Medical diagnostics: Breast Cancer Wisconsin (Diagnostic) Project Part 2

Patrycja Cieślachowska,
Aleksandra Winiarska

Data Mining
dr Adam Zagdański

February 9, 2024

Contents

1 Introduction

Breast cancer is one of the most common and devastating types of cancer affecting women worldwide. In a medical context, a key challenge is to identify the exact type and stage of breast cancer, which provides the basis for effective treatment and prediction of clinical outcomes. For this purpose, various types of research and analyzes are carried out, and one of the tools that can help understand the division of breast cancer cases and identify groups of patients with similar characteristics is clustering analysis.

In this paper, we will focus on identifying natural groups or clusters in patient data that may reflect different breast cancer subtypes or other relevant patterns. By using clustering algorithms, we will try to distinguish groups of patients with similar characteristics, which may contribute to a better understanding of the heterogeneity of this disease and to adapt treatment strategies.

1.1 Data description

Analyzed dataset is entitled Breast Cancer Wisconsin (Diagnostic) and can be downloaded at archive.ics.uci.edu. The data contains 569 instances, 30 features used for diagnosting a patient (the mean, standard error, and "worst" or largest – mean of the three largest values – of these features were computed for each image) and a columns on diagnosis and patient ID. They have no missing or unusual values.

Ten real-valued features are computed for each cell nucleus. The unique names of columns in the dataset are:

1. ID number,
2. diagnosis (M – malignant (212), B – benign (357)),
3. radius (mean of distances from center to points on the perimeter),
4. texture (standard deviation of gray-scale values),
5. perimeter,
6. area,
7. smoothness (local variation in radius lengths),
8. compactness (perimeter^2/area $-$ 1.0),
9. concavity (severity of concave portions of the contour),
10. concave points (number of concave portions of the contour),
11. symmetry,
12. fractal dimension ("coastline approximation" -1).

All feature values are recoded with four significant digits.

2 Methods

2.1 Clustering and validation

To perform clustering and validation we used libraries: utils, cluster, factoextra, clusterSim, ppclust, fclust, NbClust, dbscan, ggplot2, e1071, clValid, mclust, scales, xtable and base. All main functions are listed below:

- fviz_nbclust(),
- kmeans(),
- fviz_cluster(),
- par(),
- pam(),
- fcm(),
- plotcluster(),
- ppclust2(),
- dbscan(),
- NbClust(),
- Mclust(),
- silhouette(),
- fviz_silhouette(),
- table(),
- matchClasses(),
- clValid(),
- summary(),
- cluster.Description(),
- ppclust2(),
- SIL.F(),
- PE(),
- PC(),
- MPC(),
- agnes(),
- cutree(),
- fviz_dend(),
- diana().

2.2 Dimensional reduction

To perform dimensional reduction we used libraries: utils, cluster, factoextra, corrplot, stats, graphics, xtable and base. All main functions are listed below:

- scale(),
- prcomp(),
- summary(),
- cumsum(),
- barplot(),
- abline(),
- plot(),
- get_eigenvalue(),
- sum(),
- data.frame(),
- xtable(),
- fviz_eig(),
- get_pca_var(),
- corrplot(),
- cor(),
- fviz_contrib(),
- fviz_pca_var(),
- fviz_pca_ind(),
- fviz_pca_biplot().

2.3 Classification

To perform classification we used the same libraries and functions as in the first part of the project [3].

3 Clustering

3.1 Clustering Algorithms

In our study, we used several different clustering algorithms, including:

- partitioning methods:
  - **K-means**,
  - **PAM** (**P**artitioning **A**round **M**edoids),
  - **Fuzzy C-means**,
- hierarchical methods:
  - **AGNES** (**Ag**glomerative **Nes**ting),
  - **DIANA** (**Di**visive **Ana**lysis).

By applying these various clustering algorithms, we will try to find optimal partitions of the data that will help in understanding the similarities and differences between patients diagnosed with breast cancer.

3.2 Optimal number of groups

The key stage in clustering analysis is the selection of the optimal number of clusters that best reflects the structure of the data and enables their effective segmentation. To determine number of clusters, we used Elbow and Silhouette methods for K-means, PAM and hierarchical algorithms.

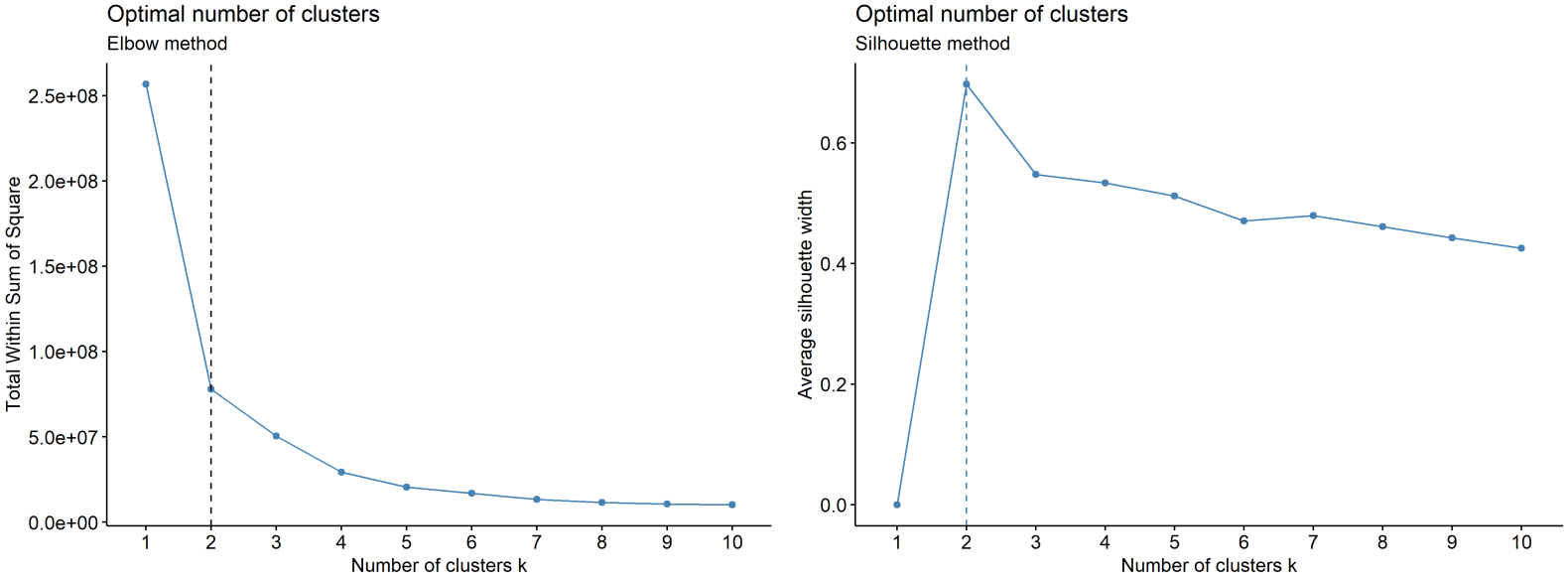

Figure 1: The optimal number of clusters for the elbow method (left) and silhouette method (right) for the K-means algorithm.

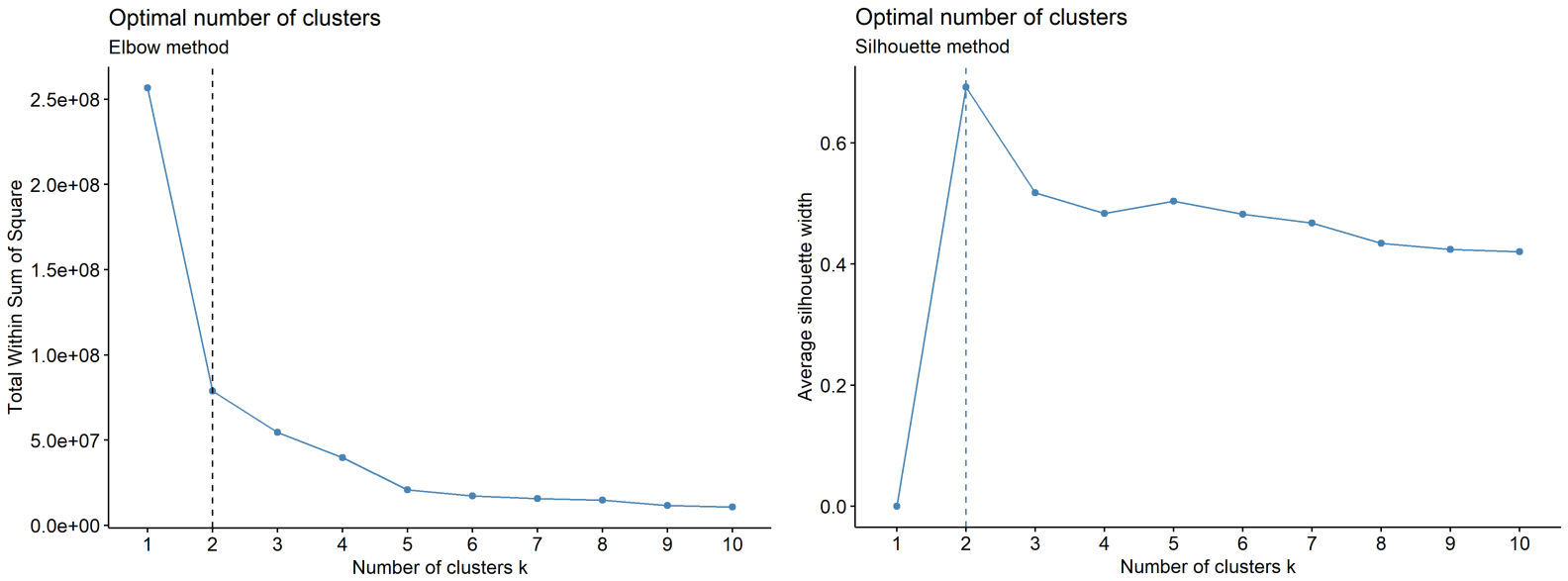

Figure 2: The optimal number of clusters for the elbow method (left) and silhouette method (right) for the PAM algorithm.

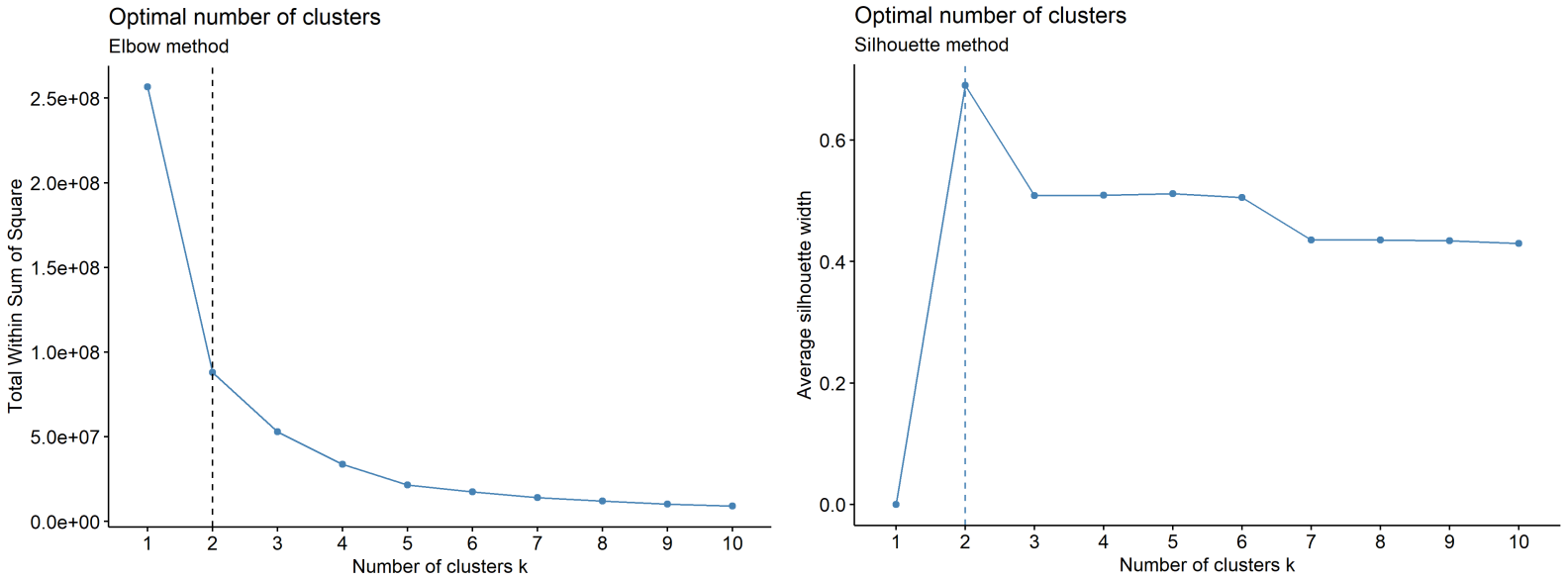

Figure 3: The optimal number of clusters for the elbow method (left) and silhouette method (right) for hierarchical methods.

Based on Figures 1, 2 and 3, it can be seen that both approaches: the Elbow method and the Eilhouette method for each of the algorithms used indicate the optimal division of data into 2 groups.

Additionally, we used the NbClust function, which is a tool used to automatically assess the optimal number of clusters in clustering analysis. This function uses various metrics and statistical methods to indicate the number of clusters that best reflects the structure of the data.

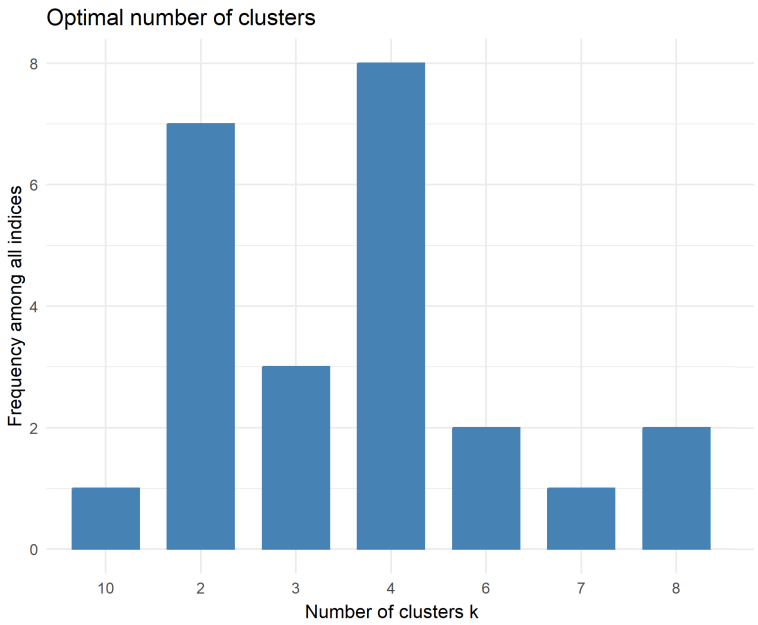

Figure 4: The optimal number of clusters for the NbClust function for complete aggregation method.

Figure 4 shows the numbers of indices suggesting different numbers of clusters as optimal. Graphical index charts are presented in appendix [4]. Based on this data, it can be concluded that most indices suggest the selection of 4 clusters when using the complete aggregation method. The NbClust function was similarly used for K-means, obtaining the following best results among all indices:

- 7 suggested 2 as the best number of clusters,
- 5 suggested 3 as the best number of clusters,
- 4 suggested 4 as the best number of clusters.

According to the majority rule, the best number of clusters is 2.

After detailed consideration and in the context of our goals, we decided to conduct further analysis using 2, 3 and 4 clusters. However, in this paper we will show graphical results only for 2 clusters. Graphical results for 3 and 4 clusters are available in appendix [4].

3.3 K-means

Figure 5 shows two clusters that were obtained by the K-means method.

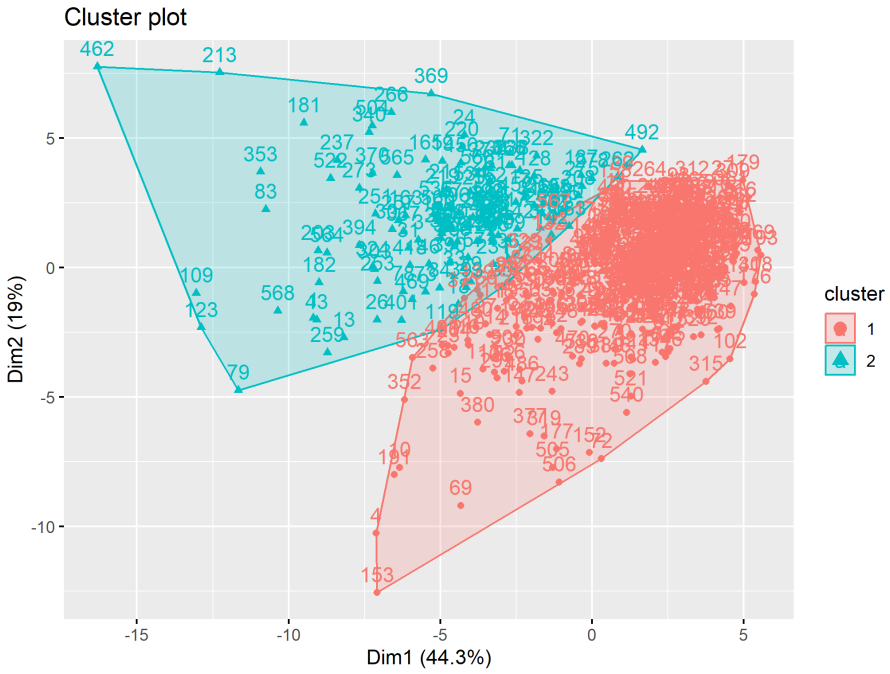

Figure 5: Clusters obtained using the 2-means method.

3.3.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for two clusters obtained using the K-means algorithm (Tables 1 and 2), it can be seen that in cluster 1 the average values of features such as radius, texture, perimeter and area are greater than in cluster 2. This means that these clusters show differences in feature values, which may suggest the existence of two different groups of objects. Also, the values of standard deviation (sd) and mean absolute deviation (mad) for most features are higher in cluster 1 than in cluster 2. This means that the data in cluster 1 are more dispersed around their mean values, which may indicate greater internal variation in this cluster. The median for most features in both clusters is close to the mean, which suggests that the distributions of features in both clusters are relatively symmetric. Therefore, the clusters obtained using the 2-means algorithm differ in terms of feature values and their dispersion. These results may be useful in further interpretation of data and identification of characteristic features of individual clusters.

	mean	sd	median	mad
Radius1	19.3799	2.4178	19.18	1.9719
Texture1	21.6946	3.9327	21.38	3.6768
Perimeter1	128.2313	16.9518	127.2	13.9364
Area1	1185.9298	312.3126	1145	229.803
Smoothness1	0.1013	0.0117	0.1003	0.0122
Compactness1	0.1486	0.0553	0.1336	0.0437
Concavity1	0.1769	0.0786	0.1626	0.0735
Concave_points1	0.1007	0.034	0.0946	0.029
Symmetry1	0.1915	0.0285	0.1875	0.0225
Fractal_dimension1	0.0606	0.0067	0.0602	0.0065
Radius2	0.7428	0.3537	0.6874	0.2423
Texture2	1.2225	0.5179	1.077	0.4077
Perimeter2	5.2506	2.7519	4.655	1.8636
Area2	95.6782	67.4767	81.89	34.8559
Smoothness2	0.0066	0.0025	0.0062	0.0018
Compactness2	0.0322	0.0173	0.0282	0.0136
Concavity2	0.0424	0.0206	0.0383	0.0165
Concave_points2	0.0157	0.0055	0.0147	0.0043
Symmetry2	0.0203	0.0095	0.0183	0.0061
Fractal_dimension2	0.004	0.0019	0.0037	0.0016
Radius3	23.7095	3.2993	23.15	3.2024
Texture3	28.9127	5.3829	28.18	4.8629
Perimeter3	158.4962	23.8009	152.5	19.7186
Area3	1753.0229	526.1722	1645	446.2626
Smoothness3	0.1404	0.0188	0.141	0.0187
Compactness3	0.3578	0.1486	0.3458	0.1207
Concavity3	0.4493	0.1679	0.4098	0.1491
Concave_points3	0.1924	0.0435	0.1899	0.0406
Symmetry3	0.3119	0.0657	0.306	0.055
Fractal_dimension3	0.0862	0.0165	0.0837	0.0127

Table 1: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for 2-means algorithm.

	mean	sd	median	mad
Radius1	12.5563	1.9127	12.6	1.8829
Texture1	18.5704	4.1462	18.15	3.8696
Perimeter1	81.1235	13.0335	81.32	12.5131
Area1	496.0619	148.7667	487.4	147.2963
Smoothness1	0.0949	0.0144	0.0941	0.0144
Compactness1	0.0911	0.0442	0.0795	0.0383
Concavity1	0.0624	0.0583	0.0433	0.0368
Concave_points1	0.0334	0.0239	0.0274	0.0183
Symmetry1	0.1781	0.0263	0.1744	0.0244
Fractal_dimension1	0.0635	0.007	0.0621	0.0057
Radius2	0.3042	0.1355	0.2734	0.1017
Texture2	1.2152	0.5619	1.1095	0.4787
Perimeter2	2.1529	0.92	1.985	0.7621
Area2	23.7853	11.943	20.74	8.4212
Smoothness2	0.0072	0.0031	0.0065	0.0023
Compactness2	0.0235	0.0176	0.0179	0.0107
Concavity2	0.0287	0.0319	0.0204	0.0149
Concave_points2	0.0106	0.0059	0.0096	0.0046
Symmetry2	0.0206	0.0079	0.0188	0.0058
Fractal_dimension2	0.0037	0.0028	0.003	0.0015
Radius3	14.0439	2.3594	13.825	2.3351
Texture3	24.7095	6.0336	24.03	6.0416
Perimeter3	91.9375	16.6383	90.01	16.5162
Area3	619.6479	206.3226	588.15	200.5216
Smoothness3	0.13	0.0234	0.1291	0.0219
Compactness3	0.2233	0.1464	0.1852	0.1036
Concavity3	0.2192	0.1896	0.172	0.1413
Concave_points3	0.0913	0.0519	0.0828	0.0413
Symmetry3	0.2836	0.0592	0.2764	0.0471
Fractal_dimension3	0.0833	0.0185	0.0787	0.0137

Table 2: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for 2-means algorithm.

