IN-STK5000 – Adaptive Methods for Data-Based Decision Making

Luca Attanasio

August 2019

1 Introduction

In the following paper, dermatological pathologies are analyzed using *K-NN algorithm*: https://archive.ics.uci.edu/ml/datasets/dermatology.

There are six classes of possible diseases that can be assigned to a person. The classes are:

- psoriasis
- seboreic dermatitis
- lichen planus
- pityriasis rosea
- cronic dermatitis
- pityriasis rubra pilaris

In the dataset there are many features which can be used to predict diseases by training a model.

The first goal is to plot some data to see if there is a correlation between the disease and the given feature.

The values of the features for from 0 (low, the patient doesn't suffer from the selected feature) to 3 (high, the patient suffers from the selected feature).

For example, we can evaluate the correlation between the diseases and the age of a person. In general we can see that most dermatological pathologies appear between the age of 15 and 70 Figure 1.

We now focus on a particular disease: psoriasis. Since I suffer from Vitiligo, I prefer to analyze this disease more in depth which is cured similarly by using UV-B lamps.

In Figure 2, we notice that *psoriasis* is related to the family history of the patient in some cases. We also notice it is more related to family history than other pathologies.

To evaluate if a patient suffers from psoriasis, a good way is to analyze the patient's erithema Figure 3. Surprisingly, we notice that they mostly don't suffer from itching Figure 4.

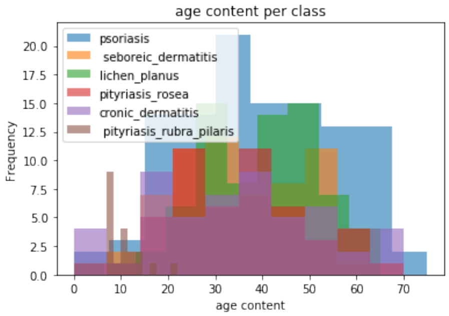

Figure 1: Age frequency

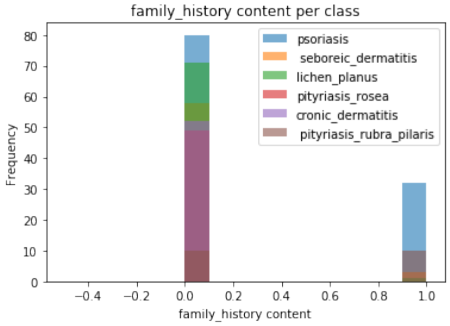

Figure 2: Family history frequency. 1 stands for related, 0 stands for unrelated.

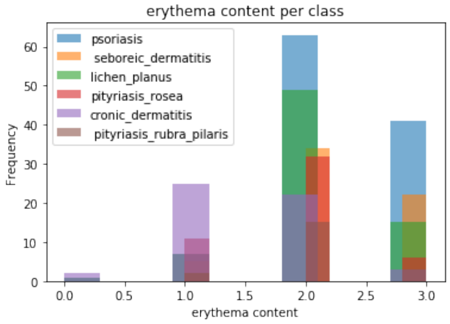

Figure 3: Erithema

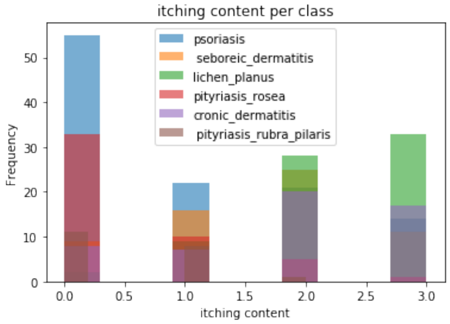

Figure 4: Itching

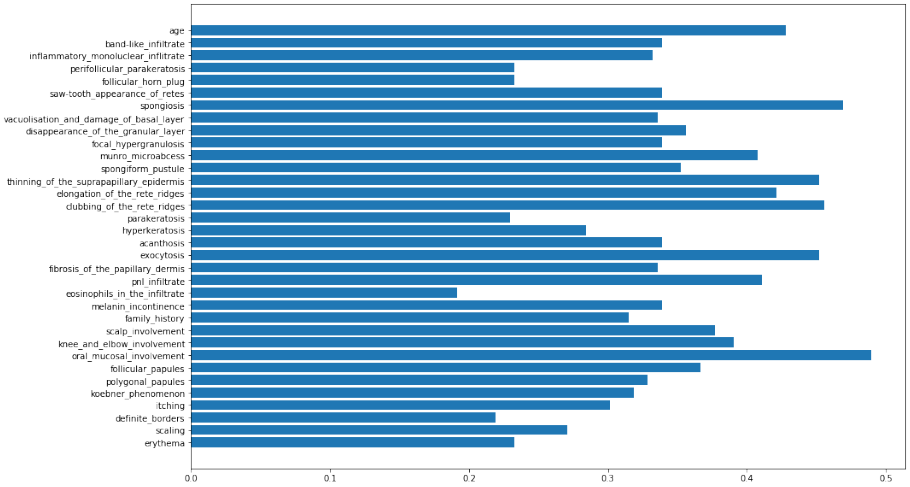

Figure 5: Feature importance on the training set

2 Find Feature Importance by Achieved Accuracy Score

The importance of the features can be analyzed by running the algorithm (i.e. with k=5) on single features. This allows to pick the best features that represent the diseases, especially by looking at the test set results. The best features, which help us to relate the patient to the pathology, have higher values (close to 1) in range a from 0 to 1. Some features are too technical and we can't get into detail with those, but the involvement of different parts of the body seems to be an important way to detect the disease.

