6 Intrinsic Dimension and Density Estimation

You can use external libraries for linear algebra operations but you are expected to write your own algorithms.

6.1 Exercise 1

Using the `dry_beans_dataset` as we did in previous laboratories (ie. follow the same preprocessing steps but **do not** perform a train-test split), program your own implementation of the two-NN estimate for the Intrinsic Dimension.

Is the result compatible with what you would expect from an analysis of PCA's spectrum?

6.2 Exercise 2

Using the following code, create a one-dimensional dataset of size $N = 100$.

```
X = np.concatenate(
(np.random.standard_t(1, int(0.04*N))-3.5,
np.random.normal(5, 1, int(0.48 * N)),
np.random.normal(7.5, 1, int(0.48 * N))
))[:, np.newaxis]
```

Compute the density estimation with your implementations of:

- Histogram Density Estimation (Freedman Diaconis rule)
- Kernel Density Estimation (KDE) - Gaussian kernel (Silverman's rule)

Notes

- You can use the `sklearn.neighbors.NearestNeighbors` class. If you have datapoints for which the distance to the first NN is null, you can ignore them (it is a very small fraction of the whole dataset).
- For the two-NN estimate, you can check if your results align with the ones provided by the following package, developed by a team in SISSA. `https://dadapy.readthedocs.io/en/latest/`