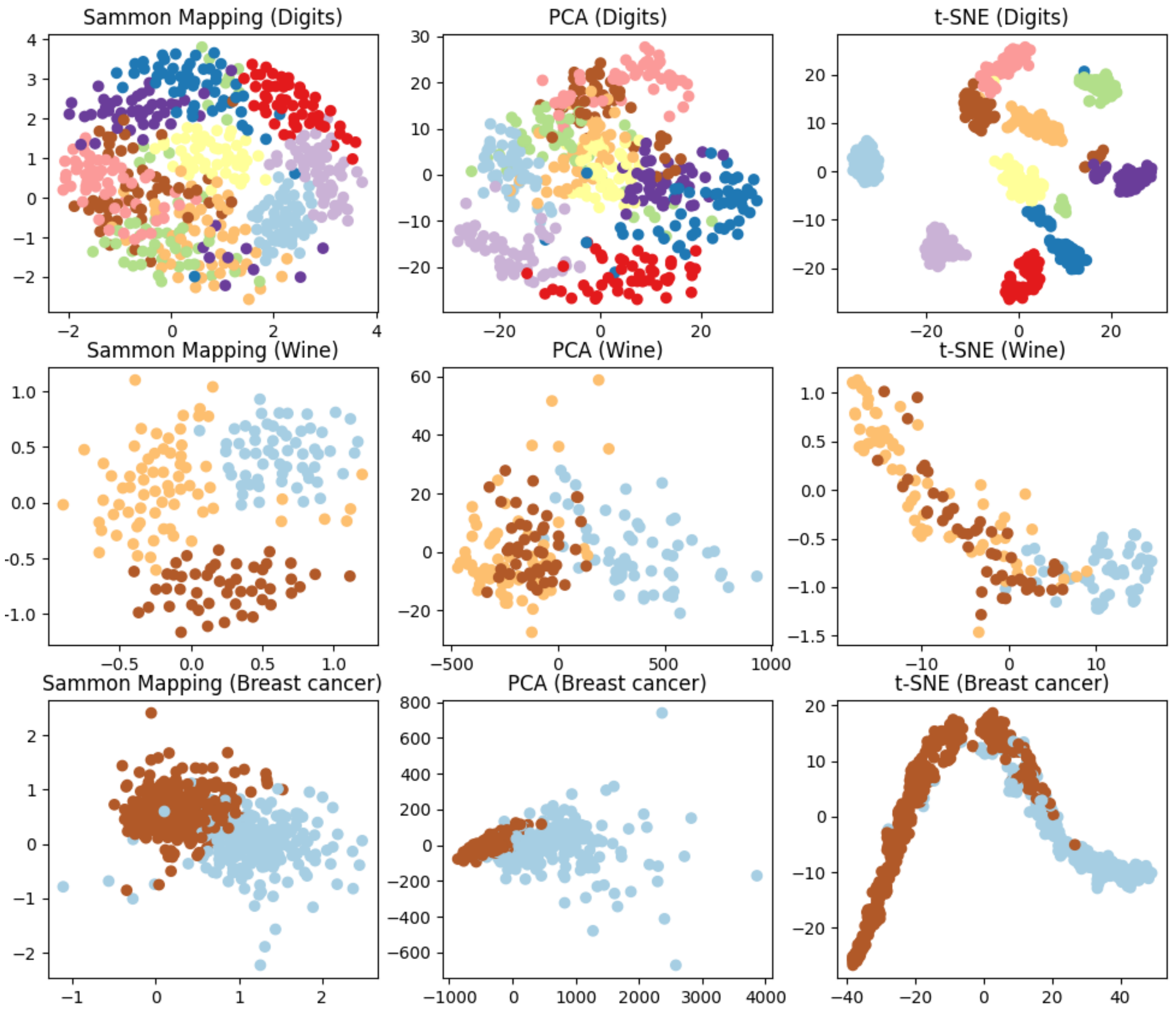
Breast cancer: In the breast cancer dataset, all three techniques work fairly well. If I had to choose only one, I would say that Sammon mapping shows the best separation of the classes as it makes it that both classes could (with some exceptions) be separated with a line.

Sammon mapping: In sammon mapping, we try to maintain the pairwise distances between data points and preserve the structure of the data. That is, points that are originally close in the input space, should remain close in the output space. We do this by computing the distance matrix of the input space (the data, in its multidimensional form), then we randomly place new points in a n*2 output space, where n is the same number of points. Then we calculate the stress, which we try to minimize using gradient descent. We do this by altering our output space, until we reach our threshold or number of iterations.

PCA: When using PCA, we try to find a plane, which can best show the most variance in the data. That is, we try to find a line where the average distance to all points is the lowest. This line shows the most variance. Then we try to find a second line (axis), that is orthogonal to the first one, using the same technique. These two lines form the plane in which the data is presented.

t-SNE: Generally, t-SNE works best on complicated datasets with distinct patterns or structures. It focuses on the preserving the local similarities between data points and creates an output space based on the pairwise similarities.