3.3.2 Validation

Dispersion

Figure 6 shows comparision of the within-cluster and beetwen-cluster dispersion for 2-means algorithm. It is easy to see that as the number of clusters increases, the value of whithin-cluster dispersion increases. Therefore clustering using 2 clusters will result in having more compact clusters than in case of choosing greater number of groups. Between-cluster dispersion also suggest using smaller number of clusters as it indicates more heterogeneous data points among different groups.

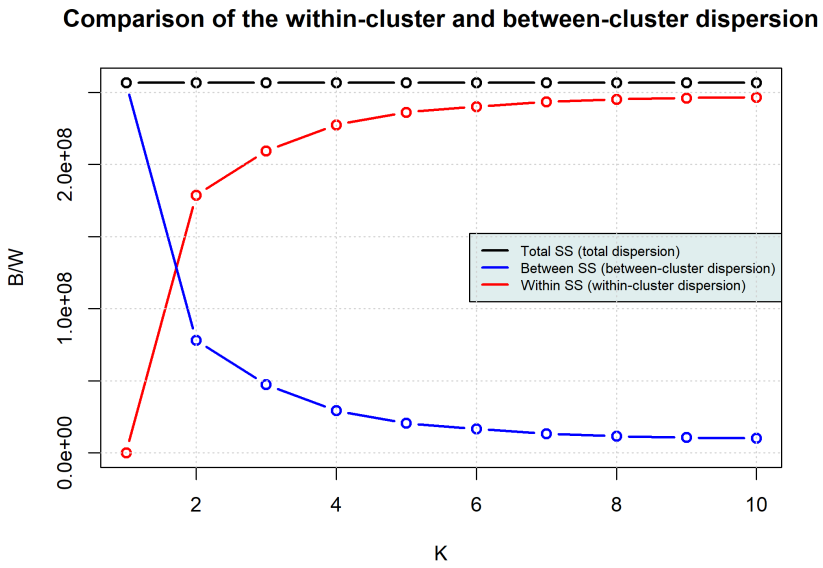

Figure 6: Comparision of the within-cluster and beetwen-cluster dispersion for 2-means algorithm.

Silhouette index

Figure 7 shows Silhouette index plot for 2-means algorithm for both clusters. The average value is 0.7, which indicates quite good clustering and will be later compared with clustering using 3 and 4 clusters.

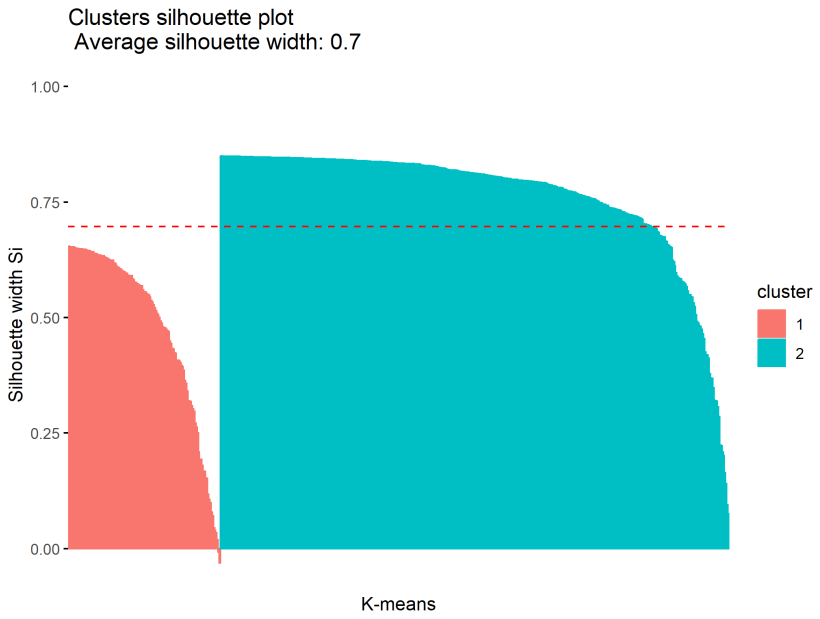

Figure 7: Silhouette index plot for 2-means algorithm.

Cluster labels vs actual labels

Table 3 shows the true labels (B – benign, M – malignant) in comparison with cluster to which they where assigned for division into 2, 3 and 4 clusters using the K-means algorithm. Analysis of the results from this tables for different sizes of clusters (2, 3 and 4) show that by increasing the number of clusters, we observe more diverse data divisions. Each additional cluster appears to identify new patterns or groups of data. However, the division into clusters do not always perfectly reflect the real labels. This suggests that the K-means algorithm is identifying different patterns in the data that do not always match the actual labels.

	Data labels	
	B	M
Cluster 1	1	130
Cluster 2	356	82

	Data labels	
	B	M
Cluster 1	40	105
Cluster 2	0	84
Cluster 3	317	23

	Data labels	
	B	M
Cluster 1	1	100
Cluster 2	262	6
Cluster 3	0	19
Cluster 4	94	87

Table 3: Table of cluster and actual labels for 2, 3 and 4 clusters for K-means algorithm.

Partition agreement

The results presented in Table 4 show the percentage of matched cases in pairs for different number of clusters (2, 3 and 4) obtained using the K-means algorithm. These results show that the three clusters achieve the highest pairwise case matching percentage (88.93%), suggesting the best fit of the model to the data.

Number of clusters	Cases in matched pairs
2	85.41%
3	88.93%
4	83.48%

Table 4: Table of cases in matched pairs for 2, 3 and 4 clusters for K-means algorithm.

Internal validation

Based on the analysis of internal validation indices (Table 5), the optimal number of clusters for the K-means algorithm seems to be two. In this case, we obtained the highest values of Dunn and Silhouette index and the lowest Connectivity, which suggests the best cluster structure and assignment of points to clusters.

Number of clusters	Connectivity	Dunn	Silhouette
2	4.2135	0.0173	0.6973
3	24.2012	0.0169	0.6696
4	35.3306	0.0053	0.5334

Table 5: Table of internal validation indices for 2, 3 and 4 clusters for K-means algorithm.

Stability indices

Stability indices for different numbers of clusters (2, 3 and 4) obtained using the K-means algorithm are presented in Table 6. Analysis of these indices suggests choosing division into 2 or 4 groups. Inidices APN and ADM shows that division into 2 clusters will benefit in more stable results. On the other hand, AD and FOM indicate 4 clusters as the best one.

Number of clusters	APN	AD	ADM	FOM
2	0.0015	377.9519	1.8766	20.1277
3	0.0032	325.1325	2.3507	16.9976
4	0.0108	325.1325	7.1683	14.3006

Table 6: Table of Stability indices for 2, 3 and 4 clusters for K-means algorithm.

3.4 PAM

Figure 8 shows two clusters that were obtained by the PAM method.

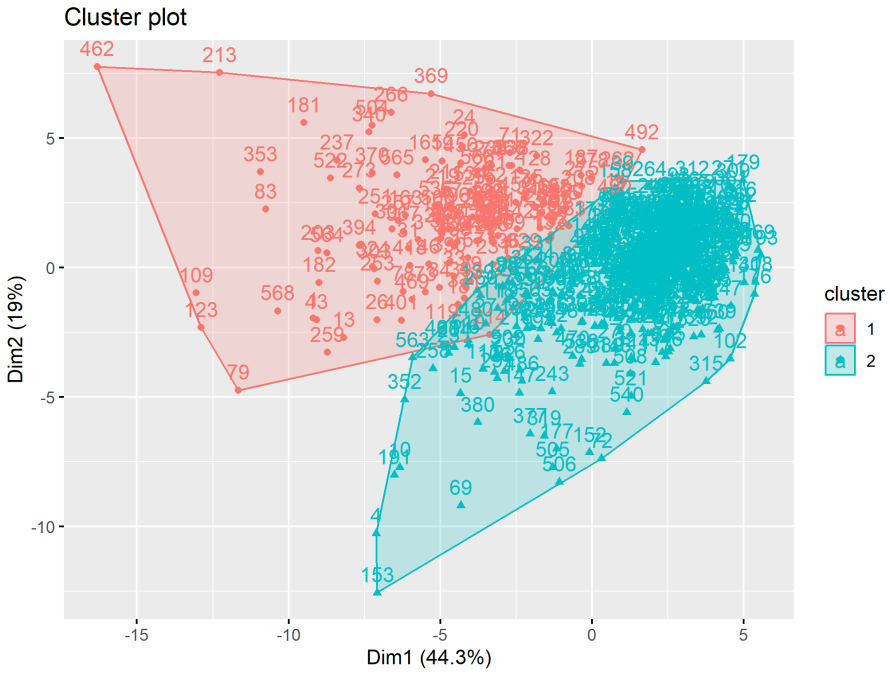

Figure 8: Two clusters obtained using the PAM method.

3.4.1 Statistics of clusters (for 2 clusters)

Statistics for the two clusters obtained using the PAM algorithm are presented in Tables 7 and 8. By comparing these results with previous results from the K-means algorithm (Tables 1 and 2), very similar conclusions can be drawn. Therefore, the analysis of data from the PAM algorithm confirms similar trends in feature values and their dispersion as ones obtained from the K-means algorithm. Both algorithms suggest the existence of two different groups of objects, differing in feature values and their distributions.

	mean	sd	median	mad
Radius1	19.2068	2.4531	19.07	2.1794
Texture1	21.756	3.8813	21.43	3.4841
Perimeter1	127.0122	17.2049	125.5	14.233
Area1	1165.6914	314.3443	1130	247.5942
Smoothness1	0.101	0.0118	0.1003	0.0128
Compactness1	0.1464	0.0548	0.1318	0.0421
Concavity1	0.1733	0.0788	0.1572	0.0704
Concave_points1	0.0984	0.0347	0.0918	0.0296
Symmetry1	0.1909	0.0286	0.1875	0.0228
Fractal_dimension1	0.0605	0.0066	0.06	0.0057
Radius2	0.7244	0.3518	0.6422	0.2623
Texture2	1.2155	0.509	1.077	0.3929
Perimeter2	5.1183	2.7279	4.542	1.8948
Area2	92.7071	66.6325	80.6	38.2511
Smoothness2	0.0067	0.0032	0.0061	0.0018
Compactness2	0.0321	0.0177	0.0279	0.0136
Concavity2	0.0426	0.0223	0.0374	0.0153
Concave_points2	0.0156	0.0059	0.0146	0.0043
Symmetry2	0.0202	0.0094	0.018	0.0057
Fractal_dimension2	0.004	0.002	0.0037	0.0016
Radius3	23.4442	3.38	22.93	3.2469
Texture3	28.9876	5.2798	28.22	4.774
Perimeter3	156.654	24.2935	152.2	21.0529
Area3	1716.8489	531.5282	1610	456.6408
Smoothness3	0.1399	0.0189	0.1408	0.019
Compactness3	0.3533	0.1469	0.3391	0.1265
Concavity3	0.4433	0.1689	0.3995	0.1543
Concave_points3	0.1893	0.0451	0.186	0.0427
Symmetry3	0.3124	0.0685	0.306	0.0559
Fractal_dimension3	0.0858	0.0162	0.0833	0.0127

Table 7: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for PAM algorithm.

	mean	sd	median	mad
Radius1	12.4853	1.856	12.515	1.8458
Texture1	18.4924	4.1278	18.045	3.7806
Perimeter1	80.6411	12.6498	81.12	12.0239
Area1	489.7693	142.5341	481	142.7744
Smoothness1	0.0949	0.0144	0.094	0.0144
Compactness1	0.0907	0.0444	0.0789	0.0379
Concavity1	0.0615	0.0579	0.0429	0.0361
Concave_points1	0.0329	0.0236	0.0272	0.0178
Symmetry1	0.178	0.0263	0.1743	0.0241
Fractal_dimension1	0.0635	0.0071	0.0622	0.0057
Radius2	0.302	0.1353	0.2716	0.0995
Texture2	1.2173	0.5653	1.1095	0.4837
Perimeter2	2.138	0.9186	1.9725	0.7635
Area2	23.4082	11.6551	20.66	8.1469
Smoothness2	0.0072	0.0029	0.0065	0.0023
Compactness2	0.0234	0.0175	0.0178	0.0106
Concavity2	0.0284	0.0316	0.0202	0.0149
Concave_points2	0.0106	0.0057	0.0096	0.0046
Symmetry2	0.0207	0.0079	0.0188	0.0058
Fractal_dimension2	0.0037	0.0028	0.003	0.0015
Radius3	13.9498	2.2763	13.755	2.3203
Texture3	24.6071	6.0292	23.74	6.0416
Perimeter3	91.2947	16.0979	89.44	16.0269
Area3	610.2553	196.1608	583.05	191.1813
Smoothness3	0.1299	0.0235	0.1291	0.0219
Compactness3	0.2223	0.1471	0.1843	0.1031
Concavity3	0.2169	0.1895	0.1694	0.1408
Concave_points3	0.0905	0.0517	0.0823	0.0418
Symmetry3	0.2829	0.0578	0.2758	0.0464
Fractal_dimension3	0.0833	0.0186	0.0787	0.0139

Table 8: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for PAM algorithm.

3.4.2 Validation

Silhouette index

Figure 9 shows Silhouette index plot for PAM algorithm for both clusters. The average value is 0.69, which indicates quite good clustering and will be later compared with clustering using 3 and 4 clusters.

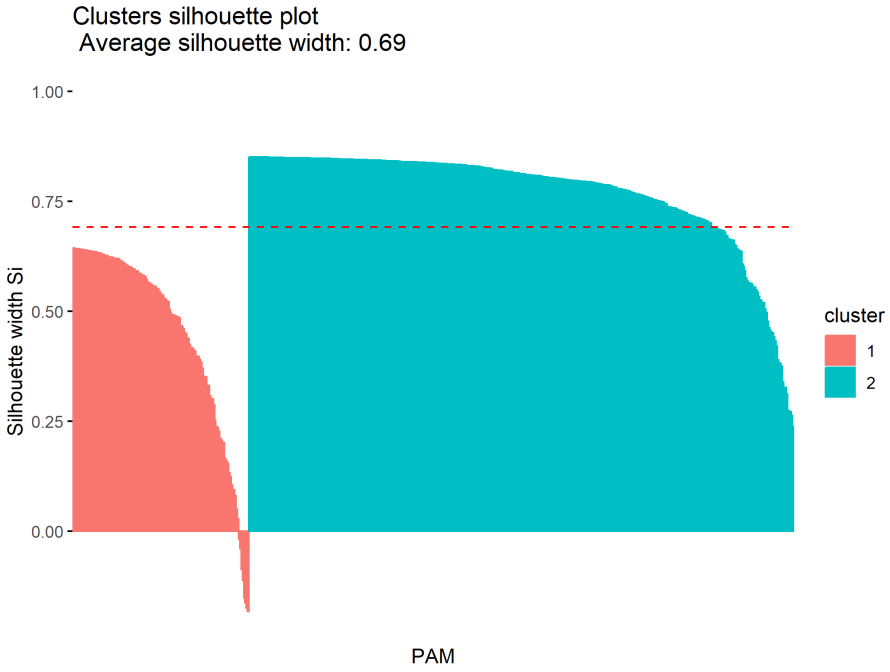

Figure 9: Silhouette index plot for PAM algorithm for 2 clusters.

Cluster labels vs actual labels

Table 9 shows the true labels (B - benign, M - malignant) in comparison with clusters to which they where assigned using PAM algorithm. By comparing these results with the previous results of the K-means algorithm (Table 3), analogous conclusions can be drawn. That being said, increasing the number of clusters leads to more diverse data division.

	Data labels	
	B	M
Cluster 1	1	138
Cluster 2	356	74

	Data labels	
	B	M
Cluster 1	0	112
Cluster 2	262	6
Cluster 3	95	94

	Data labels	
	B	M
Cluster 1	0	112
Cluster 2	182	9
Cluster 3	72	91
Cluster 4	103	0

Table 9: Table of cluster and actual labels for 2, 3 and 4 clusters for PAM algorithm.

Partition agreement

The results presented in Table 10 show the percentage of matched cases in pairs for different number of clusters (2, 3 and 4) obtained using the PAM algorithm. These results show that the two clusters achieve the highest pairwise case matching percentage (86.82%), suggesting the best fit of the model to the data.

Number of clusters	Cases in matched pairs
2	86.82%
3	82.43%
4	85.76%

Table 10: Table of cases in matched pairs for 2, 3 and 4 clusters for PAM algorithm.

Internal validation

Based on the analysis of internal validation indices (Table 11), the optimal number of clusters for the PAM algorithm seems to be two, similar to K-means. In this case, we again obtained the highest values of Dunn and Silhouette indices and the lowest Connectivity.

Number of clusters	Connectivity	Dunn	Silhouette
2	10.0155	0.0156	0.6921
3	14.1048	0.0068	0.5175
4	40.8139	0.0033	0.4834

Table 11: Table of internal validation indices for 2, 3 and 4 clusters for PAM algorithm.

Stability indices

Stability indices for different numbers of clusters (2, 3 and 4) obtained using the PAM algorithm are presented in Table 12. By analysing these indices we can draw the same conclusions as for K-means.

Number of clusters	APN	AD	ADM	FOM
2	0.0024	375.9819	2.0586	20.3935
3	0.0079	285.3756	4.4856	17.4586
4	0.0148	237.0030	7.3319	16.0768

Table 12: Table of Stability indices for 2, 3 and 4 clusters for PAM algorithm.

3.5 Fuzzy c-means

Figure 10 shows two clusters that were obtained by the Fuzzy c-means method for m=2. Analogous plots for m equal to 1.5 and 2.5 are available in the appendix [4].

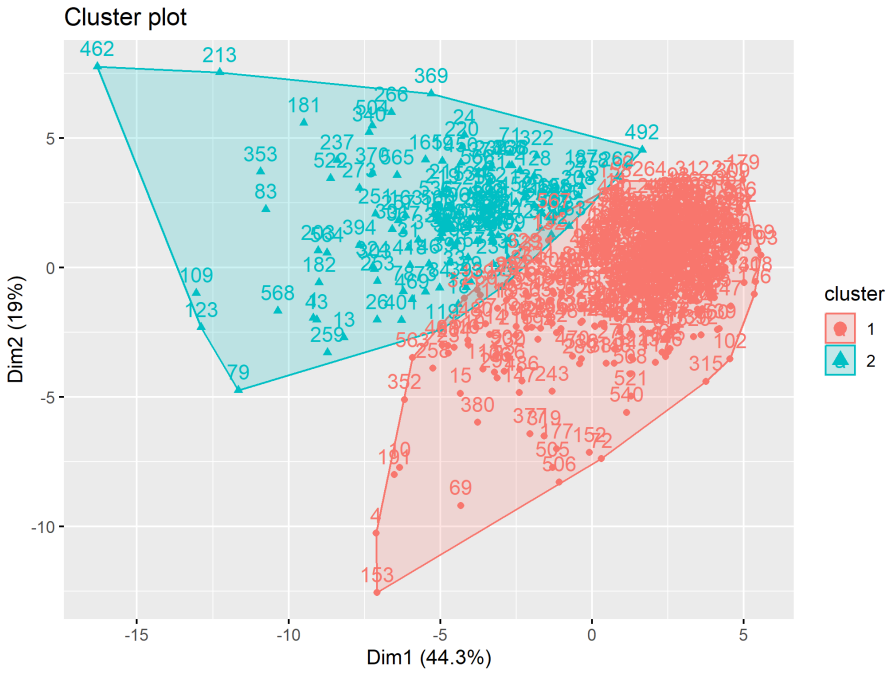

Figure 10: Two clusters obtained using the fuzzy c-means method for m=2.

