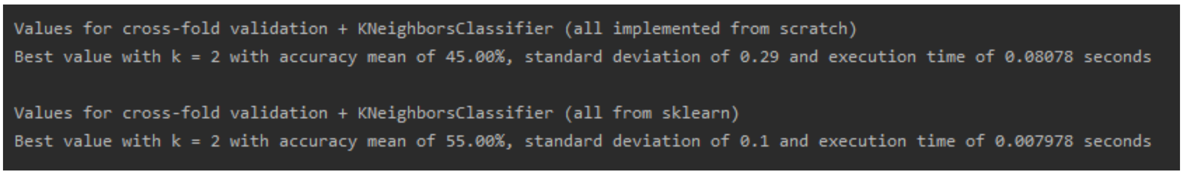
CSC-691 — Data Mining
Assignment 3

Esteban Murillo

Saturday 5th October, 2019

Summary

Both *k-Nearest Neighbors* algorithms (the one self-implemented and the one from *sklearn*) proved that they are not the right classifier for this kind of problems. This shows that we should be careful, if an algorithm performs very well for scenario X does not mean that we are going to get the same performance in scenario Y.



```
Values for cross-fold validation + KNeighborsClassifier (all implemented from scratch)
Best value with k = 2 with accuracy mean of 45.00%, standard deviation of 0.29 and execution time of 0.08078 seconds

Values for cross-fold validation + KNeighborsClassifier (all from sklearn)
Best value with k = 2 with accuracy mean of 55.00%, standard deviation of 0.1 and execution time of 0.007978 seconds
```

Figure 1: Yielded results for *k-Nearest Neighbors* algorithm in-house vs *sklearn* (tested with few review files)

Notes

- Unfortunately, it was not possible to implement the *Naive Bayes Classifier* algorithm from scratch
- Due to a bug, *Naive Bayes Classifier* yields an unexpected value for accuracy (very low), but an example without cross-validation is provided
- Due to some memory management problems, if the number of files read is too large, the program might fail
- In order to change the location of the positive and negative reviews, modify the respective value in *global_variables.py*
- Remember to run the code using the **version 3 of the *Python* interpreter**

- It is neccesary to download extra packages from the *nltk* library in order for the program to run

References

[1] *Bag of words (BoW) model in NLP*. Retrieved on October 4th from https://www.geeksforgeeks.org/bag-of-words-bow-model-in-nlp/

[2] *Building a k-Nearest-Neighbors (k-NN) Model with Scikit-learn*. Retrieved on September 18th from https://towardsdatascience.com/building-a-k-nearest-neighbors-k-nn-model-with-scikit-learn-51209555453a

[3] *Introduction to Data Mining*. Retrieved on September 6th.