We can see that the different techniques can have problems separate some classes. For example, in the digits dataset, the brown class is frequently quite mixed between other classes, such as the pink, purple and orange classes. Especially sammon mapping and PCA does this, while t-SNE does a better job of separating these classes. In the meantime, the lightblue, light purple and red classes are easier separated in all techniques. In the wine dataset, the brown and orange classes are difficult to separate, while the sammon mapping has no problem of doing this. The same type of conclusions is not as easy to point out in datasets with only two classes, such as the breast cancer, but we can at least see that in both sammon mapping and PCA, the brown class is a bit more distinct and closely clustered.



Figur 1 - Dimensionality reduction techniques

3.2 Comparison of Clustering techniques

I tried all 3 of the DR techniques on k = 3 clusters, see figure 11 and 12 for PCA and Sammon mapping. As mentioned before, different DR techniques work differently, depending on dataset. As the data plotting is performed based on the DR output space, it serves as a foundation for the following clustering. For example, if we look at the digits dataset, when we reduce the dimensionality with t-SNE, we get 10(-ish) clusters of data, which are (for the most part) easily distinguishable from each other. These all make sense when we use k = 10, and perhaps a few other k:s, such as 5, but for the most k:s, it doesn´t. It becomes quite clear how many classes the initial dataset had (2, 3 and 10). The same principle applies for the other DR techniques as well, where the big blob in Bisecting k-Means for the digit dataset, when Sammon mapping its space (fig 12), it doesn´t make sense to split it in three parts. When applying the clustering using different techniques and increasing the k each time, it does give a good insight in to how these techniques work. I therefore chose t-SNE as my DR technique for this task.

With that said, which clustering technique works the best also depends on the number of classes and how many clusters we separate it into. See clustering from k = 2 → k = 10 in figures 2-10 below. If we for example look at the breast cancer dataset in k = 2, we can see that the bisecting k-Means is quite bad at performing clustering on both the digits dataset and the breast cancer. In both cases, it splits them into illogical clusters, where both the other techniques perform rather well. And due to that initial bad clustering and its nature of dividing the biggest cluster, almost all following clusters become quite bad as well.

Digits dataset: On the digits dataset, if we look at k = 2, we can see that bisecting k-Means splits the clusters quite badly immediately. Like previously mentioned, due to the nature of bisecting k-Means, this error doesn´t go away. When we increase k to 3, 4, 5 and so on, we see that already at k = 3, k-Means start to make erroneous clustering, which only gets worse by each cluster size. Therefore, I would say **that agglomerative clustering** works best on this dataset.

Wine dataset: The wine dataset is by far the most difficult dataset to select a winner, as there are no clear clusters in the foundational plotting. Therefore, I really cannot with ease say which one is the best. One could argue that agglomerative clustering does some weird clusterings in some cases. By small differences, I would say that on k > 4, the **bisecting k-Means** shows a better result.

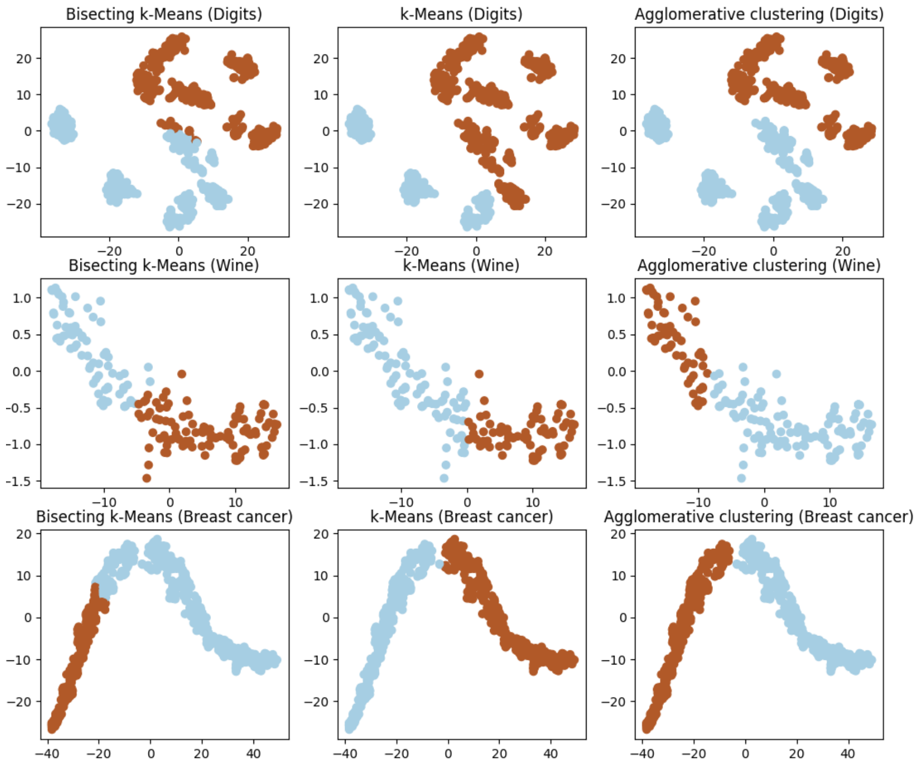
Breast cancer dataset: If we look at the breast cancer dataset with k = 2 and 3, we can conclude that the **agglomerative clustering** technique works best for that dataset. Bisecting k-Means is by far the worst. Regular k-Means and agglomerative are fairly even on most cluster sizes, but in some cases, k-Means divide the dataset quite strangely.