3.5.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for the two hard clusters obtained by using the Fuzzy c-means algorithm (Tables 13 and 14) with previously analyzed K-means (Tables 1 and 2), it can be seen that both algorithms give identical values. Therefore, similar comclusions as for K-means can be drawn. What is worth mentioning is that we only consider hard clustering given by Fuzzy c-means, which means we assign objects to one of two clusters, by choosing the highest probability of belonging to the cluster.

	mean	sd	median	mad
Radius1	19.3799	2.4178	19.18	1.9719
Texture1	21.6946	3.9327	21.38	3.6768
Perimeter1	128.2313	16.9518	127.2	13.9364
Area1	1185.9298	312.3126	1145	229.803
Smoothness1	0.1013	0.0117	0.1003	0.0122
Compactness1	0.1486	0.0553	0.1336	0.0437
Concavity1	0.1769	0.0786	0.1626	0.0735
Concave_points1	0.1007	0.034	0.0946	0.029
Symmetry1	0.1915	0.0285	0.1875	0.0225
Fractal_dimension1	0.0606	0.0067	0.0602	0.0065
Radius2	0.7428	0.3537	0.6874	0.2423
Texture2	1.2225	0.5179	1.077	0.4077
Perimeter2	5.2506	2.7519	4.655	1.8636
Area2	95.6782	67.4767	81.89	34.8559
Smoothness2	0.0066	0.0025	0.0062	0.0018
Compactness2	0.0322	0.0173	0.0282	0.0136
Concavity2	0.0424	0.0206	0.0383	0.0165
Concave_points2	0.0157	0.0055	0.0147	0.0043
Symmetry2	0.0203	0.0095	0.0183	0.0061
Fractal_dimension2	0.004	0.0019	0.0037	0.0016
Radius3	23.7095	3.2993	23.15	3.2024
Texture3	28.9127	5.3829	28.18	4.8629
Perimeter3	158.4962	23.8009	152.5	19.7186
Area3	1753.0229	526.1722	1645	446.2626
Smoothness3	0.1404	0.0188	0.141	0.0187
Compactness3	0.3578	0.1486	0.3458	0.1207
Concavity3	0.4493	0.1679	0.4098	0.1491
Concave_points3	0.1924	0.0435	0.1899	0.0406
Symmetry3	0.3119	0.0657	0.306	0.055
Fractal_dimension3	0.0862	0.0165	0.0837	0.0127

Table 13: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for Fuzzy c-means algorithm.

	mean	sd	median	mad
Radius1	12.5563	1.9127	12.6	1.8829
Texture1	18.5704	4.1462	18.15	3.8696
Perimeter1	81.1235	13.0335	81.32	12.5131
Area1	496.0619	148.7667	487.4	147.2963
Smoothness1	0.0949	0.0144	0.0941	0.0144
Compactness1	0.0911	0.0442	0.0795	0.0383
Concavity1	0.0624	0.0583	0.0433	0.0368
Concave_points1	0.0334	0.0239	0.0274	0.0183
Symmetry1	0.1781	0.0263	0.1744	0.0244
Fractal_dimension1	0.0635	0.007	0.0621	0.0057
Radius2	0.3042	0.1355	0.2734	0.1017
Texture2	1.2152	0.5619	1.1095	0.4787
Perimeter2	2.1529	0.92	1.985	0.7621
Area2	23.7853	11.943	20.74	8.4212
Smoothness2	0.0072	0.0031	0.0065	0.0023
Compactness2	0.0235	0.0176	0.0179	0.0107
Concavity2	0.0287	0.0319	0.0204	0.0149
Concave_points2	0.0106	0.0059	0.0096	0.0046
Symmetry2	0.0206	0.0079	0.0188	0.0058
Fractal_dimension2	0.0037	0.0028	0.003	0.0015
Radius3	14.0439	2.3594	13.825	2.3351
Texture3	24.7095	6.0336	24.03	6.0416
Perimeter3	91.9375	16.6383	90.01	16.5162
Area3	619.6479	206.3226	588.15	200.5216
Smoothness3	0.13	0.0234	0.1291	0.0219
Compactness3	0.2233	0.1464	0.1852	0.1036
Concavity3	0.2192	0.1896	0.172	0.1413
Concave_points3	0.0913	0.0519	0.0828	0.0413
Symmetry3	0.2836	0.0592	0.2764	0.0471
Fractal_dimension3	0.0833	0.0185	0.0787	0.0137

Table 14: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for Fuzzy c-means algorithm.

3.5.2 Validation

Fuzzy silhouette index

Table 15 shows the Fuzzy Silhouette index values for 2, 3 and 4 clusters. Based on it, it can be noticed that for 2 clusters a high Fuzzy Silhouette index result of 88.79% was obtained. For 3 clusters this result dropped to 79.77%, and for 4 clusters it decreased even more. Values of Fuzzy Silhouette index suggest that division into two clusters reflects the structure of the data better than division into a grater number of clusters.

Number of clusters	Fuzzy silhouette index
2	88.79%
3	79.77%
4	78.50%

Table 15: Table of Fuzzy silhouette index for 2, 3 and 4 clusters for Fuzzy c-means method.

Cluster labels vs actual labels

Table 16 shows the true labels in comparison with hard cluster to which they where assigned using the Fuzzy c-means algorithm. On their basis, similar conclusions can be drawn as for the previous two methods.

	Data labels	
	B	M
Cluster 1	1	130
Cluster 2	356	82

	Data labels	
	B	M
Cluster 1	0	84
Cluster 2	321	29
Cluster 3	36	99

	Data labels	
	B	M
Cluster 1	262	6
Cluster 2	94	87
Cluster 3	0	19
Cluster 4	1	100

Table 16: Table of cluster and actual labels for 2, 3 and 4 clusters for fuzzy c-means algorithm.

Partition agreement

The results presented in Table 17 show the percentage of matched cases in pairs for different number of clusters (2, 3 and 4) obtained by using the Fuzzy c-means algorithm. This time we consider fuzzy clusters, not hard as before (hard is used only for cases in matched pairs). Similar to K-means, division into three clusters achieve the highest pairwise case matching percentage (88.58%). Clustering quality metrics such as partition coefficient, partition entropy and modified partition coefficient did not clearly indicate the best number of divisions into clusters.

Number of clusters	Cases in matched pairs	Part. Coeff.	Part. Entropy	Mod. Part. Coeff.
2	85.41%	89.69%	18.09%	79.39%
3	88.58%	79.19%	37.42%	79.39%
4	83.48%	76.94%	43.04%	79.39%

Table 17: Table of cases in matched pairs, partition coefficient, partition entropy and modified partition coefficient for 2, 3 and 4 clusters for Fuzzy c-means algorithm.

3.6 AGNES with average linkage

Figure 11 shows two clusters that were obtained by the AGNES with average linkage method. Colors of the observations (at the bottom of the dendrogram) indicate the true labels.

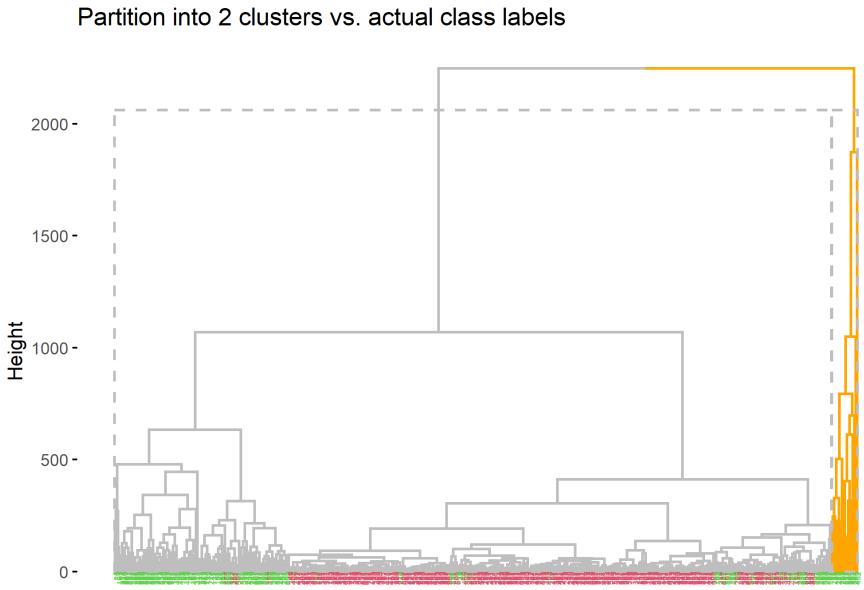

Figure 11: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with average linkage.

3.6.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for two clusters obtained using the AGNES algorithm with average linkage (Tables 18 and 19) it can be seen that for most features, such as radius, texture, perimeter or area, significant differences between clusters can be observed. The values of standard deviation and mean absolute deviation also differ between clusters. Higher values of these parameters indicate greater diversity of data in the cluster. Therefore, the algorithm divided the data into two clusters that show significant differences in feature values.

	mean	sd	median	mad
Radius1	13.7896	3.0686	13.2	2.6242
Texture1	19.156	4.2513	18.75	4.1661
Perimeter1	89.626	21.0797	85.48	17.8653
Area1	615.825	283.5074	538.4	211.567
Smoothness1	0.0961	0.014	0.0958	0.014
Compactness1	0.1018	0.0507	0.09	0.0454
Concavity1	0.0831	0.0727	0.0586	0.0573
Concave_points1	0.0457	0.0348	0.0325	0.0285
Symmetry1	0.1809	0.0273	0.1791	0.0254
Fractal_dimension1	0.0629	0.0071	0.0617	0.0061
Radius2	0.3777	0.2154	0.3141	0.1489
Texture2	1.2149	0.5529	1.108	0.4685
Perimeter2	2.6656	1.5495	2.23	1.069
Area2	35.0293	28.1693	23.92	12.8393
Smoothness2	0.007	0.0029	0.0064	0.0022
Compactness2	0.0251	0.0176	0.0202	0.0129
Concavity2	0.0312	0.03	0.0249	0.018
Concave_points2	0.0116	0.0062	0.0107	0.0051
Symmetry2	0.0206	0.0082	0.0187	0.0058
Fractal_dimension2	0.0038	0.0027	0.0031	0.0016
Radius3	15.7802	4.1418	14.8	3.3952
Texture3	25.5046	6.0536	25.22	6.2862
Perimeter3	103.8183	28.6476	96.66	24.5222
Area3	813.3239	444.0102	670.6	293.8513
Smoothness3	0.132	0.023	0.1312	0.0219
Compactness3	0.2493	0.1559	0.2089	0.1262
Concavity3	0.2632	0.2044	0.2151	0.1855
Concave_points3	0.1104	0.0626	0.0974	0.0627
Symmetry3	0.29	0.0623	0.2818	0.0511
Fractal_dimension3	0.084	0.0183	0.0799	0.0148

Table 18: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for AGNES – average linkage algorithm.

	mean	sd	median	mad
Radius1	23.396	2.4375	23.24	2.2387
Texture1	22.9575	4.1304	21.97	3.41
Perimeter1	156.285	17.9462	153.15	18.829
Area1	1727.2	359.9746	1683.5	286.8831
Smoothness1	0.1047	0.0149	0.1046	0.0145
Compactness1	0.1737	0.0623	0.1798	0.0807
Concavity1	0.2447	0.1052	0.2309	0.1428
Concave_points1	0.1362	0.0429	0.1441	0.0624
Symmetry1	0.1876	0.0308	0.18	0.0193
Fractal_dimension1	0.0593	0.0059	0.0572	0.0061
Radius2	1.1591	0.5831	1.0085	0.3002
Texture2	1.2694	0.5267	1.2165	0.4125
Perimeter2	8.3688	4.4636	7.3145	2.2587
Area2	186.035	125.7755	154.45	56.7094
Smoothness2	0.0072	0.0043	0.006	0.0016
Compactness2	0.0372	0.0233	0.0284	0.0171
Concavity2	0.05	0.03	0.0378	0.0204
Concave_points2	0.0164	0.0049	0.0154	0.0043
Symmetry2	0.0201	0.0103	0.0164	0.0046
Fractal_dimension2	0.0039	0.0019	0.0036	0.0017
Radius3	29.691	2.6532	29.545	2.2239
Texture3	30.4145	6.9195	29.475	5.2262
Perimeter3	201.77	18.4442	200.95	18.829
Area3	2726.85	534.7879	2588.5	404.7498
Smoothness3	0.1421	0.0153	0.1405	0.0164
Compactness3	0.3908	0.1377	0.4065	0.0847
Concavity3	0.5202	0.1718	0.508	0.1792
Concave_points3	0.2298	0.0425	0.2367	0.0503
Symmetry3	0.2914	0.0513	0.2844	0.0378
Fractal_dimension3	0.0822	0.0116	0.0815	0.0096

Table 19: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for AGNES – average linkage algorithm.

3.6.2 Validation

Silhouette index

Figure 12 shows Silhouette index plot for AGNES algorithm with average linkage for both clusters. The average value is 0.69, which indicates quite good clustering and will be later compared with clustering using 3 and 4 clusters. Worth noticing is that in this method we observe highly imbalanced cluster division – there are fewer objects in second cluster.

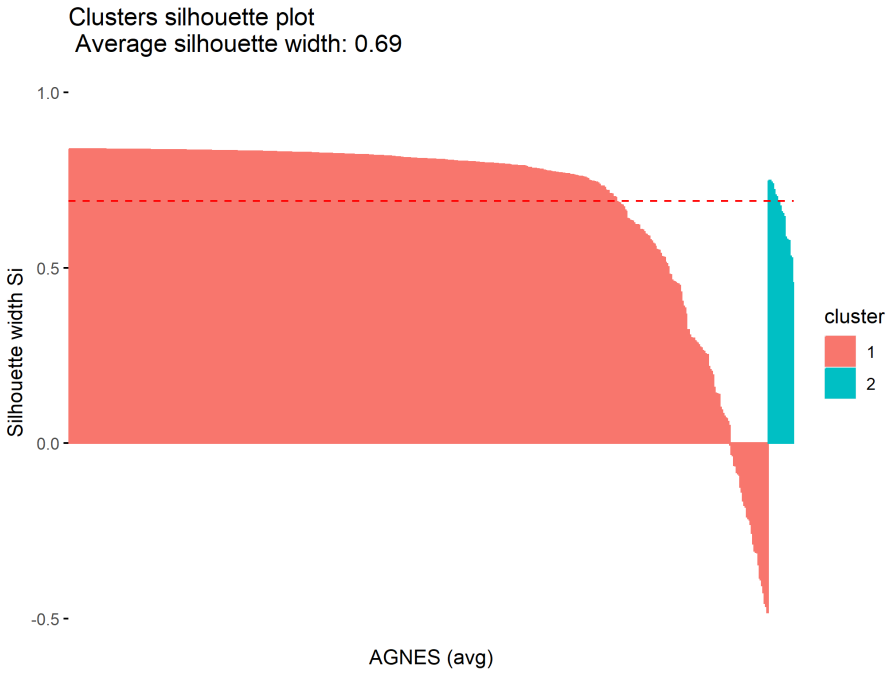

Figure 12: Silhouette index plot for AGNES method with average linkage for 2 clusters.

Cluster labels vs actual labels

Table 20 shows the assignment of true labels in comparison with cluster to which they where assigned using the AGNES algorithm with average linkage. It can be seen that for 2 clusters most of the observations are assigned to cluster 1, while cluster 2 contains only 20 observations (only malligant patients). This suggests a clear separation of the majority of observations from a small subset. Additionally, it can be seen that for a larger number of clusters, the algorithm tried to divide the clusters in more detail, but most of the observations still fall into one cluster. Therefore, in each case, one cluster contains the significant majority of observations, indicating a strong concentration of data in one cluster.

	Data labels	
	B	M
Cluster 1	357	192
Cluster 2	0	20

	Data labels	
	B	M
Cluster 1	357	192
Cluster 2	0	19
Cluster 3	0	1

	Data labels	
	B	M
Cluster 1	5	128
Cluster 2	352	64
Cluster 3	0	19
Cluster 4	0	1

Table 20: Table of cluster and actual labels for 2, 3 and 4 clusters for AGNES – average linkage algorithm.

Partition agreement
The results presented in Table 21 show the percentage of matched cases in pairs for different number of clusters (2, 3 and 4) obtained by using the AGNES with average linkage algorithm. Based on them, it can be concluded that for division into 2 or 3 clusers we obtain identical agreement, amounting to 66.26%. This means that in both cases, more than 2/3 of the cases match between the assigned clusters and the actual data labels. Increasing the number of clusters to 4 is associated with an increase in clustering agreement. This may suggest that a larger number of clusters allows for better adjustment to the data and more detailed detection of group differences.

Number of clusters	Cases in matched pairs
2	66.26%
3	66.26%
4	87.87%

Table 21: Table of cases in matched pairs for 2, 3 and 4 clusters for AGNES – average linkage algorithm.

Internal validation
Based on the analysis of internal validation indices (Table 22), the optimal number of clusters for the AGNES with average linkage algorithm seems to be two or three. For division into two clusters, we obtained the highest value of Silhouette index and the lowest Connectivity, and for three clusters the highest value of Dunn's index.

Number of clusters	Connectivity	Dunn	Silhouette
2	8.3500	0.0705	0.6909
3	11.2790	0.0747	0.6726
4	22.3663	0.0314	0.6574

Table 22: Table of internal validation indices for 2, 3 and 4 clusters for AGNES – average linkage algorithm.

Stability indices
Stability indices for different numbers of clusters (2, 3 and 4) obtained using the the AGNES with average linkage algorithm are presented in Table 23. Analysis of these indices suggests choosing division into 2 or 4 groups. Inidces APN and ADM shows that division into 2 clusters will benefit in more stable results. On the other hand, AD and FOM indicate 4 clusters as the best one.

Number of clusters	APN	AD	ADM	FOM
2	0.0006	580.2927	5.1017	31.4439
3	0.0100	573.6349	13.8864	23.2109
4	0.0105	323.9897	12.9460	16.6017

Table 23: Table of Stability indices for 2, 3 and 4 clusters for AGNES – average linkage algorithm.

3.7 AGNES with single linkage

Figure 13 shows two clusters that were obtained by the AGNES with single linkage method. Colors of the observations (at the bottom of the dendrogram) indicate the true labels.

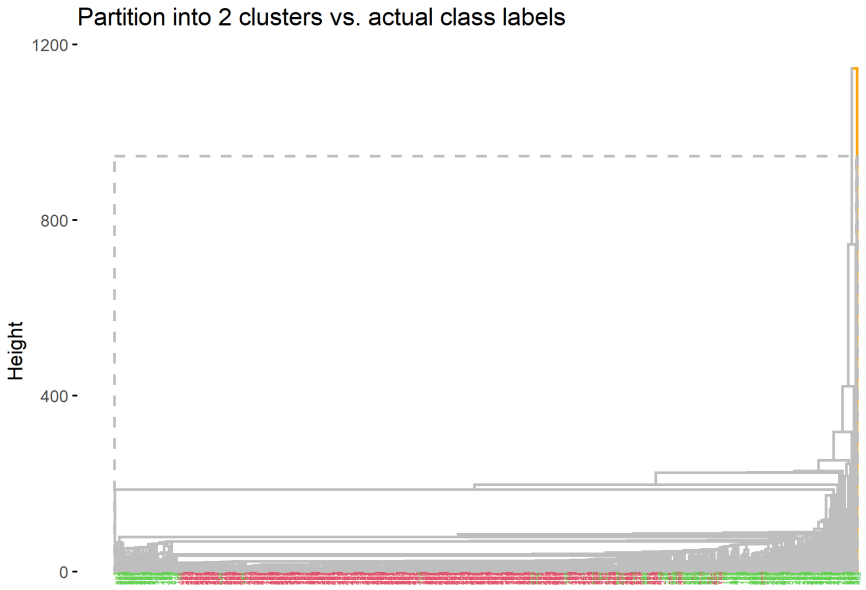

Figure 13: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with single linkage.

3.7.1 Statistics of clusters (for 2 clusters)

It makes no sense to compute statistics of clusters, as there is only 1 observation assigned to cluster 2.

3.7.2 Validation

Silhouette index

Figure 14 shows Silhouette index plot for AGNES algorithm with single linkage for both clusters. The average value is 0.8, which indicates good clustering and will be later compared with clustering using 3 and 4 clusters. Worth noticing is that in this method we observe highly imbalanced cluster division – there are hardly any objects in second cluster.

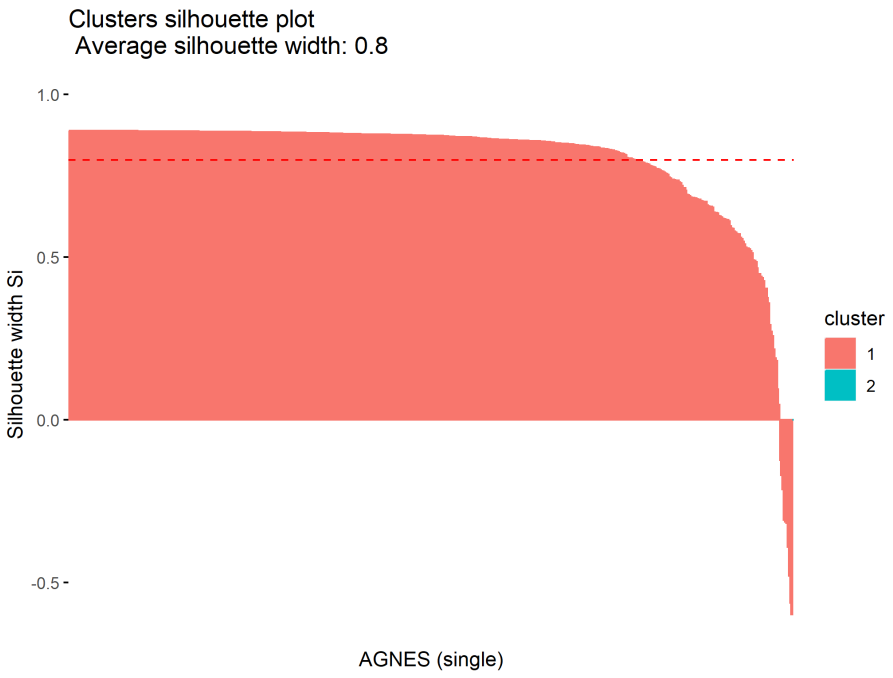

Figure 14: Silhouette index plot for AGNES method with single linkage for 2 clusters.

