IML'22 Project: Boston Housing Dataset

Background. Predicting the price of housing is important for a variety of reasons. For homeowners, understanding the value of their homes can help them make informed decisions about whether to sell, how much to ask for, and whether to invest in renovations or other improvements. For buyers, understanding the price of housing can help them make informed decisions about which homes to consider, how much to offer, and whether they can afford a particular home. Additionally, predicting the price of housing can be useful for real estate professionals, who may use this information to help clients buy or sell homes, or to make investment decisions. Machine learning can play a role in predicting the price of housing by providing more accurate and reliable estimates than traditional approaches.

Dataset. The Boston Housing Dataset is a publicly available dataset, downloadable from here, which contains 506 records of 14 attributes of per capita crime rate by town, the average number of rooms per dwelling, index of accessibility to radial highways, the pupil-teacher ratio by town, the lower status of the population, among a few more relevant attributes. Your task is to predict the Median value of owner-occupied homes in $1000's.

Requirements.

1. Formulate the machine learning problem by identifying the task (classification or regression), data, and challenges.
2. Perform Exploratory Data Analysis. This includes:
    - Plot relationship between variables
    - Identify correlated features or the ones that highly correlated with the label/outcome, if any
    - Perform data imputation for missing variables
    - Encode your outcome to a one-hot vector, if needed
    - Remove redundant variables by dimensionality reduction techniques
3. Consider splitting the data into Training, Validation, and Testing sets. The suggested split is 70%, 10%, and 20% held-out testing set, respectively. (same split for all tasks)
4. Build and develop the following models (tasks):
    - **Task#1:** Clustering in an unsupervised fashion
    - **Task#2:** Logistic/Linear Regression according to your formulation in step 1
    - **Task#3:** Neural Network for Classification/Regression according to your formulation in step 1

5. For each task, run the model *with* and *without* processing the data, e.g., without normalization or dimensionality reduction, and compare the model's performance after you normalize and/or reduce the dimensionality of the data.
6. For each task, show how you performed the *model selection*. For example, demonstrate the performance of variants of your model with different hyper-parameters, e.g., number of clusters and initialization when it comes to clustering methods.
7. For each task, perform a 5-fold Cross Validation.
8. For each task, run the corresponding evaluation metrics on each fold to demonstrate the performance of your model on the held-out testing set

Submission.

- **Prepare** a well-documented Jupyter notebook demonstrating the following sections:
    - Your name along with your Uni-ID and task assignment within the group
    - The background of the given task
    - Sample of your dataset
    - The rationale behind your ML formulation
    - The exploratory data analysis and data split
    - The developed models (Task #1-#3) including the model selection (hyper-parameters) and their results
    - A brief comparison between different models + concluding remarks
    - Lessons learned from this project
- **Consider** the following naming convention when you name your Jupyter notebook, `BostonHousingDataset_G##`, and consider replacing ## with your group number, e.g, if your group number is 5, then the name of your file should be `BostonHousingDataset_G05.ipynb`
- **Submit** the Jupyter notebook to the following folder https://uni-bonn.sciebo.de/s/tO0R2lXI1RamUWB which allows you to only upload files. You can upload the same file multiple times in case updates/modifications were required.
- **The submission deadline** is 18th Jan. 2023 at 11:59 PM

Grading.

- **Individual-level grading (65%) + Group-level grading (35%)**
    - The problem addressed (clearly stated and understood) – *Group-level*
    - ML problem formulation (rationale) – *Group-level*
    - Exploratory Data Analysis – *Group-level*
    - Task Completion and Findings – *Individual-level*
    - Interpretation and conclusion – *Individual-level*
    - Limitations – *Group-level*
    - Documentation, Submission, and Presentation – *Group-level*