Bisecting k-Means separates the data by always splitting the biggest cluster in two parts. This can clearly be seen in the following figures, where the previous clustering doesn´t change – other than the biggest cluster splitting in two.

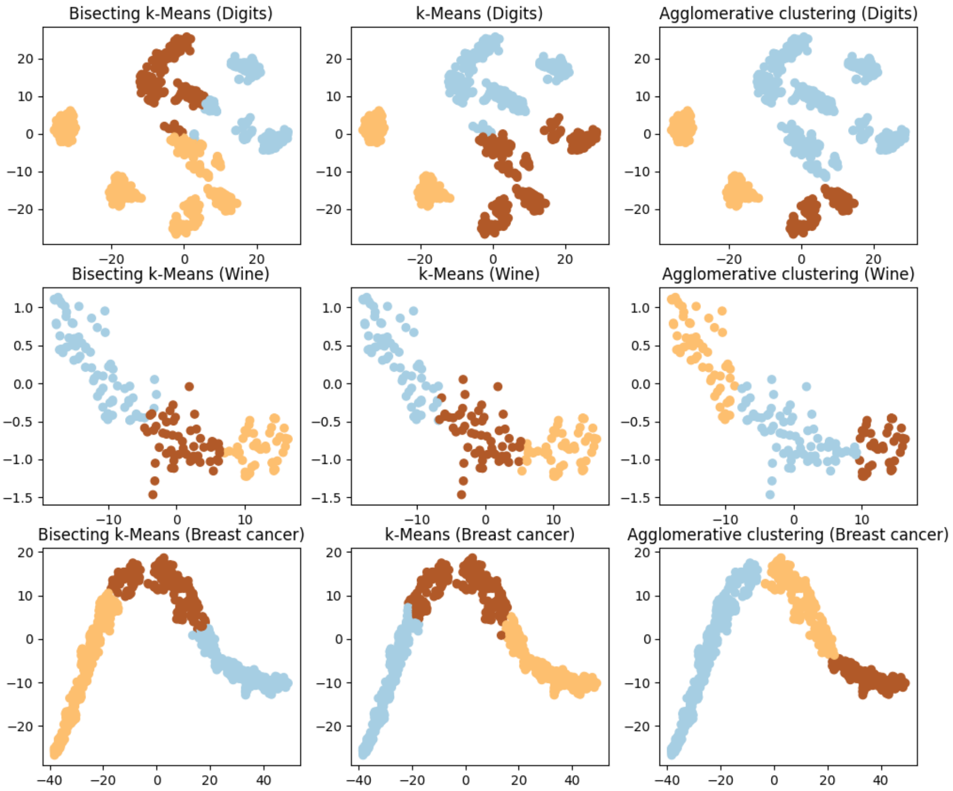
K-Means randomly places k clusters with every element belonging to a cluster. Then each clusters centroid is calculated by the means distance of each point in that cluster. Then each element is assigned to the closest centroid´s cluster. This process is repeated until the it stops changing.

Agglomerative clustering basically has the opposite approach than bisecting k-Means. Here, we use a bottom-up approach, where we start by classing each data point as its own cluster. From here, we take the two clusters that are closest to each other and merge the into one cluster. This is done repeatedly until there is k clusters.

