4DV510 / Linnaeus University

Project Requirements 1:

Clustering

Introduction

Clustering techniques process complex datasets in order to find cohesive groups of points, in order to help the analyst to make decisions regarding different groups of similar points. Although there are a multitude of available clustering techniques, we can identify some common properties in most of them: (a) they take structured data as input; (b) they can usually be customized with different hyper-parameters; and (c) the output is a description of the extracted clusters. In addition to that, most (but not all) will use distance measures to compare pairs of points, and we can also find many "validation" (or "quality") measures that are independent and external.

Requirements

You should start implementing the first few components of your data mining framework by focusing on structured data, clustering techniques, distance measures, and quality measures for clustering.

1. **Create a component for a *Dataset*.** It should expose methods to get the data points (rows) and the features (columns).
2. **Create a component for a *Distance Measure*.** It should take as input two data points and return as output a single value. Optionally, some distance measures may accept parameters also, but we will discuss this in a later lecture.
3. **Create a component for a *Clustering* technique.** It should accept as input a *Dataset* and optional hyper-parameters. The hyper-parameters must not be hard-coded; every different technique will have different ones. One parameter should be a *Distance Measure* (even though not all techniques use it, most will). The output can be a list of lists of data points, for example, depending on your language.
4. **Create a component for a *Quality Measure* for clustering.** It should take as input both the results of a clustering technique and a Dataset, and output a single value.

Notes:

Remember to write also at least three different implementations for each component. In order to test your new requirements, make sure you can run a pipeline with them where you programmatically test the three different implemented techniques, with three different datasets, with three different quality measures for each, and report the results. This will result in at least 27 different runs with the same pipeline (ignoring possible hyper-parameter searching, which you could also do).