DATA AND INFORMATION QUALITY PROJECT REPORT

PROJECT ID: 29
PROJECT NUMBER: 1

ASSIGNED DATASETS: abalone, users

STUDENTS:
Gabriele Ginestroni (10687747)
Tommaso Capacci (10654230)

ASSIGNED TASK: Clustering

1. SETUP CHOICES

Chosen ML algorithms:
Since the given datasets presented different characteristics we decided to use different algorithms for each of them.
In particular, for the "abalone" dataset we used:

- **K-Prototypes**: is a clustering method used to cluster mixed-type data (both categorical and numerical features). This algorithm combines the K-Means algorithm for numerical variables and the K-Modes algorithm for categorical variables.
- **Density-Based Spatial Clustering of Applications with Noise (DBSCAN):** is a density-based clustering algorithm that groups data points that are closely packed together (dense regions of data points) and separates data points that are sparsely located.

For the "users" one, instead, we opted for:

- **K-modes:** is a clustering algorithm that is similar to the k-means algorithm but is used for categorical data rather than numerical data. Instead of calculating the mean of a cluster, it calculates the mode, which is the most common value inside of it.
- **Spectral clustering**: is a clustering algorithm that can be used to find clusters in non-linearly separable data. It works by first constructing the Laplacian matrix (a transformation of the similarity matrix). Next, the algorithm finds the eigenvectors of this matrix corresponding to its K lowest non-zero eigenvalues, where K is the number of clusters. These eigenvectors are used as the coordinates of the data points in a new, low-dimensional space. Finally, the data points in this new space are clustered using k-means algorithm.

Chosen ML performance evaluation metrics:
In order to analyse all the relevant aspects we decided to use 2 different types of metrics: the first one gave us general information about how good the results of the application of our ML algorithms were (in fact we also used them to select the values of their parameters on the original datasets), the second one instead allowed us to compare the results of the same algorithms applied on the imputed datasets and on the original ones.
Inside the first category we chose to use:

- **Silhouette score**
- **Calinski-Harabasz score**

While for the second one:

- **Adjusted Mutual Information (AMI)**
- **Completeness score**

Imputation techniques selected:
As for the ML algorithms, different datasets required different imputation techniques due to the nature of their features.
"Abalone" dataset:

Standard) Single imputation using the median of numerical features and the mode of the categorical one

Advanced) Multiple Imputation by Chained Equations (MICE) for the numerical values and a KNN classifier trained to infer the categorical one

"Users" dataset:

Standard) sklearn SimpleImputer to infer the mode

Advanced) Multiple Imputation by Chained Equations (MICE) with KNN classifier

2. PIPELINE IMPLEMENTATION

To save time, we decided to split efforts and focus on different datasets. Anyhow, we concurrently reached very similar results, so we simply chose some common metrics and decided to refer to these.
For the "abalone" dataset the pipeline has been the following:

a) Identify optimal number of clusters using the elbow method with respect to the Silhouette score on the original dataset for K-Protypes and manually tuned eps and minPts for DBSCAN
b) Perform clustering over the original dataset: results have been visualized with both PCA and t-sne techniques to also provide a graphical representation of the newly found clusters
c) Compute "absolute" scores on the clustering results (Silhouette and Calinski-Harabasz indexes)
d) Create NaN-injected versions of the original dataset with the provided script
e) Infer missing values with the chosen imputation techniques: thanks to our choices we were able to perform imputation completely without needing to one-hot encode the categorical feature
f) Assess DQ Accuracy dimension of the imputed datasets with respect to the original one
g) Using the best performing clustering and imputation algorithm found in the previous steps plot Silhouette, AMI and Completeness scores at changing percentages of injected missing values to evaluate their impact on the performance of the ML algorithm
h) Compare ML algorithms and imputation techniques results varying the completeness percentage