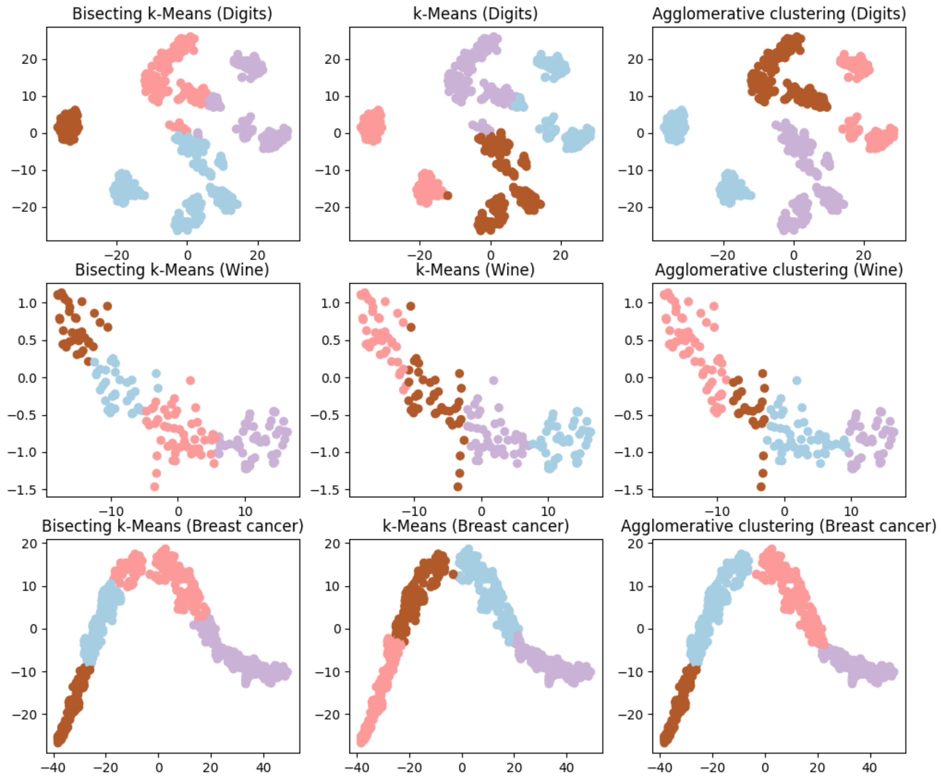
Which clusters are easier to separate from each other depends on the number k and the datasets. The wine dataset for example does now show any obvious clusters, while the digits dataset does. However, not on all k:s. When k = 2 for example, digits can be divided in multiple ways while the breast cancer dataset cannot (and basically no other k). The digit dataset is the easiest one to cluster, and is beautifully done on every k with the agglomerative clustering (contrary to the other two techniques).