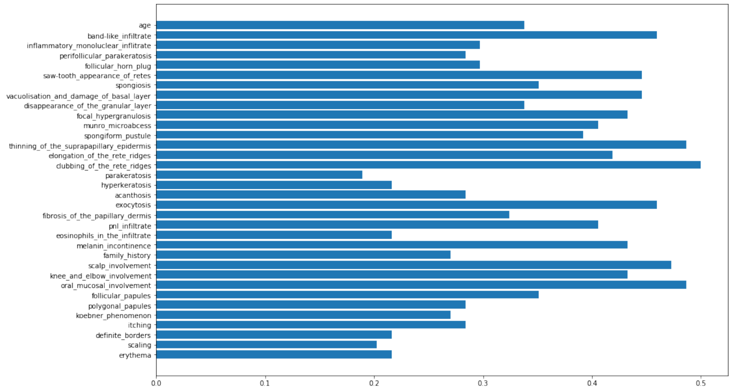

Figure 6: Feature importance on the testing set

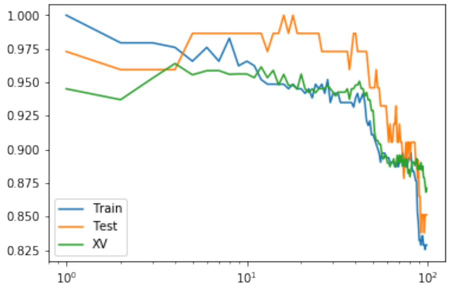

Figure 7: Best value of k.

3 Picking the best k

Before picking the best k, feature scaling (in a range between -1 and 1) is performed. This allows to train the model in a more efficient way. To pick the best k, the algorithm runs with different values of k (i.e. ranging from 0 to 100) and the accuracy score is evaluated. In particular, it is important to check the test set score. Any number below 85 is a good option for the value of k, since it obtains a testing accuracy of above 90% Figure 7.

In addition, from Figure 7, we can tell which is the best k for each evaluation technique:

- training: 1
- testing: 4
- cross-validation: 16

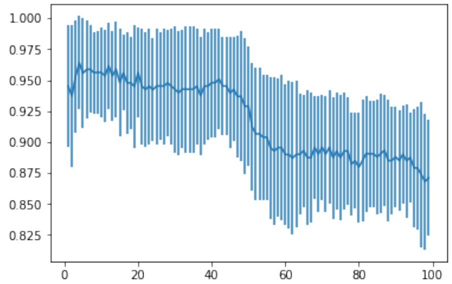

Figure 8: Error mean and std in cross-validation.

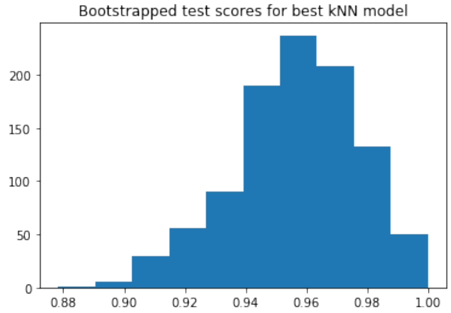

Figure 9: Bootstrapping on the test set.

4 Cross validation

We can further validate our model using cross-validation, which runs the algorithm multiple times. This can be done in conjunction with choosing the best k, since it allows a better estimate of the best value of k. The algorithm also has a random component if we re-scale the features when splitting between test and train set. The mean and standard deviation plot allows to pick the best k visually. With cross validation *restrictions*, the best values of k are between 5 and 40 Figure 8.

5 Bootstrapping

By plotting the distribution of the scores predicted by the model on random samples from the test set, the cross validation model, picked up earlier (the one with k=16 picked from cross validation) can be further analyzed. The results show that the model is achieving good performances. In fact, when looking at the test set, the accuracy is between 0.88 and 1. In addition the mean score is high: 0.96 Figure 9. We can also take a look at the training set plot, which gives similar information Figure 10.

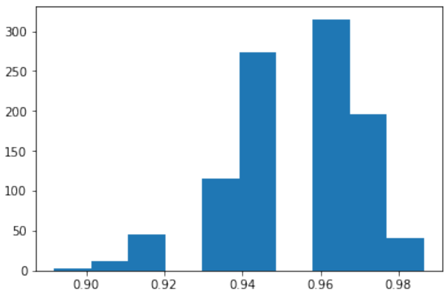

Figure 10: Bootstrapping on the training set.

6 Conclusions

The K-NN algorithm seems to be the right fit to train the dermatology dataset since the accuracy on the test set is high and the computational load is low.