Automatic detection of hyperparameters in datasets

Predictions for nearest neighbor models

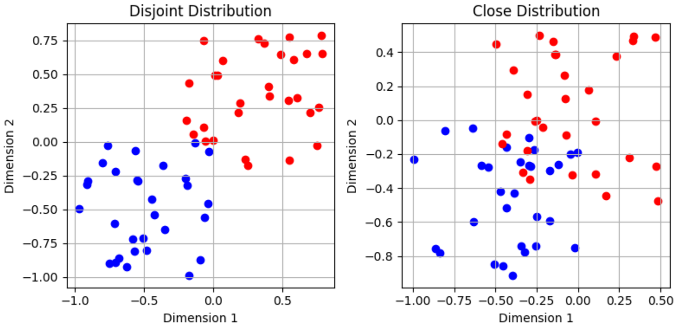
Alexis Rosenfeld & François Delafontaine

Fall 2024

1 Research Question and Motivation

In this project, we will work on the KNN algorithm and more specifically on its only hyperparameter, 'k'. But what are hyperparameters? The scikit-learn documentation defines hyperparameters as: "aspects for configuring model structure that are often not directly learned from data." Because ML algorithms tend to consider many data dimensions, we understand that mathematically modeling the optimal parameters of models, the hyperparameters, is not an easy task. The phrase "often not directly learned" highlights the common way to find them: experimentation.

Nevertheless, we observe that for simple ML models like KNN, links between "the data" and the hyperparameter 'k' exist. More formally, we can easily imagine a relationship between the distribution of the data's dimensions and 'k'. For example, the proximity between the distributions of different categories' features:



Disjoint Distribution

Close Distribution

Figure 1: Link between optimal k and the distribution of features across categories (k=1 in the first case but k=3 in the second case due to the "closeness" of the parameter's distributions among the categories)

Our question now is: how can we, using ML, formalize the relationship between dataset statistics and 'k'?

2 Dataset

We create a dataset $D = y + X$, where y is the label of our model and X is the matrix of predictors. We then use cross-validation to estimate the best k value.

We are then able to build our target dataset $D_t = y_k + X_s$, where y_k represents the different values of k we found for our dataset D using different X, whose statistics become the new matrix parameters X_s.

X_s could, for instance, be formed from the proximity between the distributions of different categories' features, the noise of the features, or the number of data points in the features.

Formally, we could hypothesize:

- The closer the distributions of different categories' features, the higher the k value.
- The more *noise* a dataset has, the higher the 'k' value.
- The more data points a dataset has, the higher the 'k' value.
- Regarding data length, we can confirm a *rule of thumb* (StackOverflow, 16.10.2024): $k = \sqrt{n}/2$.

To summarize, the dataset D_t is generated thanks to the optimal KNN hyperparameter of various D generated using arbitrary y and X defined according to some distributional features and statistics found in X_s.

3 Model

Due to the nature of our 'y', with the number of values unknown, we should only consider a regression model (with the output rounded). We have already defined the basic model:

$$k = \text{distribution closeness} + \text{noise} + \text{data length}$$

Once relevant variables have been identified, along with a formula to relate them to 'k', the focus should be on refining that model.

4 Discussion (Problems)

Our observation so far is that the 'k' value should vary widely, and, therefore, our dataset should be challenging, resulting in our model's precision being relatively poor. To evaluate the model, we should not only check if the value is exact but also how close our predicted value is to the actual optimal 'k' value.