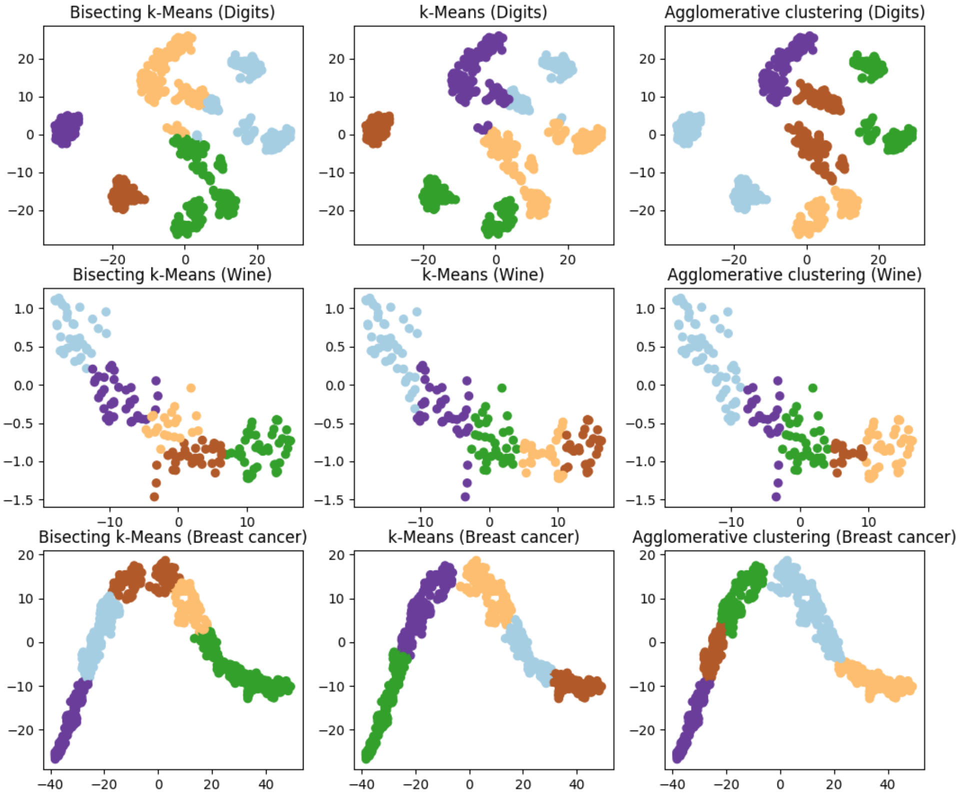
Figur 2 - Clusters with k = 2



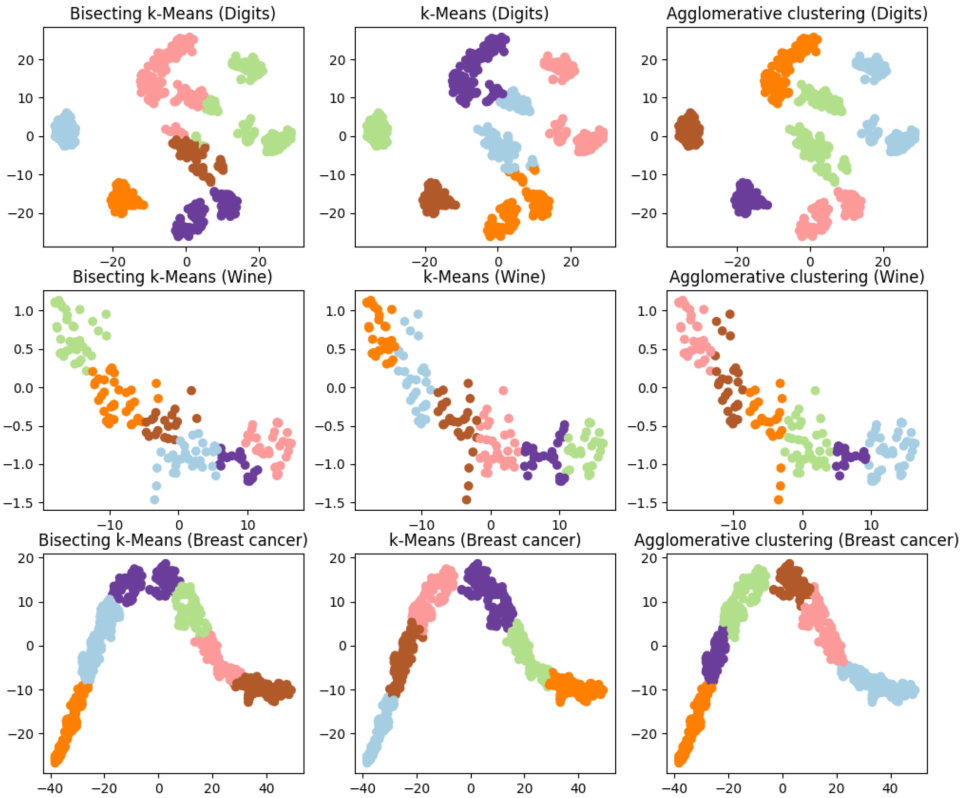
Figur 3 - Clustering with k = 3



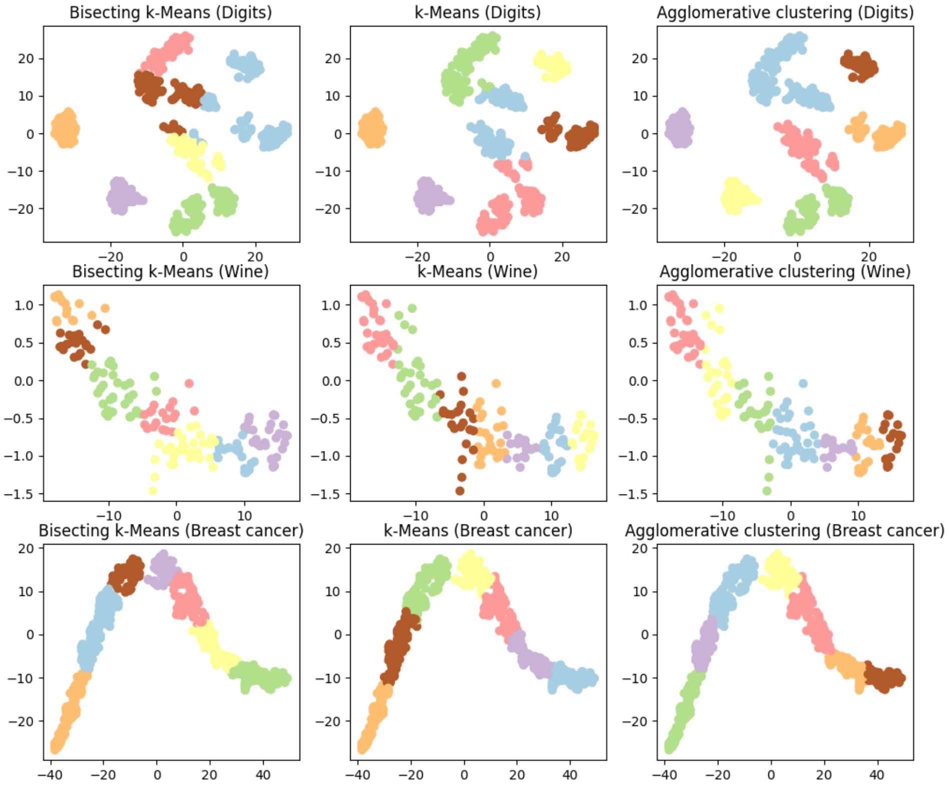