For what concerns the "users" dataset, we performed the following main steps:

a) Creating a dirty version of dataset using the provided script to inject NaNs
b) Identify optimal number of clusters using the elbow method on the original dataset
c) Perform clustering over the original dataset to select the two ML algorithms to use in the rest of pipeline. We used Silhouette and Calinski-Harabasz indexes to make this decision. At the end, we selected KModes and Spectral Clustering. During the whole pipeline, we used t-sne technique for visualizing the computed clusters in a low-dimensional space
d) Use chosen standard and advanced imputation methods to infer missing values. For the imputation of the mode, we were able to directly impute the categorically encoded version of the dirty dataset. For the MICE KNN imputer instead, we had to impute the one-hot encoded version of the dirty dataset
e) Assess DQ Accuracy dimension of the imputed dataset with respect to the original dataset
f) Evaluate the clustering results over the imputed dataset. To compare the results of the different imputation methods in a robust way, we decided to compute, for each imputation method, an average Silhouette and Calinski-Harabasz score with respect to all the clustering

algorithms. In this phase, we also computed the AMI and Completeness metrics following the same approach, to compare the closeness of the clusters obtained from the imputed dataset with the ones we got by applying the clustering on the original dataset

g) Using the best performing clustering algorithm found in the previous steps (Spectral clustering), plot Silhouette, AMI and Completeness scores at changing percentages of injected missing values to evaluate their impact on the performances of the ML algorithm, for the two different imputation methods. The same has been done for the accuracy of the imputed datasets to assess this DQ dimension

3. RESULTS

Main results:

Our tests highlight similar results in both datasets. Absolute clustering scores like Silhouette and Calinski are not reliable indicators to evaluate the overall performance of the imputation method. Indeed, with increasing number of missing values, these scores often increase: we suspect that this happens because the imputation methods seem to create "denser" data distributions, thus the resulting clusters can become more cohesive.

Moreover, in the fully categorical dataset, the advanced imputation method outperforms the simple mode imputation at any level of completeness.

Nevertheless, in the dataset with mixed categorical and numerical features, both standard and advanced imputation methods show similar results. This is probably caused by the fact that there are many more samples than in the second dataset.

Note that to assess the Accuracy dimension, we used two different methods. For categorical features we used exact matching, while for numerical features we checked the match within a range defined by a threshold. For this reason, the accuracy plot of the first dataset depends on the selected threshold value.

For what concerns the abalone dataset, we found that any clustering algorithm that accepts inputs with mixed types finds clusters that are very similar to the categorical feature labels. A possible alternative way to proceed could have been dropping the categorical feature and find the optimal number of clusters on the new dataset.

Finally, we found t-sne much better than PCA in visualizing the clusters separation in a 2D space.

Here are reported some clustering results on the abalone dataset, visualized with both techniques:

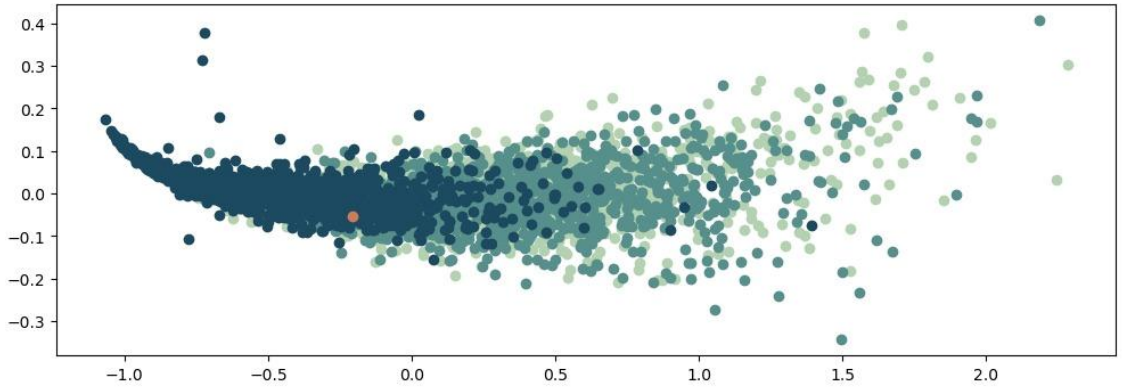

Figure 1 – PCA visualization obtained by dropping the categorical feature

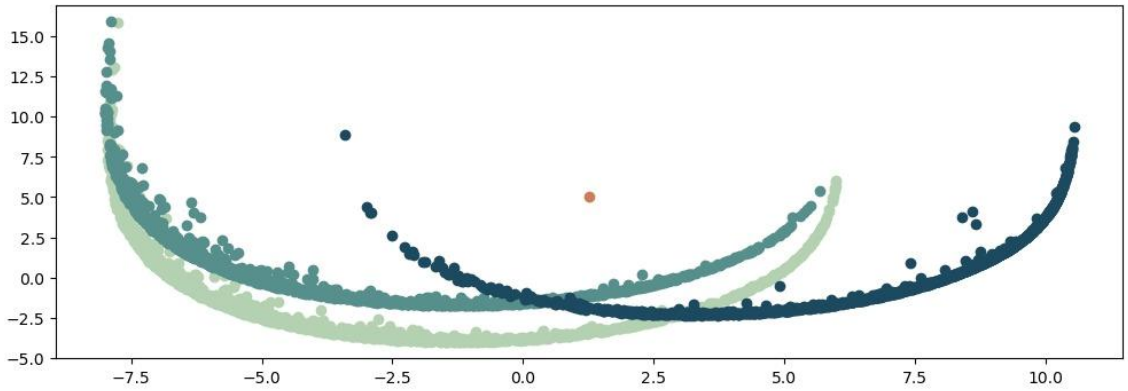

Figure 2 – PCA visualization obtained by computing the distance between points using the Gower metric

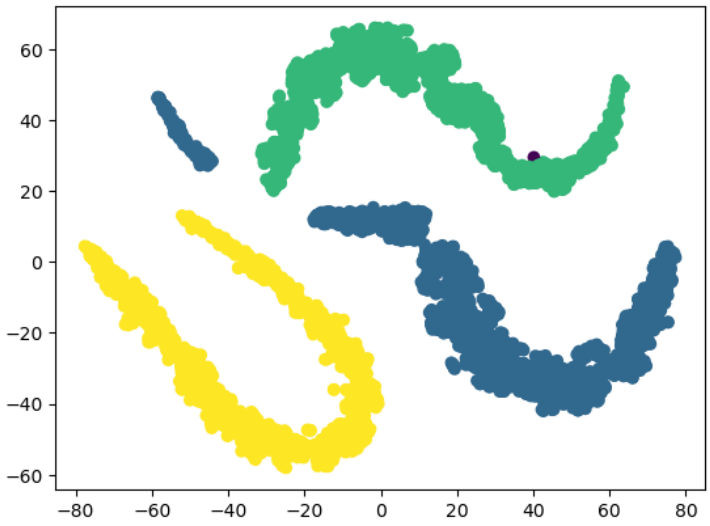

Figure 3 – tSNE visualization on DBSCAN algorithm results

Here instead is reported one of our results of clustering over the "users" dataset:

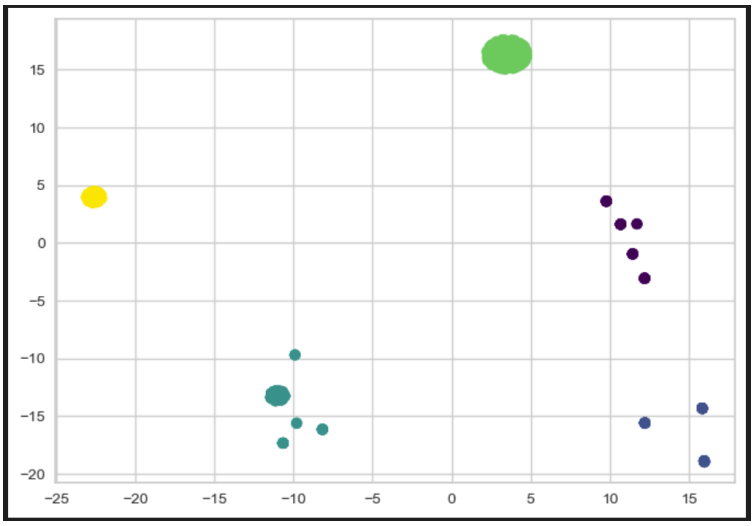

Figure 4 – tSNE visualization with MICE imputer

ML performance comparison between the imputation techniques we have implemented:
Here we'll show the plot of the results of different ML algorithms combined with different imputation techniques on all versions of our datasets.