Cluster labels vs actual labels

Table 24 shows the assignment of data labels to clusters using the AGNES algorithm using single linkage. It can be seen that for 2 clusters most of the observations are assigned to cluster 1, while cluster 2 contains only one observation. Additionally, it can be seen that for a larger number of clusters, the algorithm tried to divide the clusters in more detail, but most of the observations still fall into cluster 1. Therefore, in each case, cluster 1 contains the significant majority of observations, indicating a strong concentration of data in one cluster.

	Data labels	
	B	M
Cluster 1	357	211
Cluster 2	0	1

	Data labels	
	B	M
Cluster 1	357	210
Cluster 2	0	1
Cluster 3	0	1

	Data labels	
	B	M
Cluster 1	357	208
Cluster 2	0	2
Cluster 3	0	1
Cluster 4	0	1

Table 24: Table of cluster and actual labels for 2, 3 and 4 clusters for AGNES – single linkage algorithm.

Partition agreement

The results presented in Table 25 show the percentage of matched cases in pairs for different number of clusters (2, 3 and 4) obtained by using the AGNES with single linkage algorithm. Based on them, it can be concluded that regardless of the number of clusters, we have quite low agreement, not exceeding 63.5%. Low match index values may suggest that the single-linkage algorithm may have difficulties with correct assignment of observations into the appropriate clusters. Additionally, the match rate for different numbers of clusters is similar, suggesting that adding more clusters only slightly improve the match quality.

Number of clusters	Cases in matched pairs
2	62.92%
3	63.09%
4	63.44%

Table 25: Table of cases in matched pairs for 2, 3 and 4 clusters for AGNES – single linkage algorithm.

Internal validation

Based on the analysis of internal validation indices (Table 26), the optimal number of clusters for the AGNES with single linkage algorithm seems to be two. For this case, we obtained the highest values of Dunn and Silhouette indices and the lowest Connectivity, which suggests the best cluster structure and assignment of points to clusters.

Number of clusters	Connectivity	Dunn	Silhouette
2	2.9290	0.3097	0.7990
3	5.9829	0.2015	0.6875
4	10.8294	0.1206	0.6928

Table 26: Table of internal validation indices for 2, 3 and 4 clusters for AGNES – single linkage algorithm.

Stability indices

Stability indices for different numbers of clusters (2, 3 and 4) obtained using the the AGNES with single linkage algorithm are presented in Table 27. Analysis of these indices leads to analogous conclusions as for the previous methods. However, we need to underline the fact that there is only one observation in all but one cluster.

Number of clusters	APN	AD	ADM	FOM
2	0.0001	672.1851	0.3833	33.7801
3	0.0001	664.4625	0.4361	33.0041
4	0.0012	647.6612	2.7254	30.8900

Table 27: Table of Stability indices for 2, 3 and 4 clusters for AGNES – single linkage algorithm.

3.8 AGNES with complete linkage

Figure 15 shows two clusters that were obtained by the AGNES with complete linkage method. Colors of the observations (at the bottom of the dendrogram) indicate the true labels.

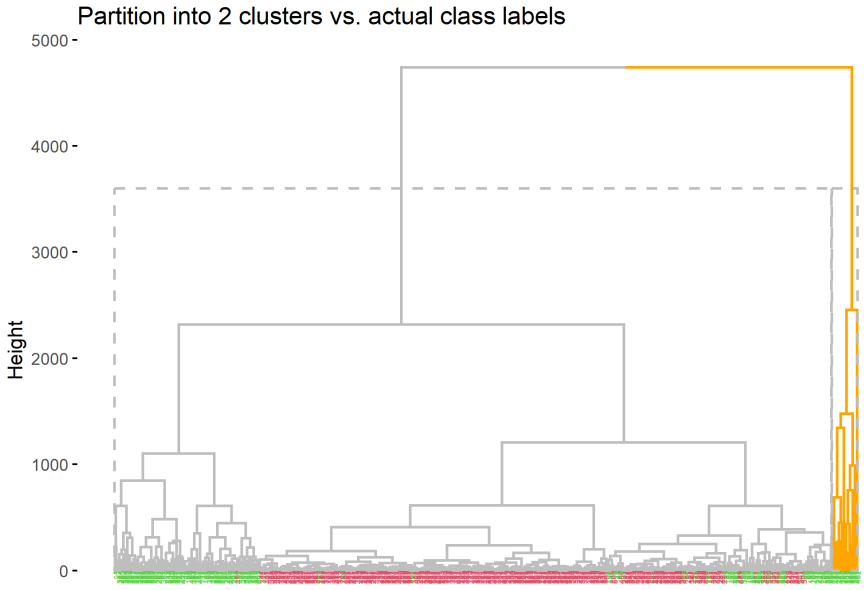

Figure 15: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with complete linkage.

3.8.1 Statistics of clusters (for 2 clusters)

By comparing the statistics for division into two clusters obtained by using the AGNES method with complete linkage (Tables 28 and 29) with AGNES method with average linkage (Tables 18 and 19), it can be seen that both algorithms give identical statistical values. Therefore conclusions are no different then ones for that case.

	mean	sd	median	mad
Radius1	13.7896	3.0686	13.2	2.6242
Texture1	19.156	4.2513	18.75	4.1661
Perimeter1	89.626	21.0797	85.48	17.8653
Area1	615.825	283.5074	538.4	211.567
Smoothness1	0.0961	0.014	0.0958	0.014
Compactness1	0.1018	0.0507	0.09	0.0454
Concavity1	0.0831	0.0727	0.0586	0.0573
Concave_points1	0.0457	0.0348	0.0325	0.0285
Symmetry1	0.1809	0.0273	0.1791	0.0254
Fractal_dimension1	0.0629	0.0071	0.0617	0.0061
Radius2	0.3777	0.2154	0.3141	0.1489
Texture2	1.2149	0.5529	1.108	0.4685
Perimeter2	2.6656	1.5495	2.23	1.069
Area2	35.0293	28.1693	23.92	12.8393
Smoothness2	0.007	0.0029	0.0064	0.0022
Compactness2	0.0251	0.0176	0.0202	0.0129
Concavity2	0.0312	0.03	0.0249	0.018
Concave_points2	0.0116	0.0062	0.0107	0.0051
Symmetry2	0.0206	0.0082	0.0187	0.0058
Fractal_dimension2	0.0038	0.0027	0.0031	0.0016
Radius3	15.7802	4.1418	14.8	3.3952
Texture3	25.5046	6.0536	25.22	6.2862
Perimeter3	103.8183	28.6476	96.66	24.5222
Area3	813.3239	444.0102	670.6	293.8513
Smoothness3	0.132	0.023	0.1312	0.0219
Compactness3	0.2493	0.1559	0.2089	0.1262
Concavity3	0.2632	0.2044	0.2151	0.1855
Concave_points3	0.1104	0.0626	0.0974	0.0627
Symmetry3	0.29	0.0623	0.2818	0.0511
Fractal_dimension3	0.084	0.0183	0.0799	0.0148

Table 28: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for AGNES – complete linkage algorithm.

	mean	sd	median	mad
Radius1	23.396	2.4375	23.24	2.2387
Texture1	22.9575	4.1304	21.97	3.41
Perimeter1	156.285	17.9462	153.15	18.829
Area1	1727.2	359.9746	1683.5	286.8831
Smoothness1	0.1047	0.0149	0.1046	0.0145
Compactness1	0.1737	0.0623	0.1798	0.0807
Concavity1	0.2447	0.1052	0.2309	0.1428
Concave_points1	0.1362	0.0429	0.1441	0.0624
Symmetry1	0.1876	0.0308	0.18	0.0193
Fractal_dimension1	0.0593	0.0059	0.0572	0.0061
Radius2	1.1591	0.5831	1.0085	0.3002
Texture2	1.2694	0.5267	1.2165	0.4125
Perimeter2	8.3688	4.4636	7.3145	2.2587
Area2	186.035	125.7755	154.45	56.7094
Smoothness2	0.0072	0.0043	0.006	0.0016
Compactness2	0.0372	0.0233	0.0284	0.0171
Concavity2	0.05	0.03	0.0378	0.0204
Concave_points2	0.0164	0.0049	0.0154	0.0043
Symmetry2	0.0201	0.0103	0.0164	0.0046
Fractal_dimension2	0.0039	0.0019	0.0036	0.0017
Radius3	29.691	2.6532	29.545	2.2239
Texture3	30.4145	6.9195	29.475	5.2262
Perimeter3	201.77	18.4442	200.95	18.829
Area3	2726.85	534.7879	2588.5	404.7498
Smoothness3	0.1421	0.0153	0.1405	0.0164
Compactness3	0.3908	0.1377	0.4065	0.0847
Concavity3	0.5202	0.1718	0.508	0.1792
Concave_points3	0.2298	0.0425	0.2367	0.0503
Symmetry3	0.2914	0.0513	0.2844	0.0378
Fractal_dimension3	0.0822	0.0116	0.0815	0.0096

Table 29: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for AGNES – complete linkage algorithm.

3.8.2 Validation

Silhouette index

Figure 16 shows Silhouette index plot for AGNES algorithm with complete linkage for both clusters. The average value is 0.69, which indicates quite good clustering and will be later compared with clustering using 3 and 4 clusters. Worth noticing is that in this method we observe highly imbalanced cluster division – there are fewer objects in second cluster.

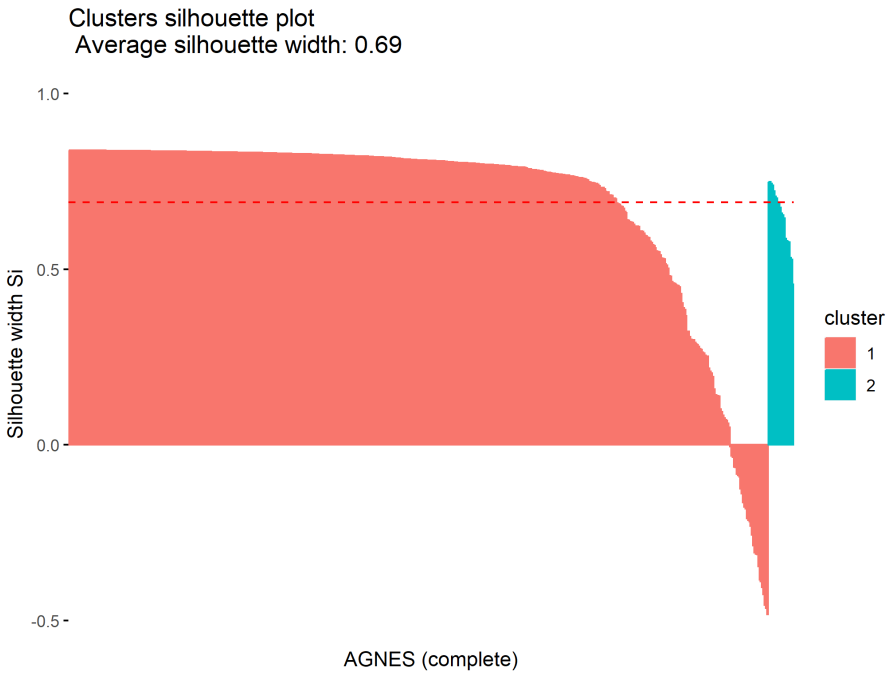

Figure 16: Silhouette index plot for AGNES method with complete linkage for 2 clusters.

Cluster labels vs actual labels

Table 30 shows the true labels in comparison with cluster to which they where assignes for division into 2, 3 and 4 clusters using the AGNES method with complete linkage. By comparing these results with the previous results of AGNES methods, analogous conclusions can be drawn. Therefore, in each case, cluster 1 contains the significant majority of observations, indicating a strong concentration of data in one cluster.

	Data labels	
	B	M
Cluster 1	357	192
Cluster 2	0	20

	Data labels	
	B	M
Cluster 1	357	192
Cluster 2	0	19
Cluster 3	0	1

	Data labels	
	B	M
Cluster 1	1	110
Cluster 2	356	82
Cluster 3	0	19
Cluster 4	0	1

Table 30: Table of cluster and actual labels for 2, 3 and 4 clusters for AGNES – complete algorithm.

Partition agreement

The results presented in Table 31 show the percentage of matched cases in pairs for different

number of clusters (2, 3 and 4) obtained using the AGNES with complete linkage algorithm. Similar to AGNES with average linkage, the results suggest that increasing the number of clusters improves the fit of observations to clusters.

Number of clusters	Cases in matched pairs
2	66.26%
3	66.26%
4	85.41%

Table 31: Table of cases in matched pairs for 2, 3 and 4 clusters for AGNES – complete linkage algorithm.

Internal validation

Based on the analysis of internal validation indices (Table 32), the optimal number of divisions into clusters for the AGNES with complete linkage algorithm seems to be two or three. For two clusters, we obtained the highest value of Silhouette index and the lowest Connectivity, and for three clusters the highest value of Dunn's index – analogously to AGNES with average linkage algorithm.

Number of clusters	Connectivity	Dunn	Silhouette
2	8.3500	0.0705	0.6909
3	11.2790	0.0747	0.6726
4	15.4925	0.0412	0.6707

Table 32: Table of internal validation indices for 2, 3 and 4 clusters for AGNES – complete linkage algorithm.

Stability indices

Stability indices for different numbers of clusters (2, 3 and 4) obtained using the the AGNES with complete linkage algorithm are presented in Table 33. Analysis of these indices leads to analogous conclusions as for the previous methods.

Number of clusters	APN	AD	ADM	FOM
2	0.0032	584.4664	12.1418	30.0527
3	0.0480	575.5565	61.1540	18.5246
4	0.0180	342.1326	32.3592	18.1178

Table 33: Table of Stability indices for 2, 3 and 4 clusters for AGNES – complete linkage algorithm.

3.9 Diana

Figure 17 shows two clusters that were obtained by the DIANA method. Colors of the observations (at the bottom of the dendrogram) indicate the true labels.

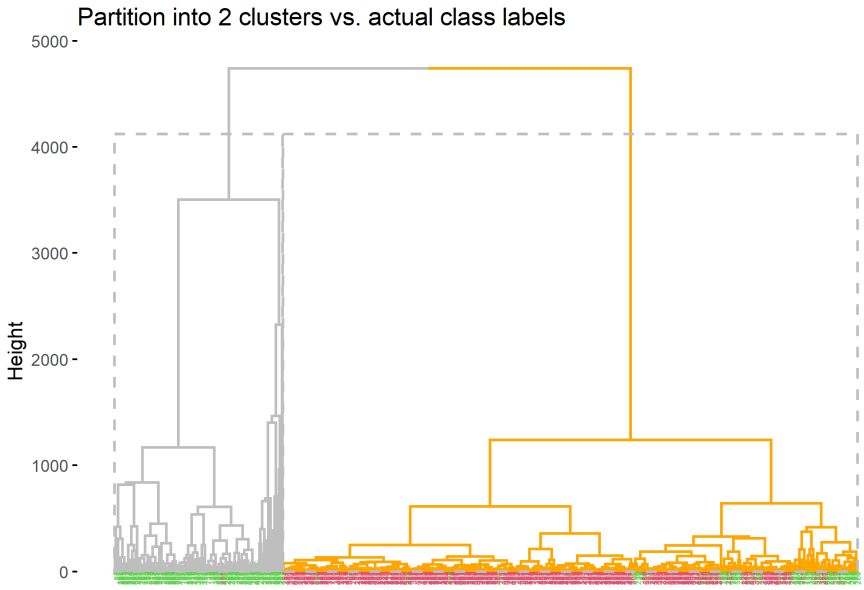

Figure 17: Dendrogram divided into 2 clusters with comparison of real classes for the DIANA method.

3.9.1 Statistics of clusters (for 2 clusters)

Analyzing the cluster statistics tables (Tables 34 and 35) for the DIANA algorithm, it is again visible that the clusters differ significantly in terms of feature values. This suggests that the algorithm successfully divided the observations into subgroups with different properties.

	mean	sd	median	mad
Radius1	19.4404	2.3864	19.19	1.9126
Texture1	21.6552	3.9483	21.31	3.5731
Perimeter1	128.5713	16.8518	127.5	13.4917
Area1	1192.6426	309.9709	1148	231.7304
Smoothness1	0.1012	0.0117	0.1001	0.0125
Compactness1	0.1472	0.0538	0.1328	0.0412
Concavity1	0.1759	0.0782	0.1594	0.0712
Concave_points1	0.1005	0.0341	0.0946	0.029
Symmetry1	0.1911	0.0283	0.1867	0.0218
Fractal_dimension1	0.0604	0.0065	0.06	0.006
Radius2	0.7409	0.3521	0.6874	0.2418
Texture2	1.2146	0.5101	1.077	0.4077
Perimeter2	5.2264	2.7352	4.655	1.8636
Area2	95.7507	67.7396	81.89	34.4853
Smoothness2	0.0066	0.0024	0.0062	0.0018
Compactness2	0.032	0.0172	0.0279	0.0134
Concavity2	0.0422	0.0205	0.0383	0.0165
Concave_points2	0.0156	0.0054	0.0147	0.0041
Symmetry2	0.0203	0.0096	0.0183	0.0061
Fractal_dimension2	0.0039	0.0019	0.0037	0.0016
Radius3	23.766	3.2929	23.17	3.1579
Texture3	28.8311	5.3749	28.14	4.8036
Perimeter3	158.6822	23.9359	152.9	19.8668
Area3	1760.8605	526.4227	1646	440.3322
Smoothness3	0.1402	0.0188	0.1408	0.0188
Compactness3	0.3546	0.1473	0.342	0.1223
Concavity3	0.4466	0.1678	0.4024	0.1474
Concave_points3	0.1919	0.0435	0.1872	0.0388
Symmetry3	0.3111	0.0657	0.3055	0.0543
Fractal_dimension3	0.0859	0.0165	0.0837	0.0126

Table 34: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for Diana algorithm.

	mean	sd	median	mad
Radius1	12.5696	1.9185	12.62	1.8977
Texture1	18.5961	4.1549	18.165	3.8696
Perimeter1	81.2379	13.1165	81.35	12.4909
Area1	497.2295	149.4386	489.45	147.8893
Smoothness1	0.095	0.0144	0.0942	0.0145
Compactness1	0.0918	0.0455	0.0795	0.0385
Concavity1	0.0633	0.0596	0.0434	0.0369
Concave_points1	0.0338	0.0245	0.0275	0.0184
Symmetry1	0.1783	0.0265	0.1745	0.0245
Fractal_dimension1	0.0635	0.0071	0.0622	0.0057
Radius2	0.3067	0.1433	0.2743	0.1019
Texture2	1.2175	0.5638	1.1095	0.4787
Perimeter2	2.174	0.9953	1.994	0.7754
Area2	24.0908	13.1436	20.755	8.4286
Smoothness2	0.0072	0.0031	0.0065	0.0023
Compactness2	0.0236	0.0177	0.0179	0.0107
Concavity2	0.0289	0.0319	0.0205	0.0149
Concave_points2	0.0107	0.0059	0.0097	0.0047
Symmetry2	0.0206	0.0078	0.0188	0.0058
Fractal_dimension2	0.0038	0.0028	0.003	0.0015
Radius3	14.0712	2.3887	13.845	2.3647
Texture3	24.7526	6.0559	24.075	6.0564
Perimeter3	92.1855	17.0032	90.19	16.3901
Area3	622.5018	210.1545	590.25	201.9301
Smoothness3	0.1301	0.0234	0.1292	0.022
Compactness3	0.2249	0.1479	0.1855	0.1044
Concavity3	0.2211	0.1912	0.1736	0.1429
Concave_points3	0.0919	0.0526	0.0829	0.0417
Symmetry3	0.2839	0.0594	0.2766	0.0477
Fractal_dimension3	0.0834	0.0185	0.0787	0.0137

Table 35: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for Diana algorithm.

3.9.2 Validation

Silhouette index

Figure 18 shows Silhouette index plot for DIANA algorithm for both clusters. The average value is 0.7, which indicates quite good clustering and will be later compared with clustering using 3 and 4 clusters.

