Assignment 3

Assignment 3 consists of a written part and coding that can be incorporated directly into IPython Notebook files. The programming part can be submitted either as IPython Notebooks (recommended) or as stand-alone scripts. Do not include absolute paths! All references to the external files should have relative paths. Python interpreter and imported libraries should be compatible with the latest Anaconda distribution (https://www.anaconda.com/).

Written part (30 points)

You have a small dataset that describes to irises types – setosa and virginica. Build a decision tree using greedy strategy using

1) Gini impurity index
2) Information gain

5.1	3.5	1.4	0.2	Iris-setosa
4.9	3.0	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
6.3	3.3	6.0	2.5	Iris-virginica
5.8	2.7	5.1	1.9	Iris-virginica
7.1	3.0	5.9	2.1	Iris-virginica

Programming part (70 points)

Download hear disease dataset https://archive.ics.uci.edu/dataset/45/heart+disease. You would be predicting diagnosis of angiographic disease status (variable name is num).

-- Value 0: < 50% diameter narrowing

-- Value 1: > 50% diameter narrowing

Part 1 (30 points)

1. Build a decision tree model. Usage of packages is allowed. Run 10-fold cross-validation. Report F1 score, accuracy, and AUROC for the model.
2. Use Random Forest model to make predictions. Run 10-fold cross-validation. Report F1 score, accuracy, and AUROC for the model.
3. Use Boosting (e.g., LGBM, XGBoost, etc.) model to make predictions. Run 10-fold cross-validation. Report F1 score, accuracy, and AUROC for the model.
4. Compare three models performance

Part 2 (40 points)

1. Implement K-Means clustering manually. Do not use packages with ready-made implementation for this problem.
2. Estimate the number of clusters. You can visualize the data using any method discussed in lectures.
3. Run your K-Means implementation. Visualize the results.
4. Run spectral clustering. You can use scikit-learn implementation. Visualize the result, compare to K-means.