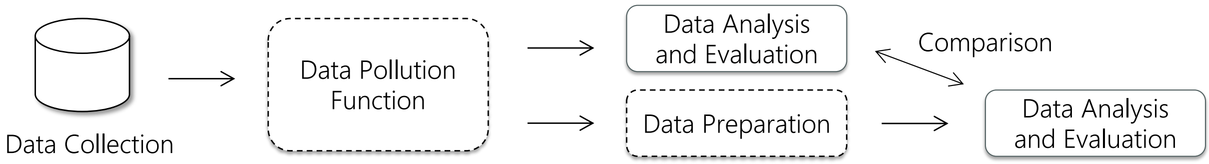
GUIDELINES FOR DIQ PROJECTS

The project gives you the opportunity to obtain a maximum of **3 additional points**

EVALUATION
Deliver a report (few pages) and the code you made (.py, or .ipynb) — more details on writing the report at the end of the document

DEADLINE 24/01/2024

PIPELINE



1. Data Collection GIVEN —> A_data_collection.py
Fixed default parameters
Can be changed according to the needs of the DQ issue/s to be injected
If you change parameters: justify why you have changed the default settings inside the report

2. Data Pollution Function TODO
Inject errors/values related to the assigned DQ issue at different (%)
Combined with dataset.make to inject the assigned DQ issue/s

3. Data Analysis and Evaluation GIVEN —> D_data_analysis.py and E_plot_results.py
Metrics: Performance, Distance train-test Performance, Speed
(only Performance and Speed for Clustering)
Creation of plots and tables with the numeric results
N.B Add also a table/tables with the numeric results inside the report TODO

4. Data Preparation TODO
Apply different DQ improvements to correct the injected DQ issue
Could be requested or not, depending on the assigned DQ issue/s

5. Data Analysis and Evaluation
(Perform this phase again if data preparation is performed)
GIVEN —> D_data_analysis.py and E_plot_results.py

6. Compare the obtained results TODO
Write your considerations on the obtained results inside the report
N.B You must justify the results you have obtained, not only describing the plots and the table, but trying to think about the reasons why you got those results

Inside main.py you can find:
— A recap of the main phases to perform
— An example of **1 experiment** for classification, regression and clustering
N.B. The experiment is a toy example in which 10 dataset with a different number of samples (1000+1 at each iteration) has been generated and the respective performance metrics has been extracted with the generation of plots

EVALUATION METRICS

CLASSIFICATION: Performance (F1 weighted score), Distance between train & test performances (F1 weighted score), Speed of training

REGRESSION: Performance (RMSE Root mean squared error), Distance between train & test performances (RMSE Root mean squared error), Speed of training

CLUSTERING: Performance (Silhouette coefficient), Speed of training

DQ ISSUES GUIDELINES

Here you can find guidelines on how to generate the final output for the project, based on the DQ issue that has been assigned to your Group/you.

N.B For each DQ issue you are requested to generate 10 different experiments

1 experiment correspond to the generation of a plot/with related tables.
For 1 experiment, the DQ issue assigned to your Group/you must be injected in your data with a different %. Following, detailed guidelines on which are the expected results for each assigned DQ issue:

Completeness
— consider both Missing Not at Random (MNAR) and Missing Completely at Random (MCAR) missing value distributions
— 10 experiments with a different distribution of missing values: for example, you can distribute the missing values uniformly across all columns, or change the percentage of missing values for columns with different informativeness (changing the default parameter in the data collection). Try to think of a way to simulate also MNAR distributions!
— For one experiment: varying % of missing values, for example from 5% to 50% (with an increasing step of 5%) of injected missing values

Accuracy
— 10 experiments with a different range of outliers' distance: for example, injecting outliers slightly outside the original range and then gradually further away
— For one experiment: varying % of outliers, for example from 5% to 50% (with an increasing step of 5%) of injected outliers

Feature Dependency
— 10 experiments with a different number of correlated features: for example, injecting more and more features which are correlated to the originals
— For one experiment: varying level of correlation, for example to 0.5 to 1, based on a fixed correlation coefficient (ex. Pearson, Spearman, Kendall…)

Variables Types
— 10 experiments with different combinations of numeric, categoric and boolean features
— Note that dataset.make creates only numeric dataset

Distinctness
— 10 experiments with a different number of polluted features: for example, substituting/polluting a different number of the original features, or adding more and more features
— For one experiment: varying % of distinctness, from very low distinctness (all values constant) to very high (all unique values)
— You can also try to combine low-distinctness and high-distinctness columns in some experiments

Duplication (not-exact)
— 10 experiments with different % similarity: for example, creating and adding non-exact duplicated rows, based on a fixed similarity measure
— For one experiment: varying % duplicated rows, for example from 5% to 50% (with an increasing step of 5%) of duplicated rows

Dimensionality
— 10 experiments with fixed number of samples and varying number of features
— 10 experiments with fixed number of features and varying number of samples
Obviously you have to change the default parameters of the data collection

N.B. You can also propose your own solution, it is not mandatory to follow the examples given for each DQ issue.

Data Preparation is requested for: Completeness, Accuracy, Duplication
Select one/more data preparation techniques that we have seen at exercise lectures and implement it/them inside the pipeline.
For Completeness apply Data Imputation;
for Accuracy apply Outlier Detection (and try to deal with the detected outliers, dropping or correcting them);
for Duplication, detect the duplicates using record linkage and simply drop them once they has been detected.

GUIDELINES ON WRITING THE PROJECT REPORT

PROJECT ID
ASSIGNED DQ ISSUE/S AND ML TASK
STUDENTS (NAME SURNAME ID)

1. SETUP CHOICES
   Describe the choices made:
   In the data collection phase
   The experiments you decided to implement

2. PIPELINE IMPLEMENTATION
   Description of the steps you performed
   Describe the pipeline you executed, paying particular attention to the steps you implemented

3. RESULTS
   Discussion on the main results obtained
   Describe the results you have obtained by showing plots and tables
   Justify you results: you must justify the results not only describing the plots/tables, but trying to think about the reasons why you got those results

N.B. Together with the report you must submit also the code you have implemented so far (.ipynb or .py)