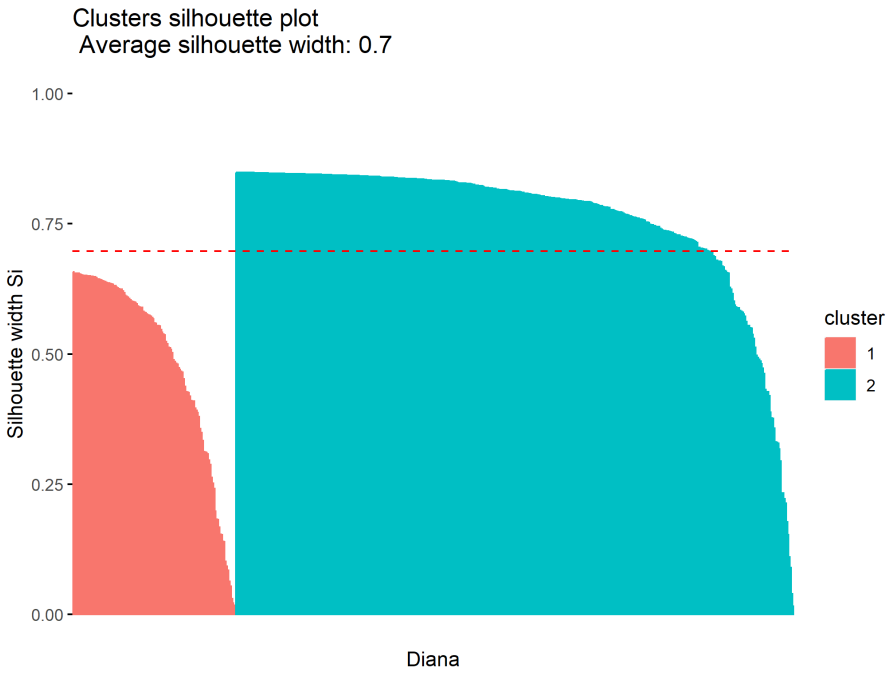

Figure 18: Silhouette index plot for DIANA method for 2 clusters.

Cluster labels vs actual labels

Table 36 shows the true labels in comparison with cluster to which they where assigned for division into 2, 3 and 4 clusters using the DIANA method. Based on that, it can be seen that for a larger number of clusters, the newer clusters mainly consist of observations from malligant class. This suggests that the algorithm split the data in order to ephasize diversity in the classes of observations. Therefore, additional clusters may better deal with data diversity and more precisely separate subgroups of observations.

	Data labels	
	B	M
Cluster 1	1	128
Cluster 2	356	84

	Data labels	
	B	M
Cluster 1	1	109
Cluster 2	356	84
Cluster 3	0	19

	Data labels	
	B	M
Cluster 1	1	109
Cluster 2	356	84
Cluster 3	0	18
Cluster 4	0	1

Table 36: Table of cluster and actual labels for 2, 3 and 4 clusters for Diana algorithm.

Partition agreement
Table 37 describe the percentage of matched observations in pairs for different number of clusters (2, 3 and 4) for DIANA algorithm shows that regardless of the number of clusters, the percentage of matched pairs remains at 85.06%.

Number of clusters	Cases in matched pairs
2	85.06%
3	85.06%
4	85.06%

Table 37: Table of cases in matched pairs for 2, 3 and 4 clusters for diana algorithm.

Internal validation
Based on the analysis of internal validation indices (Table 38), the optimal number of clusters for the DIANA algorithm seems to be two or four. For two clusters, we obtained the highest value of Silhouette index and the lowest Connectivity, and for four clusters the highest value of Dunn's index.

Number of clusters	Connectivity	Dunn	Silhouette
2	11.1397	0.0115	0.6975
3	20.0468	0.0173	0.6749
4	22.9758	0.0275	0.6716

Table 38: Table of internal validation indices for 2, 3 and 4 clusters for diana algorithm.

Stability indices
Stability indices for different numbers of clusters (2, 3 and 4) obtained by using the the DIANA algorithm are presented in Table 39. Analysis of these indices leads to analogous conclusions as for the previous methods.

Number of clusters	APN	AD	ADM	FOM
2	0.0016	379.0658	1.7402	20.1483
3	0.0025	333.0032	2.8803	17.4587
4	0.0026	327.9710	2.8878	17.0843

Table 39: Table of Stability indices for 2, 3 and 4 clusters for diana algorithm.

3.9.3 Summary

Based on the analysis of various clustering algorithms, the following conclusions can be drawn:

- K-means: For division into 3 clusters, the highest match percentage of case pairs (88.93%) was obtained, suggesting the best fit of the model to the data. However, internal validation metrics indicate that the optimal number of clusters for the K-means algorithm is two.
- PAM: The highest case pair matching percentage (86.82%) was obtained for division into two clusters. Internal validation indices also suggest division into two clusters as the most suitable. Therefore, the optimal number of clusters seems to be two.

- Fuzzy C-means: Fuzzy Silhouette index values indicate that the division into two clusters reflects the structure of the data the best. However, the pairwise case matching results suggest that division into three clusters achieved the highest match percentage (88.58%), and the cluster quality metrics did not clearly indicate the best number of clusters.

- AGNES: Analysis of internal validation metrics suggests that the optimal number of clusters is two or three. While for 4 clusters the highest matching percentage of case pairs (87.87% - average linkage, 63.44% - single linkage, 85.41% - complete linkage) was obtained. However, the unbalanced division into clusters may be problematic.

- DIANA: The results of internal validation metrics indicate that the optimal number of clusters for the DIANA algorithm is two or four. Additionally, regardless of the number of clusters, the percentage of matched pairs remains constant (85.06%). Therefore, the optimal number of clusters for the DIANA algorithm is two or four.

It is worth noting that the K-means, PAM, Fuzzy c-means and DIANA methods achieve a high percentage of matched cases for all considered cluster numbers, which suggests their relative universality and effectiveness in clustering data. However, for the AGNES method using average and complete linkage, depending on the number of clusters (2/3 vs. 4), significant differences were observed in the percentage of matched cases.

All in all K-means and Fuzzy C-means have the highest percentage of matched pairs, nevertheless values for most of the other algorithms are not signifficantly worse.

4 Dimesionality reduction

We will perform dimensionality reduction using principal component analysis (PCA). Its goal is to simplify the data structure by reducing the number of explanatory variables while retaining the relevant information contained in the data. This may contribute to better understanding of the data and making more accurate decisions in the context of breast cancer diagnosis. Due to signifficant differences in variances of features PCA was preceded by normalization of the dataset.

The bar chart 19 shows the variance for all newly created principal components (PC). Based on that, we can notice large differences in variances between individual PCs, which indicates the existence of components that have a significant impact on the variability of the data, in comparison to other ones. Therefore, it is worth considering dimension reduction to focus on more influential features, thus reducing the number of dimensions of the data.

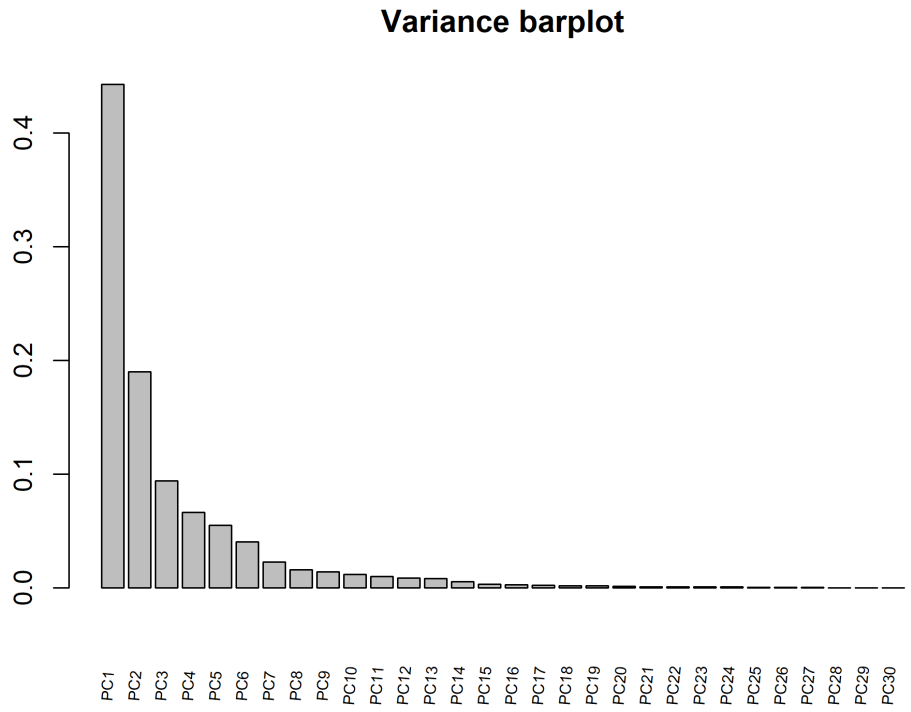

Figure 19: Barplot of variance for all features.

Figure 20 shows barplot of the cumulative variance for all principal components. Based on this, it can be seen that the first five PCs exceed the cumulative variance threshold of 80%. This means that first five components explain over 80% of the variance. One may concider first seven principal components as they explan over 90% of the variability of the dataset. However we want to reduce the dimensionality as much as we can, without losing too much information.

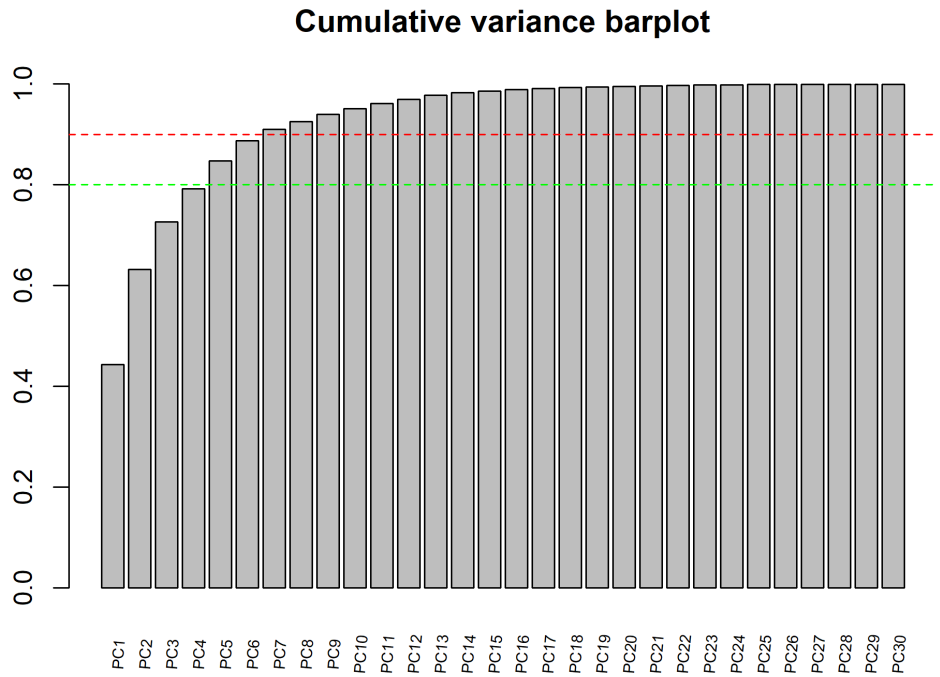

Figure 20: Barplot of cumulative variance for all features with horizontal lines for 80% (green) and 90% (red).

Based on Table 40 presenting eigenvalues, variances and cumulative variances percentages, it can be concluded that the first five dimensions explain over 80% of the variability in the dataset (84.73%). This is consistent with previous findings that indicated the possibility of reducing data dimensions while maintaining over 80% of the variance. Additionally, the eigenvalues for the first five dimensions are significantly larger than for the subsequent dimensions, indicating their greater importance in explaining the variability of the dataset. Therefore, reducing dimensions to five may be appropriate to retain the relevant information contained in the data.

Additionally, plots of contribution of all features (and top 10 having the most impact) to first five principal components are presented in appendix 4.

	Eigenvalue	Variance percentage	Cumulative variance percentage
Dim.1	13.28	44.27	44.27
Dim.2	5.69	18.97	63.24
Dim.3	2.82	9.39	72.64
Dim.4	1.98	6.60	79.24
Dim.5	1.65	5.50	84.73
Dim.6	1.21	4.02	88.76
Dim.7	0.68	2.25	91.01
Dim.8	0.48	1.59	92.60
Dim.9	0.42	1.39	93.99
Dim.10	0.35	1.17	95.16
Dim.11	0.29	0.98	96.14
Dim.12	0.26	0.87	97.01
Dim.13	0.24	0.80	97.81
Dim.14	0.16	0.52	98.34
Dim.15	0.09	0.31	98.65
Dim.16	0.08	0.27	98.92
Dim.17	0.06	0.20	99.11
Dim.18	0.05	0.18	99.29
Dim.19	0.05	0.16	99.45
Dim.20	0.03	0.10	99.56
Dim.21	0.03	0.10	99.66
Dim.22	0.03	0.09	99.75
Dim.23	0.02	0.08	99.83
Dim.24	0.02	0.06	99.89
Dim.25	0.02	0.05	99.94
Dim.26	0.01	0.03	99.97
Dim.27	0.01	0.02	99.99
Dim.28	0.00	0.01	100.00
Dim.29	0.00	0.00	100.00
Dim.30	0.00	0.00	100.00

Table 40: Table of eigenvalues, variance percentage and cumulative variance percentage for PCA.

Figure 21 shows correlation of variables and principal components. It is easily seen that first five dimensions have highly correlated features. In particular one can see that the first two have strong negative correlation and as the number of dimensions increases values of correlations become closer to 0.

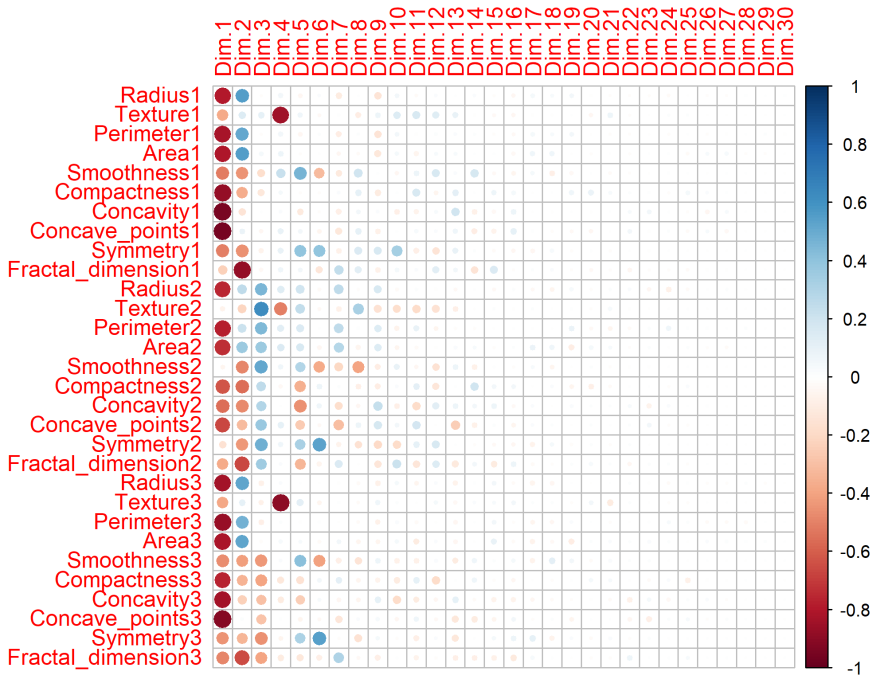

Figure 21: Correlation of variables and principal components (PCs).

Graph 22 presents features with colors indicating level of contribution of each of them to first two principal components. One can see that a lot of them are grouped together and have large impact on PCs. Detailed contribution to all PCs can be seen in appendix 4.

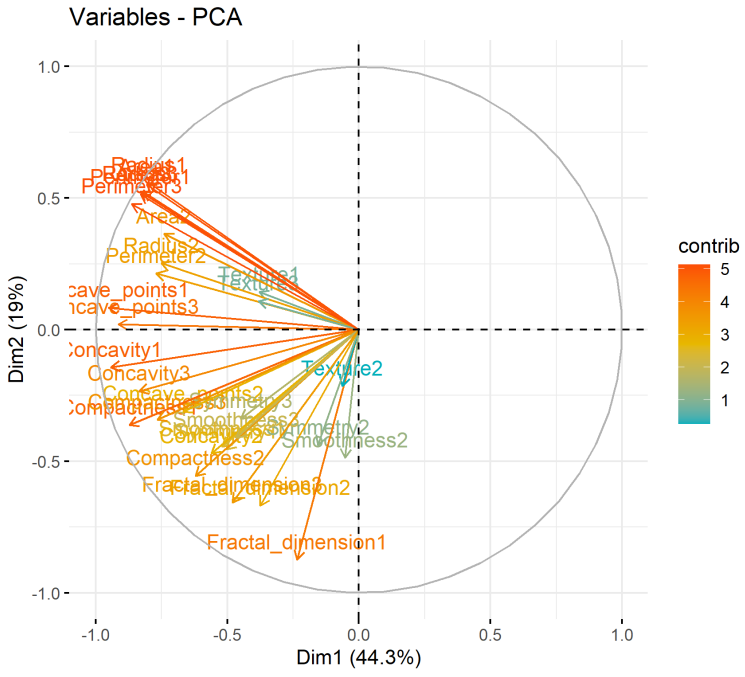

Figure 22: Graph of features with colors corresponding to the contribution of each feature for PC1 and PC2.

Figures 23 and 24 present multidimensional distribution of data points (the first one only for first two principal components) with colors indicating true labels. One can see that the first component divide objects neatly into two quite well sematated groups. As the component number increase, the division becomes less strict.

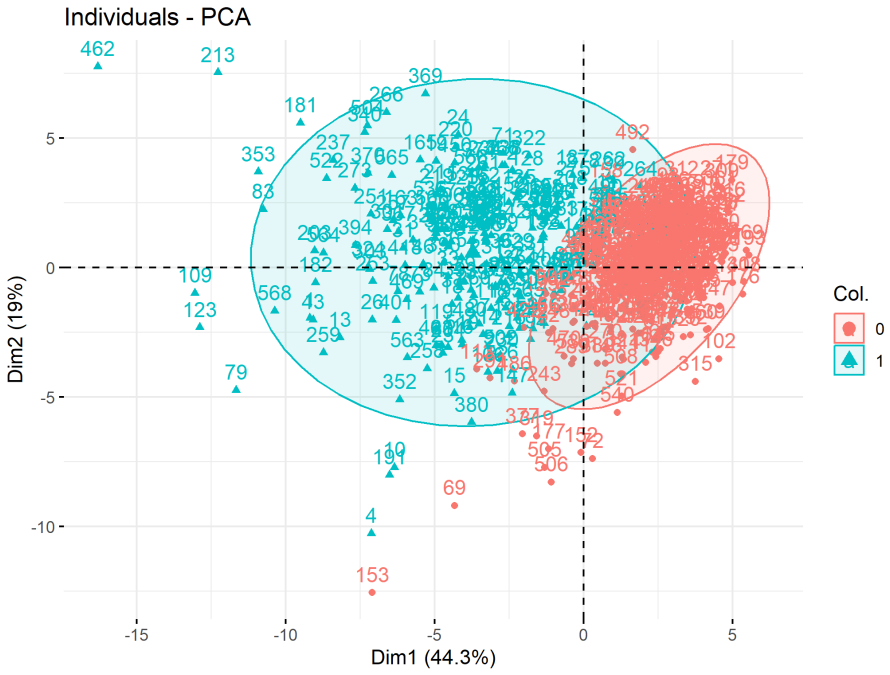

Figure 23: Scatterplot in space (PC1, PC2) with group membership information.

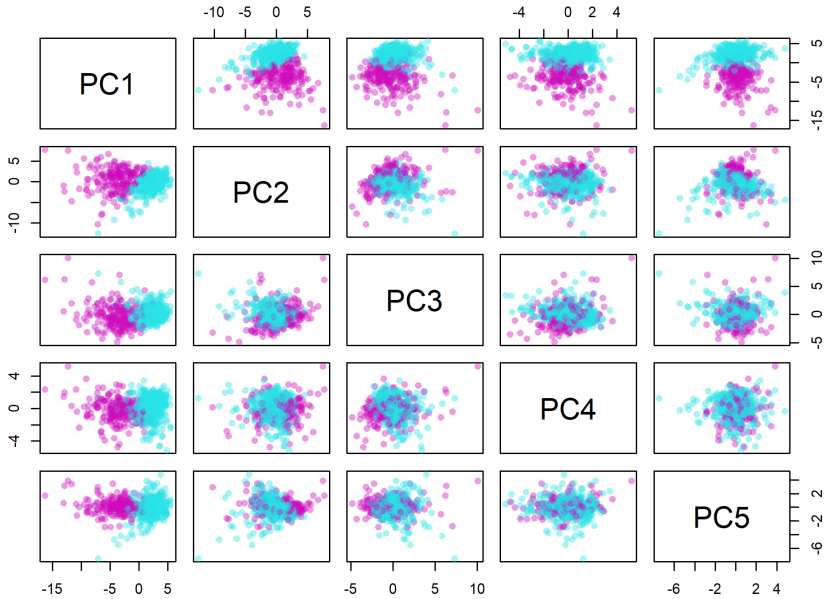

Figure 24: Two dimensional scatterplots of data points with colors indicating true labels.

More dimension reduction visualizations are available in appendix [4].