Figur 4 - Clustering with k = 4



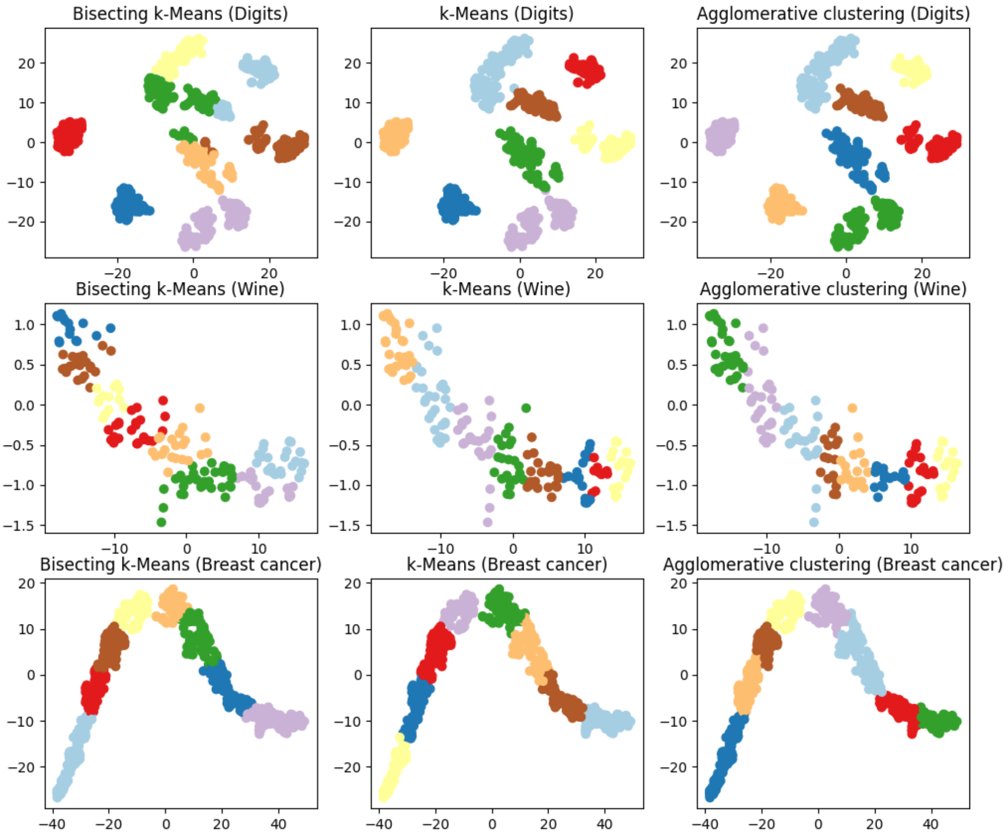
Figur 5 - Clustering with k = 5



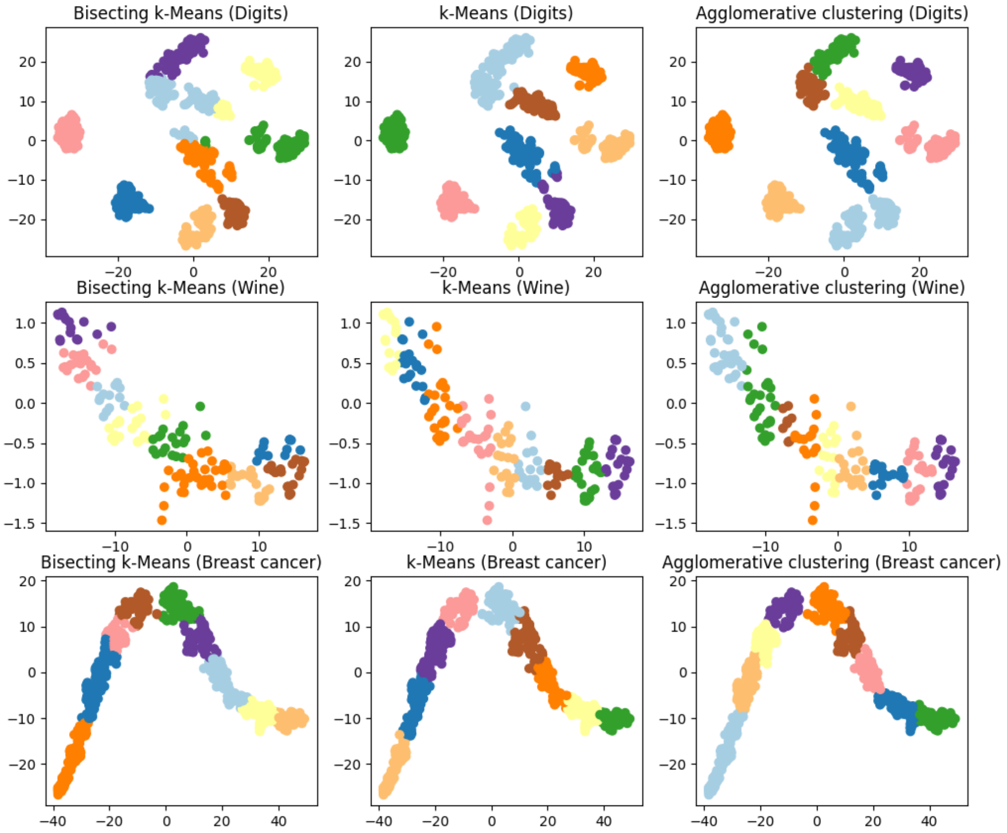
Figur 6 - Clustering with k = 6