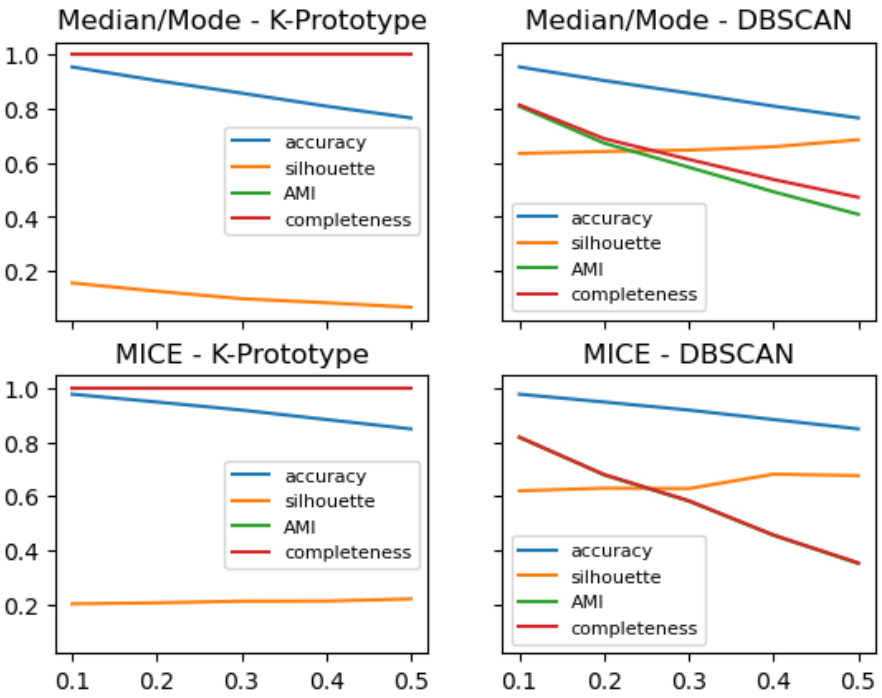

Figure 5 – Results on the "abalone" datasets: we can see that K-Prototype seems to find always the same clusters with very low silhouette, while DBSCAN produces different results.

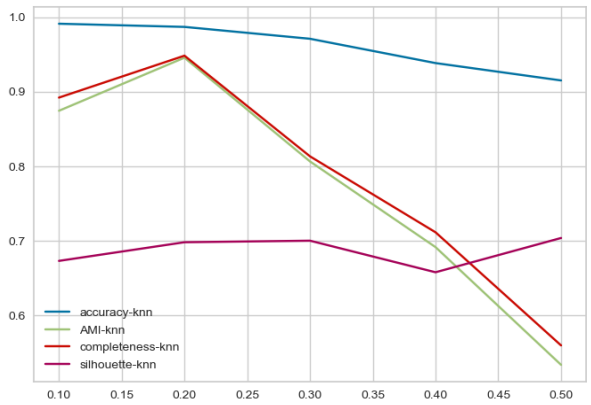

Figure 6 - Results on the "users" dataset using MICE imputation and Spectral Clustering as ML algorithm

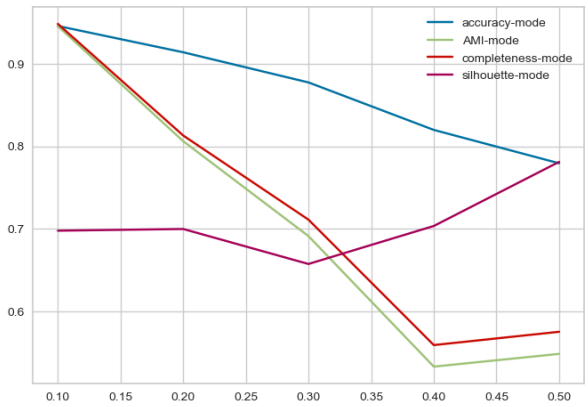

Figure 7 – Results on the "users" dataset using SimpleImputer (with mode) and Spectral Clustering as ML algorithm

For both datasets we can clearly see that MICE imputation results to be the best choice in terms of accuracy.

Finally we can clearly see what we previously mentioned about the Silhouette score: for lower degrees of completeness we experience an increase in this score, while AMI and Completeness scores decrease.