5 Clustering after dimensionality reduction

5.1 Optimal number of groups

To determine optimal number of clusters, we used the Elbow, Silhouette and Gap statistic method for K-means, PAM and hierarchical algorithms. Below we present graphs only for hierarchical algorithms, as the graphs looked similar for each group.

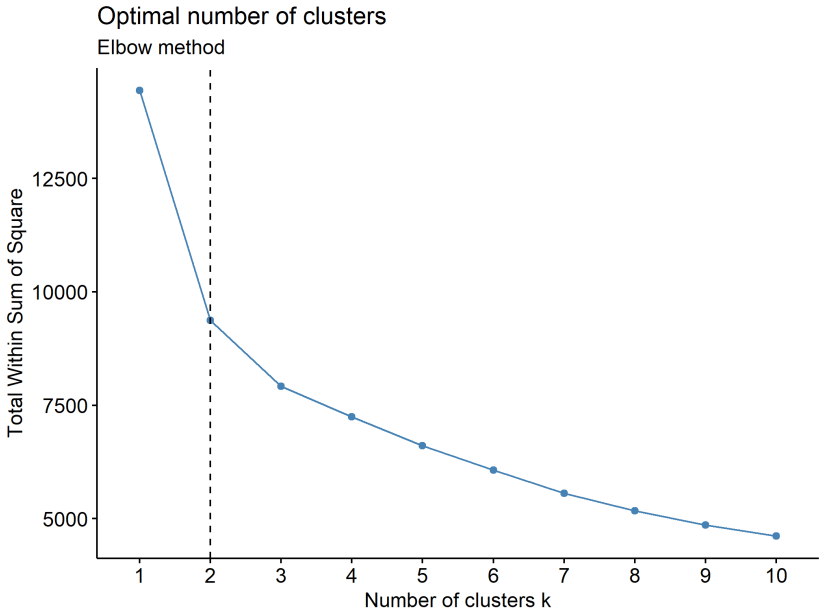

Figure 25: The optimal number of clusters for the elbow method for hierarchical algorithms after dimensionality reduction.

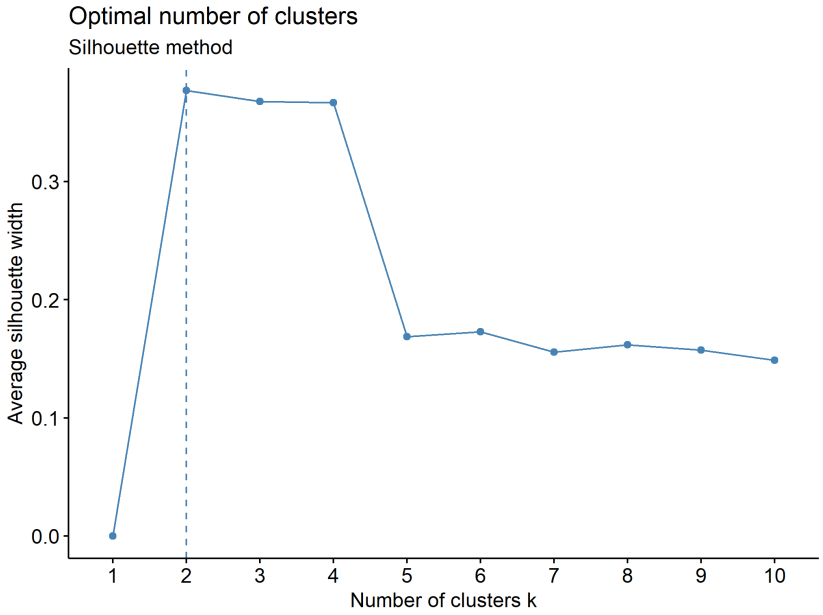

Figure 26: The optimal number of clusters for the silhouette method for hierarchical algorithms after dimensionality reduction.

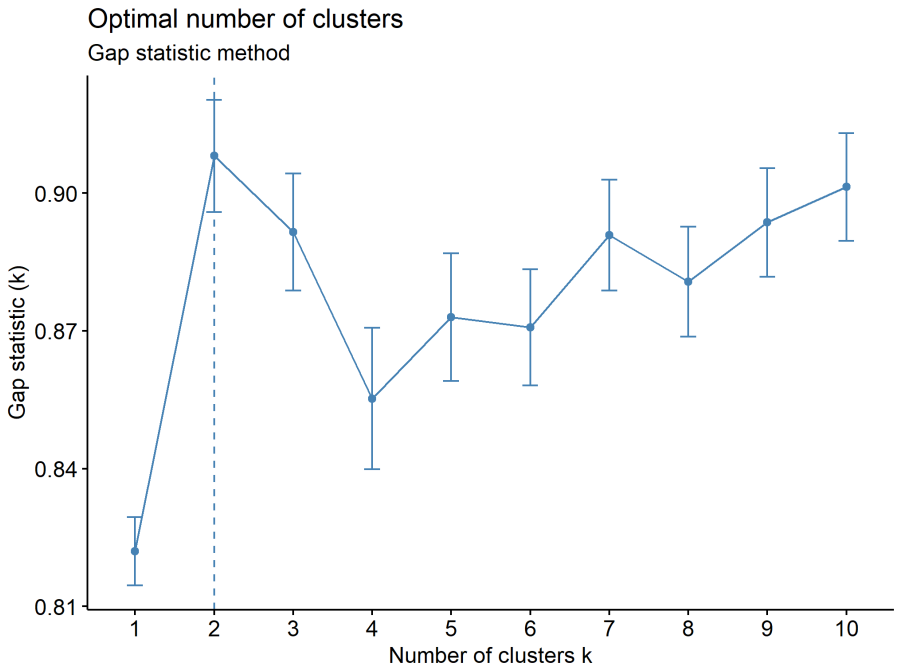

Figure 27: The optimal number of clusters for the gap statistic method for hierarchical algorithms after dimensionality reduction.

Based on Figures 25, 26 and 27, it can be seen that all used approaches indicate the optimal division of data into 2 groups.

After detailed consideration and in the context of our goals, we decided to conduct further analysis using division into only 2 clusters, as the performance can be compared to one before dimensional reducion.

Moreover, as visualization is an important aspect of analyzing clusterization, multidimensional scatterplots for all concidered methods are presented in appendix 4.

5.2 K-means

Figure 28 shows two clusters that were obtained by the K-means method after dimensionality reduction. It is easy to see that unlike the clusters before the reduction, these overlap to a large extent.

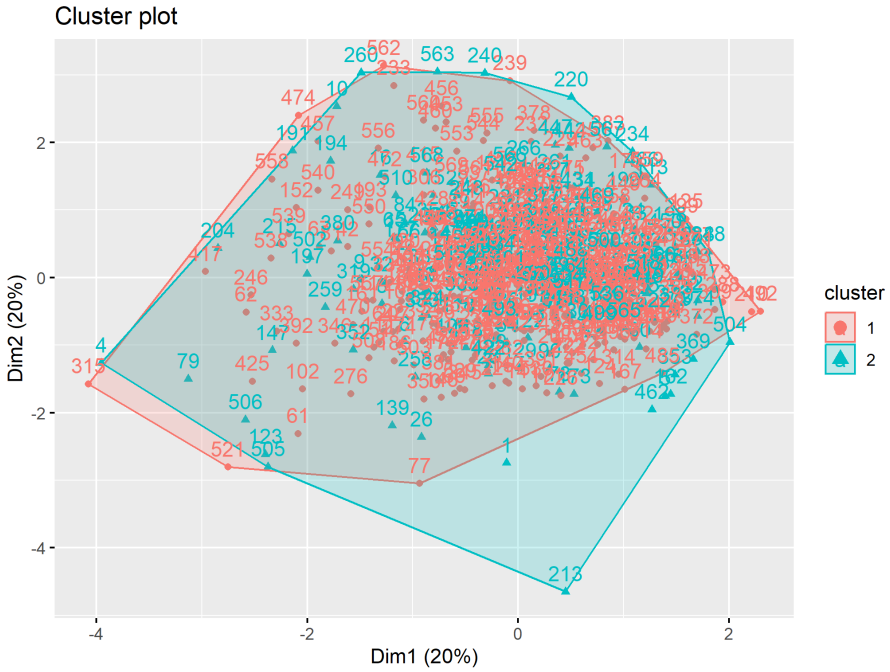

Figure 28: Clusters obtained using the 2-means method after dimensionality reduction.

5.2.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for two clusters obtained using the K-means algorithm after dimensionality reduction (Tables 41 and 42), it can be seen that there are significant differences between clusters in terms of means, standard deviations, medians and mean absolute deviations for each principal component. These differences suggest that these clusters may represent different data structures after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	2.2262	1.4084	2.2823	1.4994
PC2	0.0337	1.7357	0.1791	1.6583
PC3	0.0887	1.3127	-0.0759	1.1617
PC4	0.0334	1.3883	0.1846	1.3055
PC5	0.0159	1.1664	-0.0071	0.9088

Table 41: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for K-means algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	-4.3032	2.6459	-3.7104	2.1094
PC2	-0.0651	3.3023	0.3086	2.9495
PC3	-0.1714	2.2157	-0.5236	1.8751
PC4	-0.0645	1.4449	0.0343	1.2743
PC5	-0.0308	1.4881	0.0731	0.9283

Table 42: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for K-means algorithm after dimensionality reduction.

5.2.2 Validation

Dispersion

Figure 29 illustrates the comparison between within-cluster and between-cluster dispersion for the 2-means algorithm after dimensionality reduction. Again, it is evident that with an increase in the number of clusters, the within-cluster dispersion also increases. Consequently, opting for 2 clusters will yield more compact clusters compared to selecting a larger number of groups. The between-cluster dispersion further supports the notion of using a smaller number of clusters, indicating greater diversity among data points across different groups.

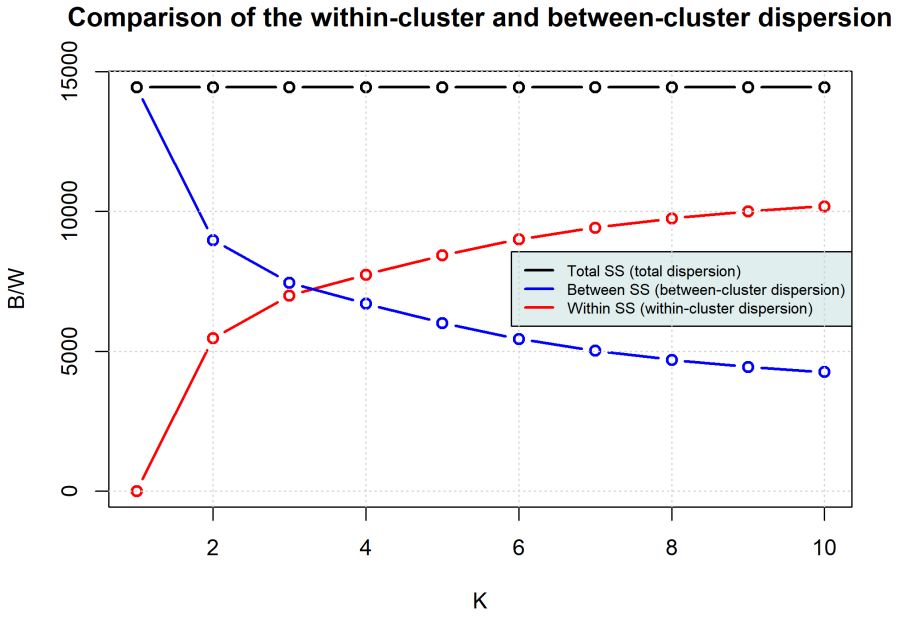

Figure 29: Comparision of the within-cluster and beetwen-cluster dispersion for 2-means algorithm after dimensionality reduction.

Silhouette index

Figure 30 shows Silhouette index plot for 2-means algorithm for both clusters after dimensionality reduction. The average value is 0.39, suggesting that the clustering is moderately good. Clusters are reasonably well-defined but there is still room for improvement.

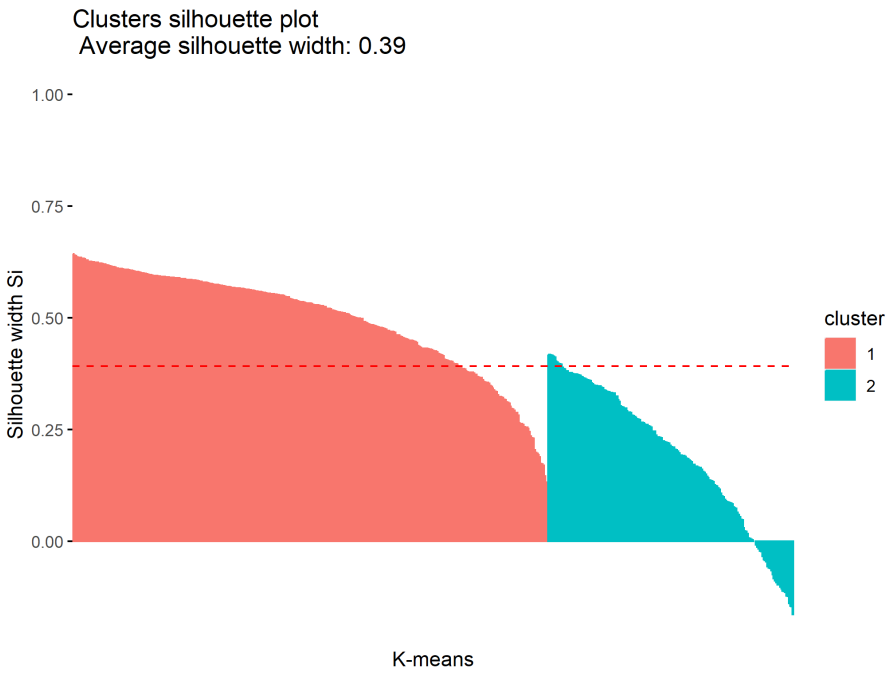

Figure 30: Silhouette index plot for 2-means algorithm after dimensionality reduction.

Cluster labels vs actual labels

Table 43 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the K-means algorithm after dimensionality reduction. It is easily noticeable that cluster 1 contains most of the cases labeled B and only a small number of cases labeled M. And cluster 2 is dominated by malligant cases, while the number of benign cases is much smaller.

	Data labels	
	B	M
Cluster 1	339	36
Cluster 2	18	176

Table 43: Table of cluster and actual labels for 2 clusters for K-means algorithm after dimensionality reduction.

5.3 PAM

Figure 31 shows two clusters that were obtained by the PAM method after dimensionality reduction. It is easy to see that unlike the clusters before the reduction, these overlap to a large extent.

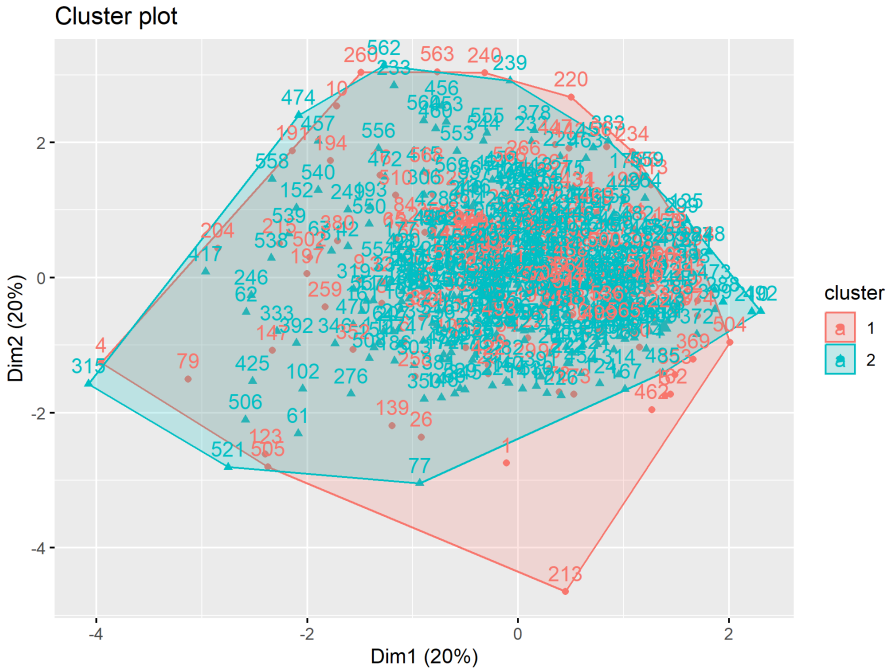

Figure 31: Two clusters obtained using the PAM method after dimensionality reduction.

5.3.1 Statistics of clusters (for 2 clusters)

Statistics for the two clusters obtained using the PAM algorithm after dimensionality reduction are presented in Tables 44 and 45. By comparing these results with previous results from the K-means algorithm (Tables 41 and 42), very similar conclusions can be drawn. Both algorithms suggest the existence of two different groups of objects, differing in feature values and their distributions.

Feature	mean	sd	median	mad
PC1	-4.4024	2.6277	-3.7802	2.0482
PC2	0.1095	3.2021	0.614	2.7425
PC3	-0.1947	2.2145	-0.5236	1.8751
PC4	-0.0601	1.4526	0.049	1.2726
PC5	0.0112	1.4519	0.0856	0.92

Table 44: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for PAM algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	2.1723	1.4611	2.2389	1.5067
PC2	-0.054	1.8581	0.137	1.6696
PC3	0.0961	1.3307	-0.0759	1.1617
PC4	0.0296	1.3855	0.1826	1.3084
PC5	-0.0055	1.1946	-0.0161	0.9338

Table 45: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for PAM algorithm after dimensionality reduction.

5.3.2 Validation

Silhouette index

Figure 32 shows Silhouette index plot for PAM algorithm for both clusters after dimensionality reduction. The average value is again 0.39, suggesting that the clustering is moderately good.

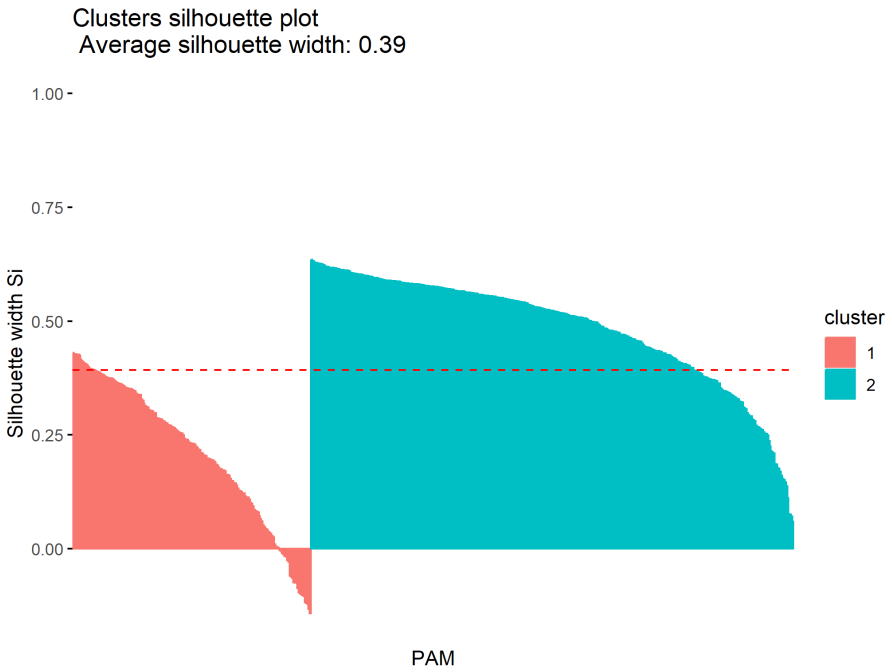

Figure 32: Silhouette index plot for PAM algorithm for 2 clusters after dimensionality reduction.

Cluster labels vs actual labels

Table 46 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the PAM algorithm after dimensionality reduction. It is easily noticeable that cluster 1 is dominated by malligant cases, while the number of benign cases is much smaller. Cluster 2 contains most of the cases labeled B and only a small number of cases labeled M.

	Data labels	
	B	M
Cluster 1	12	176
Cluster 2	345	36

Table 46: Table of cluster and actual labels for 2 clusters for PAM algorithm after dimensionality reduction.

5.4 Fuzzy c-means

Figure 33 shows two clusters that were obtained by the Fuzzy c-means method after dimensionality reduction. It is easy to see that unlike the clusters before the reduction, these overlap to a large extent.