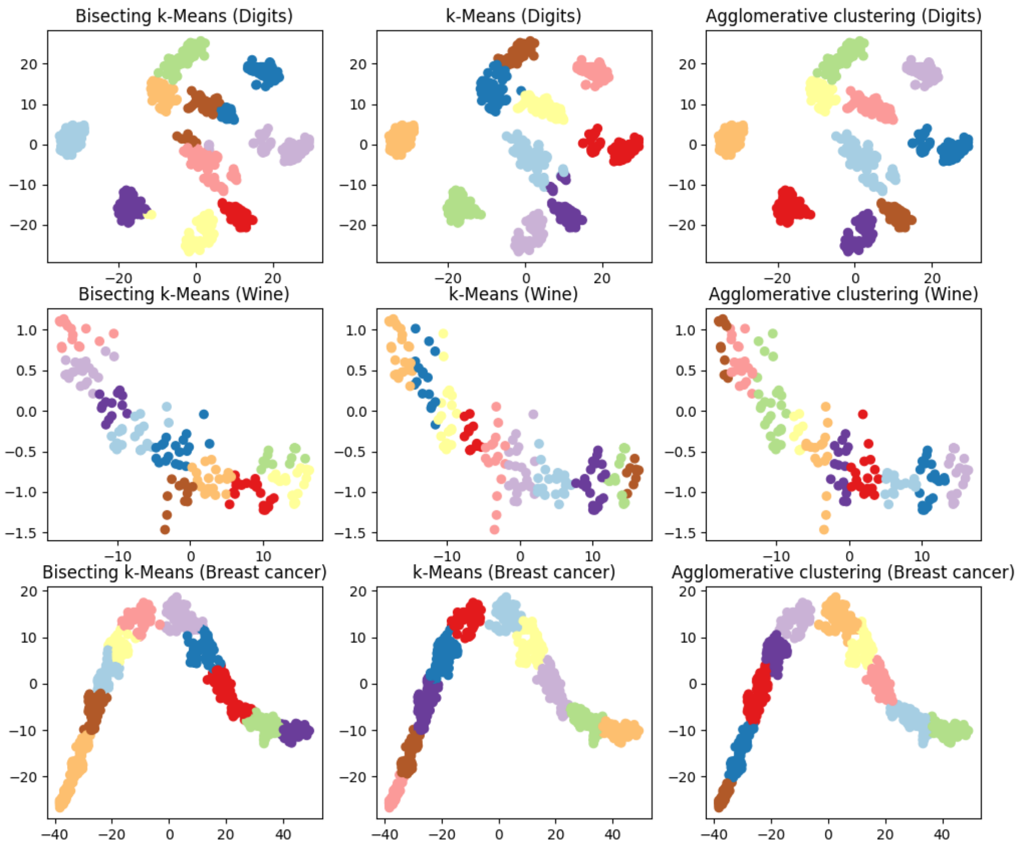
Figur 7 - Clustering with k = 7



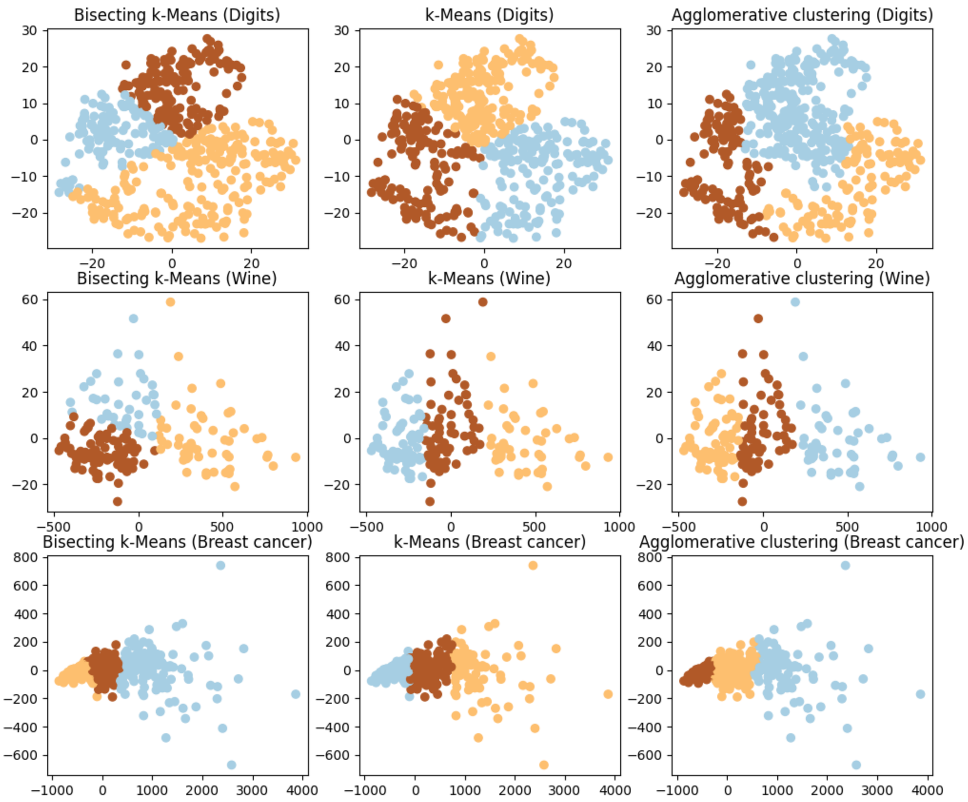
Figur 8 - Clustering with k = 8



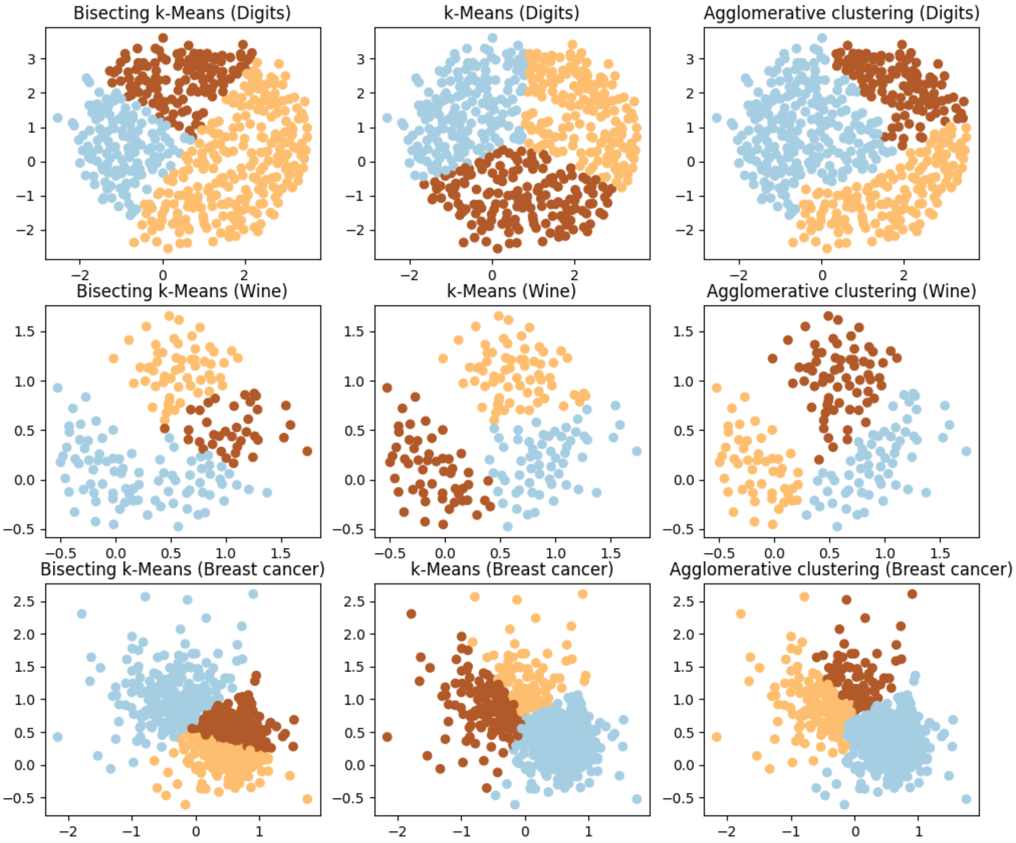
Figur 9 - Clustering with k = 9



Figur 10 - Clustering with k = 10



Figur 11 - Clustering with k = 3, using PCA as DR



Figur 12 - Clustering with k = 3, using Sammon mapping as DR