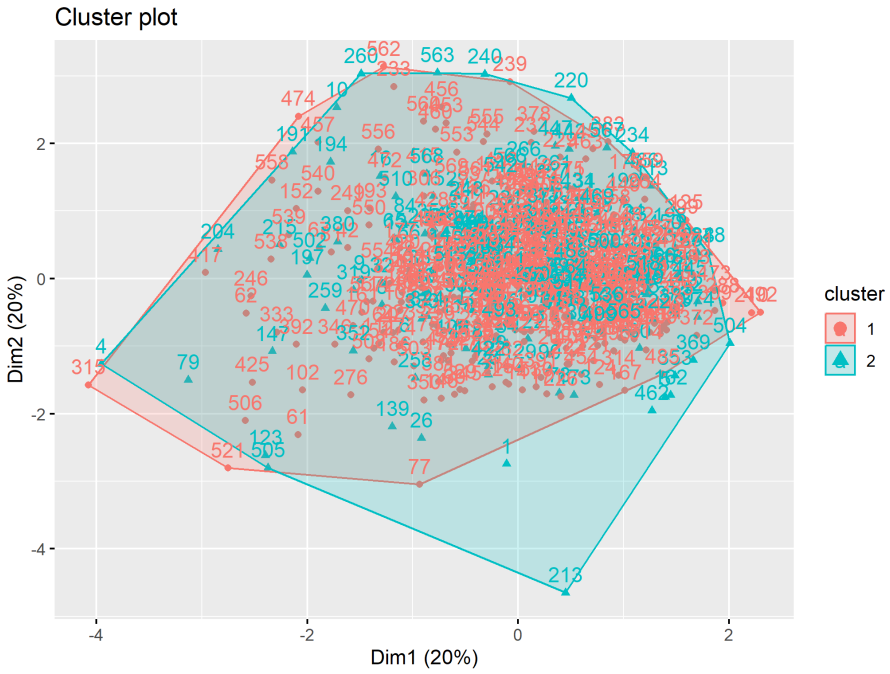

Figure 33: Two clusters obtained using the fuzzy c-means method for m=2 after dimensionality reduction.

5.4.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for the two hard clusters obtained by using the Fuzzy c-means method after dimensionality reduction (Tables 47 and 48) with previously analyzed PAM (Tables 44 and 45), it can be seen that both algorithms give similar results. Therefore, analogous conclusions as for PAM can be drawn.

Feature	mean	sd	median	mad
PC1	-4.2829	2.6539	-3.7084	2.1535
PC2	0.045	3.2109	0.3785	2.8612
PC3	-0.2082	2.1916	-0.5513	1.8292
PC4	-0.0733	1.4375	0.03	1.2712
PC5	-0.018	1.4765	0.0714	0.9238

Table 47: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for fuzzy c-means algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	2.2331	1.4043	2.2904	1.4999
PC2	-0.0235	1.8174	0.1568	1.6536
PC3	0.1086	1.3261	-0.066	1.1485
PC4	0.0382	1.3918	0.1853	1.3055
PC5	0.0094	1.1733	-0.0116	0.9104

Table 48: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for fuzzy c-means algorithm after dimensionality reduction.

5.4.2 Validation

Cluster labels vs actual labels

Table 49 shows the true labels in comparison with hard clusters to which they where assigned for 2 clusters using the Fuzzy c-means algorithm after dimensionality reduction. It can be seen that the results are analogous to those for the PAM method.

	Data labels	
	B	M
Cluster 1	16	179
Cluster 2	341	33

Table 49: Table of cluster and actual labels for 2 clusters for fuzzy c-means algorithm after dimensionality reduction.

5.5 AGNES with average linkage

Figure 34 shows two clusters that were obtained by the AGNES with average linkage method after dimensionality reduction. Colors of the observations (at the bottom of the dendrogram) indicate the true labels. It can be easily seen that most of the observations are assigned to one of the clusters leaving the second one with few observations.

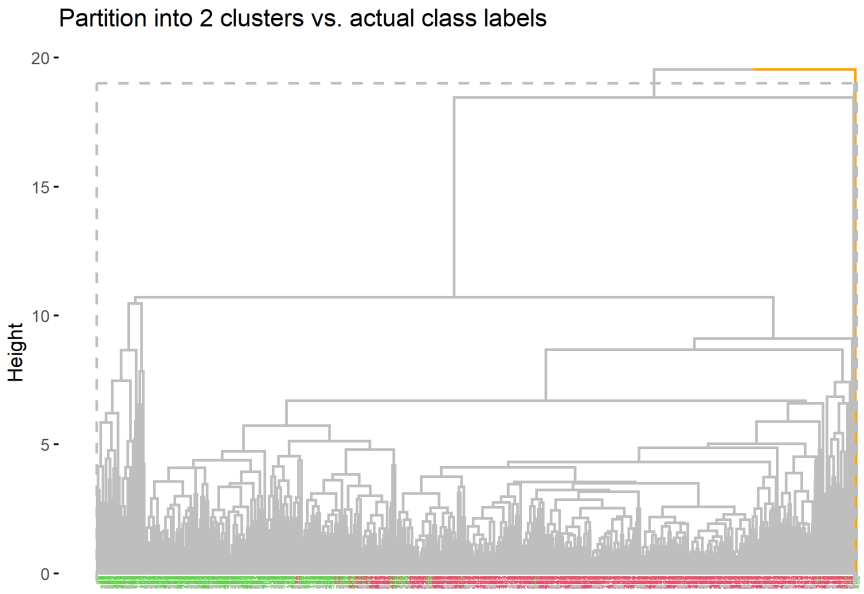

Figure 34: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with average linkage after dimensionality reduction.

5.5.1 Statistics of clusters (for 2 clusters)

Comparing the statistics for two clusters obtained using the AGNES algorithm with average linkage after dimensionality reduction (Tables 50 and 51) it can be seen that for cluster 1, the mean values for all principal components are close to zero and the standard deviation values are relatively high, suggesting that the data in this cluster are scattered around the center of the coordinate system. For cluster 2 it is the opposite. Therefore, the algorithm divided the data into two clusters that show significant differences in data dispersion and concentration.

Feature	mean	sd	median	mad
PC1	0.0504	3.5483	1.175	3.0889
PC2	-0.027	2.346	0.188	2.0009
PC3	-0.0288	1.6057	-0.161	1.3474
PC4	-0.0133	1.3894	0.0941	1.2988
PC5	-0.0098	1.274	0.0278	0.8922

Table 50: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for AGNES – average linkage algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	-14.2896	2.8501	-14.2896	2.9879
PC2	7.6529	0.1642	7.6529	0.1722
PC3	8.1667	2.739	8.1667	2.8715
PC4	3.7579	2.0249	3.7579	2.1228
PC5	2.7785	1.514	2.7785	1.5872

Table 51: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for AGNES – average linkage algorithm after dimensionality reduction.

5.5.2 Validation

Silhouette index

Figure 35 shows Silhouette index plot for AGNES algorithm with average linkage for both clusters after dimensionality reduction. The average value is 0.67, which indicates quite good clustering. Worth noticing is that in this method we observe highly imbalanced cluster division – there are fewer objects in second cluster.

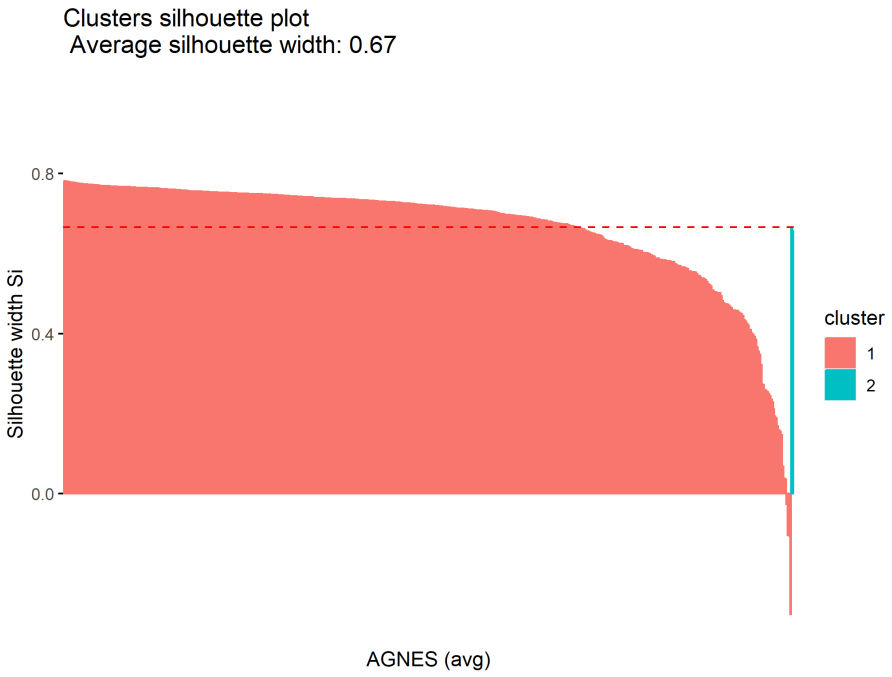

Figure 35: Silhouette index plot for AGNES method with average linkage for 2 clusters after dimensionality reduction.

Cluster labels vs actual labels

Table 52 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the AGNES method with average linkage after dimensionality reduction. It can be seen that most of the observations are assigned to cluster 1, while cluster 2 contains only 2 M observations.

	Data labels	
	B	M
Cluster 1	357	210
Cluster 2	0	2

Table 52: Table of cluster and actual labels for 2 clusters for AGNES – average linkage algorithm after dimensionality reduction.

5.6 AGNES with single linkage

Figure 36 shows two clusters that were obtained by the AGNES with single linkage method after dimensionality reduction. Colors of the observations (at the bottom of the dendrogram) indicate the true labels. It can be easily seen that most of the observations are assigned to one of the clusters leaving the second one with few observations.

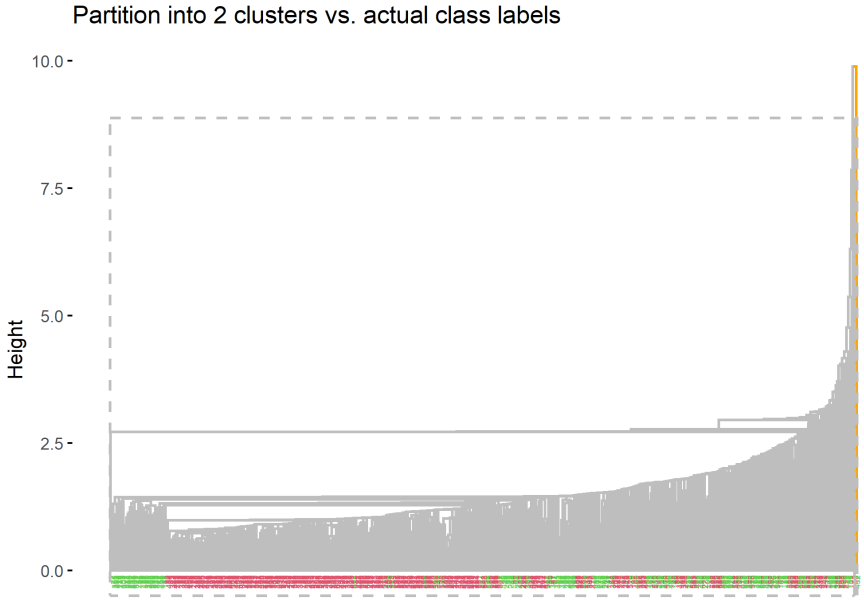

Figure 36: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with single linkage after dimensionality reduction.

5.6.1 Statistics of clusters (for 2 clusters)

By comparing the statistics for division into two clusters obtained by using the AGNES method with single linkage after dimensionality reduction (Tables 53 and 54) with AGNES method with average linkage after dimensionality reduction (Tables 50 and 51), it can be seen that both algorithms give identical statistical values. Therefore conclusions are no different then ones for that case.

Feature	mean	sd	median	mad
PC1	0.0504	3.5483	1.175	3.0889
PC2	-0.027	2.346	0.188	2.0009
PC3	-0.0288	1.6057	-0.161	1.3474
PC4	-0.0133	1.3894	0.0941	1.2988
PC5	-0.0098	1.274	0.0278	0.8922

Table 53: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for AGNES – single linkage algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	-14.2896	2.8501	-14.2896	2.9879
PC2	7.6529	0.1642	7.6529	0.1722
PC3	8.1667	2.739	8.1667	2.8715
PC4	3.7579	2.0249	3.7579	2.1228
PC5	2.7785	1.514	2.7785	1.5872

Table 54: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for AGNES – single linkage algorithm after dimensionality reduction.

5.6.2 Validation

Silhouette index

Figure 37 shows Silhouette index plot for AGNES algorithm with single linkage for both clusters after dimensionality reduction. The average value is 0.67, which indicates quite good clustering. Again, it is worth noting that in this method we observe a very unbalanced division of clusters – there are fewer objects in the second cluster.

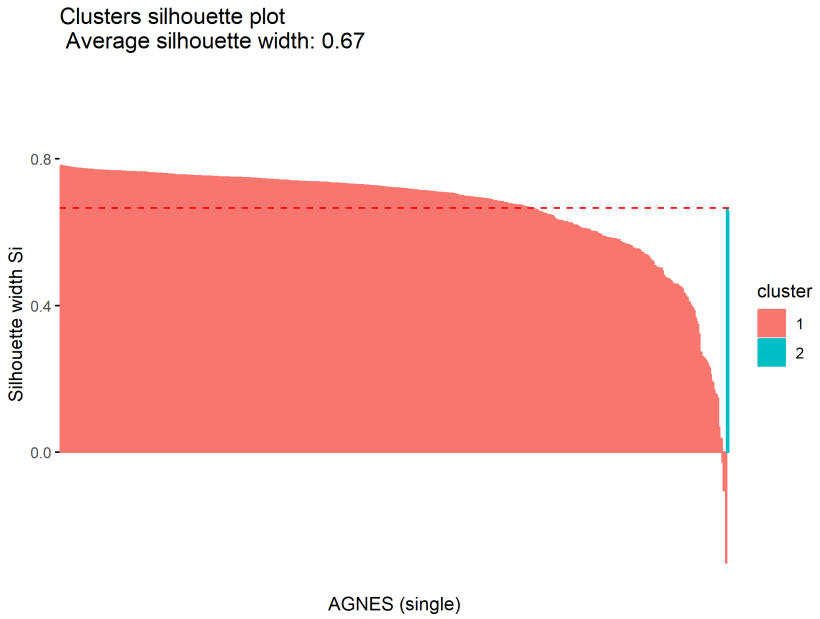

Figure 37: Silhouette index plot for AGNES method with single linkage for 2 clusters after dimensionality reduction.

Cluster labels vs actual labels
Table 55 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the AGNES method with single linkage after dimensionality reduction. It can be seen that the results are analogous to those for the previous AGNES method.

	Data labels	
	B	M
Cluster 1	357	210
Cluster 2	0	2

Table 55: Table of cluster and actual labels for 2 clusters for AGNES – single linkage algorithm after dimensionality reduction.

5.7 AGNES with complete linkage

Figure 38 shows two clusters that were obtained by the AGNES with complete linkage method after dimensionality reduction. Colors of the observations (at the bottom of the dendrogram) indicate the true labels. It can be easily seen that most of the observations are assigned to one of the clusters leaving the second one with few observations.

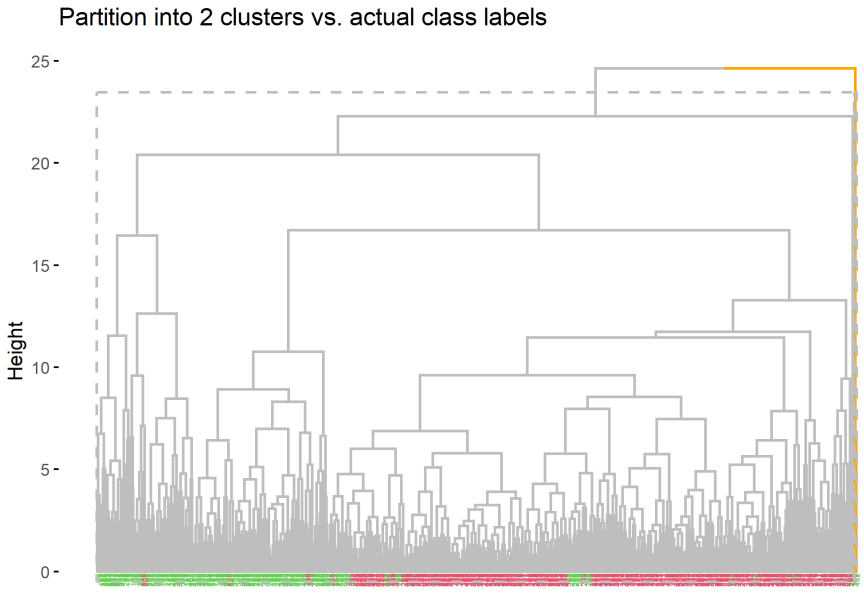

Figure 38: Dendrogram divided into 2 clusters with comparison of real classes for the AGNES method with complete linkage after dimensionality reduction.

5.7.1 Statistics of clusters (for 2 clusters)

By comparing again the statistics for division into two clusters obtained by using the AGNES method with complete linkage after dimensionality reduction (Tables 56 and 57) with AGNES method with average linkage after dimensionality reduction (Tables 50 and 51), it can be seen

that both algorithms give identical statistical values. Therefore conclusions are no different then ones for that case.

Feature	mean	sd	median	mad
PC1	0.0504	3.5483	1.175	3.0889
PC2	-0.027	2.346	0.188	2.0009
PC3	-0.0288	1.6057	-0.161	1.3474
PC4	-0.0133	1.3894	0.0941	1.2988
PC5	-0.0098	1.274	0.0278	0.8922

Table 56: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for AGNES – complete linkage algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	-14.2896	2.8501	-14.2896	2.9879
PC2	7.6529	0.1642	7.6529	0.1722
PC3	8.1667	2.739	8.1667	2.8715
PC4	3.7579	2.0249	3.7579	2.1228
PC5	2.7785	1.514	2.7785	1.5872

Table 57: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for AGNES – complete linkage algorithm after dimensionality reduction.

5.7.2 Validation

Silhouette index

Figure 39 shows Silhouette index plot for AGNES algorithm with complete linkage for both clusters after dimensionality reduction. The average value is 0.67, as for the previous AGNES methods, which indicates quite good clustering. Once again, it is worth noting that in this method we observe a very unbalanced distribution of data points in clusters.

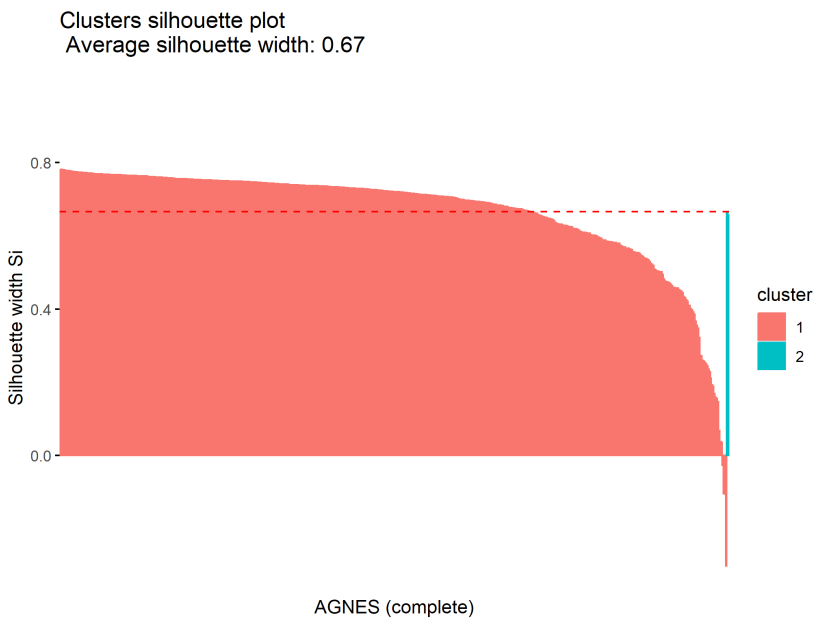

Figure 39: Silhouette index plot for AGNES method with complete linkage for 2 clusters after dimensionality reduction.

Cluster labels vs actual labels

Table 58 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the AGNES method with complete linkage after dimensionality reduction. It can be seen that the results are analogous to those obtained for the previous AGNES methods.

	Data labels	
	B	M
Cluster 1	357	210
Cluster 2	0	2

Table 58: Table of cluster and actual labels for 2 clusters for AGNES – complete linkage algorithm after dimensionality reduction.

5.8 Diana

Figure 40 shows two clusters that were obtained by the DIANA method after dimensionality reduction. Colors of the observations (at the bottom of the dendrogram) indicate the true labels.

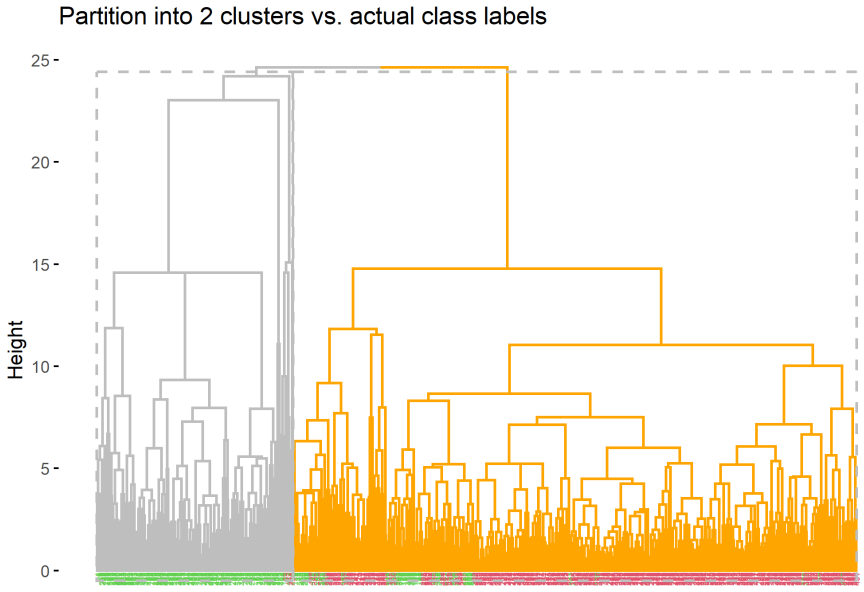

Figure 40: Dendrogram divided into 2 clusters with comparison of real classes for the DIANA method after dimensionality reduction.

5.8.1 Statistics of clusters (for 2 clusters)

Analyzing the cluster statistics tables (Tables 59 and 60) for the DIANA algorithm after dimensionality reduction, it is again visible that the clusters differ significantly in terms of feature values. This suggests that the algorithm successfully divided the observations into subgroups with different properties.

Feature	mean	sd	median	mad
PC1	-5.1539	2.4876	-4.5451	1.9312
PC2	0.3218	3.3124	0.9898	2.8446
PC3	-0.0762	2.3608	-0.3489	1.9829
PC4	-0.0942	1.5147	0.0467	1.2881
PC5	0.01	1.4529	0.0748	0.9207

Table 59: Table of mean, standard deviation, median and mean absolute deviation for Cluster 1 for Diana algorithm after dimensionality reduction.

Feature	mean	sd	median	mad
PC1	1.7953	1.8076	2.0572	1.7932
PC2	-0.1121	1.9557	0.0663	1.757
PC3	0.0265	1.3661	-0.0832	1.2095
PC4	0.0328	1.3683	0.1834	1.2933
PC5	-0.0035	1.2216	-0.006	0.9362

Table 60: Table of mean, standard deviation, median and mean absolute deviation for Cluster 2 for Diana algorithm after dimensionality reduction.

5.8.2 Validation

Silhouette index

Figure 41 shows Silhouette index plot for DIANA algorithm for both clusters after dimensionality reduction. The average value is 0.41, suggesting that the clustering is moderately good, but there is potential for improvement.

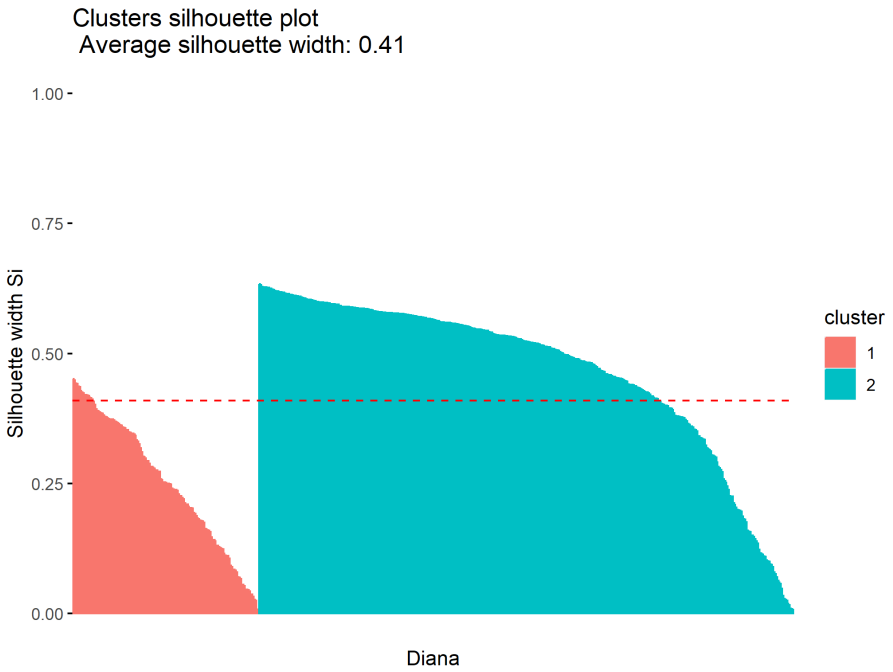

Figure 41: Silhouette index plot for DIANA method for 2 clusters after dimensionality reduction.

Cluster labels vs actual labels

Table 61 shows the true labels (B – benign, M – malignant) compared to the two clusters to which they were assigned using the DIANA algorithm after dimensionality reduction. It is easily noticeable that cluster 1 is dominated by M cases, while the number of B cases is much smaller. Cluster 2 contains most of the bening cases and only a small number of malligant cases.

	Data labels	
	B	M
Cluster 1	4	143
Cluster 2	353	69

Table 61: Table of cluster and actual labels for 2 clusters for Diana algorithm after dimensionality reduction.

5.9 Validation

Table 62 shows the internal validation, stability indices and partition agreement for all methods (except fuzzy c-means) for division into two clusters after dimensionality reduction. Based on it, we can see that for the AGNES methods we obtained the highest values of Dunn and Silhouette index and the lowest Connectivity, which suggests the best cluster structure and assignment of points to clusters. Although these methods achieve the lowest percentage of pairwise case matching (approximately 63%). Worth noticing is that for K-means and PAM we obtain a high agreement of 90%. Values of stability indices are inconsistent in choosing the most stable method. APN and ADM have the lowest values for AGNES with single linkage and AD for K-means. FOM has difficulties to differentiate between all methods leaving us with simmilar values.

	K-means	PAM	AGNES (avg)	AGNES (single)	AGNES (complete)	Diana
Connectivity	65.6742	61.3079	3.8579	3.8579	3.8579	68.2377
Dunn	0.0510	0.0519	0.4434	0.4434	0.4434	0.0331
Silhouette	0.3932	0.3927	0.6660	0.6660	0.6660	0.4095
APN	0.0950	0.1208	0.0049	0.0014	0.2146	0.0579
AD	5.2460	5.2799	6.2713	6.2634	6.2960	5.5392
ADM	0.7033	0.8427	0.1167	0.0664	1.2447	1.1788
FOM	2.0801	2.0807	2.0671	2.0475	2.0767	2.0701
Cases in matched pairs	90.51%	90.50%	63.09%	63.09%	63.09%	87.17%

Table 62: Table describing internal validation, Stability indices and partition agreement for K-means, PAM, AGNES (average, single and complete linkage) and Diana algorithms for two clusters after dimensionality reduction.

Fuzzy silhouette index and partition agreement for fuzzy c-means

Table 63 presents Fuzzy Silhouette index and partition agreement for Fuzzy c-means. The value of Fuzzy Silhouette index is equal to 70% which indicates quite good division into clusters. Moreover, high percentage of pairwise case matching also confirms that conclusion. Partition coefficient also states that our clustering is done well, but the values of partition entropy and modified partition coefficient should be as small as possible in this case, which leaves a room for improvement.

Fuzzy silh. ind.	Cases in matched pairs	Part. Coeff.	Part. Entropy	Mod. Part. Coeff.
70.00%	91.39%	70.70%	45.76%	41.40%

Table 63: Table of fuzzy silhouette index, cases in matched pairs, partition coefficient, partition entropy and modified partition coefficient for 2 clusters for fuzzy c-means algorithm after dimensionality reduction.

5.10 Clustering before and after dimension reduction

By comparing the values of cases in matched pairs before and after dimension reduction for different clustering algorithms, we can see that dimension reduction contributed to improving the effectiveness of the algorithms in assigning cases to clusters for most of the clustering methods tested. Although for some algorithms, such as AGNES, the effect of dimensionality reduction on improvement of the performance may be insignificant. However, the Silhouette Index for all methods for division into 2 clusters after dimensionality reduction is lower than before reduction (also for the fuzzy one).

6 Classification after dimensionality reduction

Now we will focus on classifying our data after applying dimensionality reduction techniques. In the medical context, classification is crucial because it enables precise diagnosis of diseases and prediction of their course. Therefore, the use of advanced data analysis methods such as dimensionality reduction can significantly improve classification performance.

The data was divided into two sets: training and testing. The proportions of classes for the whole dataset, train and test sets are presented in Table 64. It can be seen that the proportions after the division were quite well preserved.

	Class label	
Set	M	B
Dataset	37.26%	62.74%
Learning set	33.77%	66.23%
Test set	44.21%	55.79%

Table 64: Class proportion for dataset, learning and test set.

Using the stepwise method multiple times for the training set, we select various combinations of features for which we will create models. We use the *stepclass* function with the *forward* and *both* directions for three methods: *lda*, *qda* and *sknn*. We obtained the following results:

1. PC1 + PC2 + PC3 + PC4 + PC5,
2. PC1 + PC2 + PC3,
3. PC1 + PC2 + PC5.

For all three features subsets mentioned above, we construct classification rules based on the selected variables. We build 7 different classification models using:

- linear discriminant analysis (lda),
- quadratic discriminant analysis (qda),
- 5-nearest neighbors method (knn),
- logistic regression (logit),
- classification tree (tree),
- bagging method (bagging),
- random forest (randomForest).

Next, we predict class labels using test set on the basis of algorithms trained with learning set.

6.1 LDA

Table 65 shows the group means for all feature subsets for LDA. We can notice that for each subset of features and class (B - benign, M - malignant), the average values differ for each main component (PC).

		PC1	PC2	PC3	PC4	PC5
1 feature subset	B	2.137	-0.403	0.220	0.126	-0.151
	M	-3.309	0.418	-0.455	-0.415	0.111
2 feature subset	B	2.137	-0.403	0.220	—	—
	M	-3.309	0.418	-0.455	—	—
3 feature subset	B	2.137	-0.403	—	—	-0.151
	M	-3.309	0.418	—	—	0.111

Table 65: The group means for all feature subsets for LDA.

Table 66 shows the matrix of coefficients that transform observations into discriminant functions for all feature subsets in LDA. These coefficients reflect the importance of each feature (represented by principal components) in the discrimination process between classes.

	PC1	PC2	PC3	PC4	PC5
1 feature subset	-0.532	0.195	-0.132	-0.152	0.225
2 feature subset	-0.517	0.194	-0.152	—	—
3 feature subset	-0.536	0.206	—	—	0.249

Table 66: A matrix of coefficients that transforms observations to discriminant functions for all feature subsets for LDA.

6.2 QDA

Table 67 shows the group means for all feature subsets for QDA. We can notice that the results are the same as for LDA.

		PC1	PC2	PC3	PC4	PC5
1 feature subset	B	2.137	-0.403	0.220	0.126	-0.151
	M	-3.309	0.418	-0.455	-0.415	0.111
2 feature subset	B	2.137	-0.403	0.220	—	—
	M	-3.309	0.418	-0.455	—	—
3 feature subset	B	2.137	-0.403	—	—	-0.151
	M	-3.309	0.418	—	—	0.111

Table 67: The group means for all feature subsets for QDA.

Table 68 shows a matrix of coefficients that transform the observations so that the within-group covariance matrix is spherical for each group in QDA.

	1.B	2.B	3.B	4.B	5.B	1.M	2.M	3.M	4.M	5.M
PC1	-0.58	0.43	-0.01	-0.04	-0.38	-0.39	-0.04	0.03	-0.05	-0.01
PC2	0.00	-0.55	0.20	0.15	0.17	0.00	0.37	-0.13	0.07	0.06
PC3	0.00	0.00	0.80	0.24	0.11	0.00	0.00	0.54	0.04	0.06
PC4	0.00	0.00	0.00	0.78	0.05	0.00	0.00	0.00	-0.81	-0.06
PC5	0.00	0.00	0.00	0.00	0.79	0.00	0.00	0.00	0.00	0.93

Table 68: A matrix of coefficients which transforms observations so that within-groups covariance matrix is spherical (for each group) for QDA.

6.3 Logistic regression

Table 69 shows the values of coefficients, their significance and Z-score as well as p-value for the logistic regression model for different sets of features. For a significance level of 0.01, it can be concluded that only those coefficients for which the p-value is less than 0.01 are significant. Therefore, for the first subset of features, the coefficients associated with PC1, PC2, PC4 and PC5 are significant, for the second subset of features: PC1, PC2 and PC3, and for the third subset of features: PC1, PC2 and PC5.

		Intercept	PC1	PC2	PC3	PC4	PC5
1 feature subset	Estimate	-0.433	-2.627	1.350	-0.411	-0.755	1.154
	Z value	-1.237	-6.169	4.633	-2.240	-3.079	3.018
	Pr($>\|z\|$)	0.21608	6.88e-10	3.60e-06	0.02512	0.00208	0.00255
	Signif. codes		***	***	*	**	**
2 feature subset	Estimate	-0.587	-2.136	1.152	-0.517	—	—
	Z value	-2.156	-6.698	5.104	-3.312	—	—
	Pr($>\|z\|$)	0.031106	2.12e-11	3.32e-07	0.000927	—	—
	Signif. codes	*	***	***	***	—	—
3 feature subset	Estimate	-0.090	-2.343	1.256	—	—	1.121
	Z value	-0.299	-6.887	5.090	—	—	3.726
	Pr($\|z\|$)	0.765126	5.69e-12	3.58e-07	—	—	0.000195
	Signif. codes		***	***	—	—	***

Table 69: Coefficients and their significance for Logit.
(Signif. codes: '***' - 0.001 ;'**' - 0.01 ; '*' - 0.05 ; '.' - 0.1 ; ' ' - 1)

Table 70 contains the Akaike Information Criterion (AIC) values for the Logit model. The results indicate that the best model among those considered is for the first feature subset with the lowest AIC value of around 77. For the remaining models, the AIC values are about 100 and 89, which suggests a poorer fit of these models to the data.

	1 feature subset	2 feature subset	3 feature subset
AIC	77.237	100.09	89.203

Table 70: AIC values for Logit.

6.4 Classification tree

Table 71 provides a brief summary of the overall fit of the models in the Classification Tree. Analysis of this table suggests that models for all feature subsets have similar complexity parameters and number of splits. However, it is worth noting that the model based on the third subset of features appears to achieve the lowest cross-validation error, suggesting its potentially better performance in classifying new data.

		1	2	3	4
1 feature subset	Complexity param.	0.742	0.043	0.016	0.010
	Number of splits	0	1	3	5
	Cross-validated error	1.000	0.336	0.328	0.289
2 feature subset	Complexity param.	0.742	0.043	0.016	0.010
	Number of splits	0	1	3	5
	Cross-validated error	1.000	0.313	0.313	0.305
3 feature subset	Complexity param.	0.742	0.031	0.016	0.010
	Number of splits	0	1	3	5
	Cross-validated error	1.000	0.289	0.289	0.227

Table 71: A brief summary of the overall fit of the models for Classification tree.

Analysis of Table 72 about feature importance for Classification Tree shows that PC1 is significantly more important than the other components for each set of features. It suggests that this principle component has the greatest impact on the classification process. However, the importance of other principal components should also be taken into account as they can complement the information provided by PC1, which may lead to more accurate and stable classification models.

	PC1	PC2	PC3	PC4	PC5
1 feature subset	118.871	30.149	23.769	11.398	3.967
2 feature subset	118.871	30.149	23.769	—	—
3 feature subset	120.453	28.039	—	—	7.772

Table 72: Feature importance for Classification tree.

6.5 Random forest

Analysis of Table 73 about feature importance based on the Gini impurity index indicates that PC1 is significantly more important than the other components for each set of features.

It suggests that this principle component has the greatest impact on the process classification. However, the importance of other principal components should also be taken into account as they can complement the information provided by PC1, which may lead to more accurate classification models.

	PC1	PC2	PC3	PC4	PC5
1 feature subset	106.615	21.769	19.878	10.477	10.421
2 feature subset	104.781	32.535	31.865	—	—
3 feature subset	116.483	31.758	—	—	20.654

Table 73: Measure of variable importance based on the Gini impurity index.

6.6 Performance

We calculate and compare the accuracy, sensitivity and specificity for all classification models used and all feature sets considered.

The accuracy results are presented in Table 74 and all values are rounded to two decimal places. It can be seen that the studied classes were effectively separated by each classifier. It is challenging to determine which one performed best, because for all of the features sets under examination almost all of them have accuracy values above 95%. However, the worst results were achieved by the QDA classifier for the second set of features - 93.68%, and the best results were also achieved by the QDA classifier, but for the third set of features - accuracy is almost 98%. In conclusion, the following classifiers are the most successful for each features subset:

- first features subset – logistic regression classifier with 96.84% accuracy,
- second features subset – classification tree with 96.32% accuracy,
- third features subset – QDA classifiers with 97.89% accuracy.

Comparing these accuracies with those before dimension reduction (presented in the first part of the project [3]) it can be concluded that the overall classification results have improved, as the lowest accuracy value has increased from around 88% (before reduction, achieved for 5-NN) to over 93% (after reduction). Additionally, most methods achieved an accuracy of over 95% after reduction, where before it was only about 91%. However, the highest accuracy score before dimensionality reduction was 97.37% (achieved by Random Forest) and after performing such it increased to 97.89% (achieved by QDA). Thus, the growth is not significant, but the best performing method (considering accuracy) has changed. It is worth mentioning that before dimensionality reduction, the highest accuracy for QDA was 96.3% (for one of the feature subset).

	1 features subset	2 features subset	3 features subset
LDA	95.26%	95.26%	94.21%
QDA	95.79%	93.68%	97.89%
KNN	96.32%	94.21%	94.21%
Logit	96.84%	95.79%	97.37%
Tree	96.32%	96.32%	97.37%
Bagging	95.79%	94.21%	95.79%
Random Forest	96.32%	94.74%	95.26%

Table 74: Accuracy.

Based on the sensitivity values included in Table 75, it can be easily seen that the sensitivity of the LDA model remains at the highest level (100%) for all tested feature subsets. The sensitivity of the Logit model is also very high for all tested feature subsets, although slightly lower than for LDA. Overall, all tested classification models demonstrate high data sensitivity, suggesting their potential application in medical practice. However, it is the LDA model that stands out as the most stable and effective, maintaining excellent sensitivity to various feature combinations.

	1 features subset	2 features subset	3 features subset
LDA	1.00	1.00	1.00
QDA	0.98	0.95	1.00
KNN	0.99	0.96	0.95
Logit	0.99	0.98	1.00
Tree	0.97	0.97	0.97
Bagging	0.97	0.96	0.96
Random Forest	0.98	0.95	0.96

Table 75: Sensitivity.

Based on the specificity values included in Table 76, we can conclude that all tested classification models show high specificity for all tested feature subsets, although slightly lower than sensitivity. However, Random Tree model stands out as having the highest specificity among the models studied and LDA having the lowest.

	1 features subset	2 features subset	3 features subset
LDA	0.89	0.89	0.87
QDA	0.93	0.92	0.95
KNN	0.93	0.92	0.93
Logit	0.94	0.93	0.94
Tree	0.95	0.95	0.98
Bagging	0.94	0.92	0.95
Random Forest	0.94	0.94	0.94

Table 76: Specificity.

7 Summary

In our analysis, we focused on identifying natural groups of breast cancer patients that reflect different breast cancer subtypes or other relevant patterns. Using various clustering algorithms, we tried to distinguish groups of patients with similar characteristics, which may contribute to a better understanding of the diversity of this disease and to adapt treatment strategies. In our study, we used several different clustering methods, including partitioning methods (K-means, PAM, Fuzzy C-means) and hierarchical methods (AGNES, DIANA). Using them, we tried to find the optimal division of the data that would help to understand the similarities and differences between patients diagnosed with breast cancer. The analysis showed that the K-means and Fuzzy C-means methods achieved the highest percentage of matching pairs of cases, especially for the division into three clusters, which suggests their effectiveness in grouping data. The PAM and DIANA methods also proved to be highly effective. However, the AGNES method showed some differences depending on the connection used, which suggests the need to take this diversity into account during the clustering analysis.

We also performed data dimensionality reduction using principal component analysis (PCA), which aimed to simplify the data structure by reducing the number of explanatory variables while retaining relevant information contained in the data. We found that dimensionality reduction contributed to improving the algorithms' performance in assigning cases to clusters for most of the clustering methods tested. Nevertheless, for some algorithms, such as AGNES, the impact of dimensionality reduction on performance improvement was insignificant. However, it is worth noting that the Silhouette Index for all methods for 2 clusters after dimension reduction is lower than before reduction (also for the fuzzy algorithm). This means that although the efficiency of assigning cases to clusters improves after dimension reduction, the structure of clusters may be less clear or more dispersed.

We then focused on data classification after applying dimensionality reduction using PCA. By comparing the classification results before and after dimension reduction, we found an overall improvement in the classification results. The lowest accuracy values increased from approximately 88% (before reduction) to over 93% (after reduction), and most methods achieved accuracy above 95% after reduction. It is worth noting that the best classification method (in terms of accuracy) changed from Random Forest to QDA after dimensionality reduction.

In summary, data dimensionality reduction using PCA contributed to improving both the clustering and classification efficiency of breast cancer data, which may have important clinical implications in the diagnosis and treatment of this disease.

8 Bibliography

1. my.clevelandclinic.org
2. Data Mining Course materials
3. "Medical diagnostics: Breast Cancer Wisconsin (Diagnostic) Project" – P.Cieślachowska, A. Winiarska
4. "Medical diagnostics: Breast Cancer Wisconsin (Diagnostic) Project Part 2 – Additional Plots" – P.Cieślachowska, A. Winiarska