Model Selection and evaluation

Pablo García Fernández
(pablo.garcia.fernandez2@rai.usc.es)

Saturday 15th January, 2022

Abstract

The aim of this report is to present the work carried out during the execution of the exercises of the unit 4 assignment, as well as the obtained results. In addition to this report, we also attach the developed .py code, the obtained figures and a .txt file with the results.

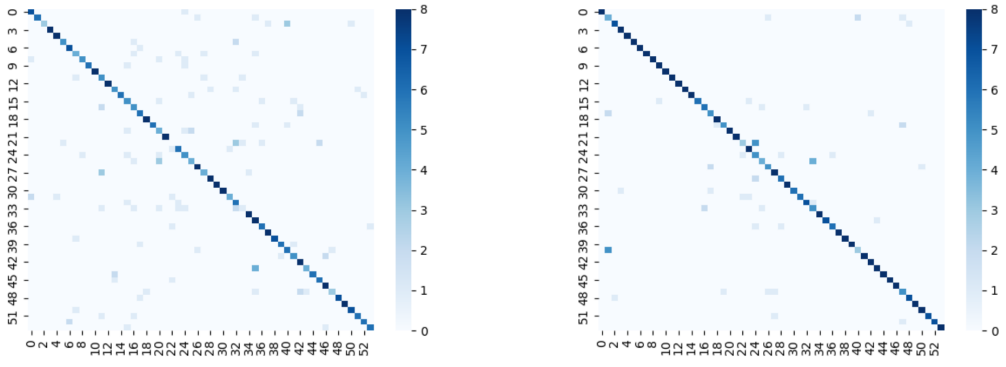
Exercise 1 and 2: *Run 1-NN over each texture feature and report accuracy, kappa and confusion matrix.*

We start from the already computed features: **Coocur** (with the files *trainCoocur.dat* and *testCoocur.dat*), **LBP** (with the files *trainLBP.dat* and *testLBP.dat*). If we train a 1-NN classifier on the training split, we obtain the following test split results ("Tab. 1").

Table 1: Results for the 1-NN over both datasets

Dataset	No. features	Accuracy (%)	Kappa (%)
Co-occurrence matrices	12	75.00	74.53
Local binary patterns	10	87.50	87.26

Since the total number of classes is very large (53), "Fig. 1" shows the confusion matrix as an image for both sets of texture features.



CM. CooCur CM. LBP

Figure 1: Confusion matrix for CooCur and LBP. 1-NN

From the results it can be concluded that for this particular dataset Contrib TC 00006 (formed by images representing textures of different objects) LBP has a higher discriminative power when differentiating textures.

Exercise 3: *Include cross-validation with 4 folds.*

We first concatenate the training and test sets to perform a proper cross-validation. On the full feature set, we create the 4-fold partitions using the `create_folds` function. Since parameter tuning is not necessary for this exercise, the validation and training partitions are grouped together. Finally, we run a 1-NN classifier on the training/validation partitions and test on the remaining one. We repeat this process 4 times (changing the test partition each time). In the end, accuracy, kappa and confusion matrix results are the average of the 4 evaluations. This process is repeated for both CooCur and LBP datasets. "Tab. 2" shows the obtained results.

Table 2: Results for the 1-NN over both datasets using 4-folds cross validation

Dataset	**No. features**	**Accuracy (%)**	**Kappa (%)**
Co-occurrence matrices	12	71.76	71.23
Local binary patterns	10	86.69	86.44

As before, "Fig. 2" shows the confusion matrix as an image for both sets of texture features.

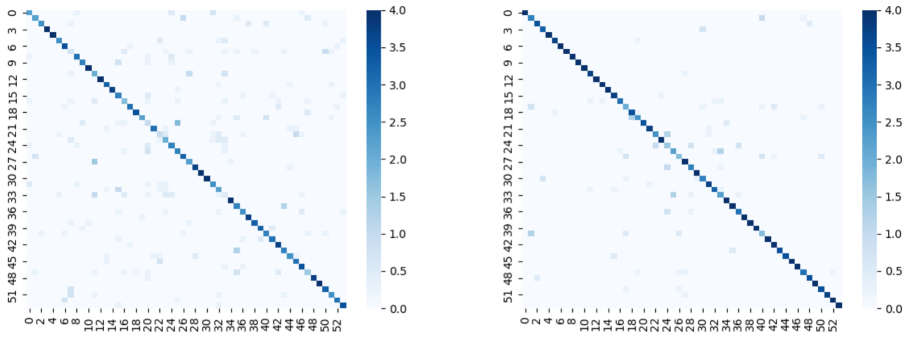


Figure 2: Confusion matrix for CooCur and LBP. 1-NN 4-folds cross validation

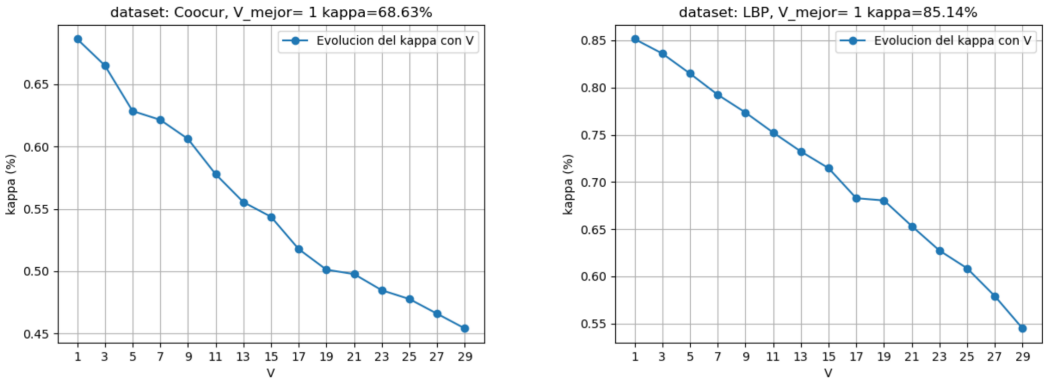
Since cross-validation averages different model test runs, the results obtained are more robust and less biased. In fact, if we compare "Tab. 1" and "Tab. 2", we can see how accuracy and kappa results have been slightly reduced. This means that, from the previous exercise results, the model seems to work better than it actually does.

Exercise 4: K-NN 4-folds cross validation with K (no. neighbors) tuning.

We first concatenate the training and test sets to perform a proper cross-validation. On the full feature set, we create the 4-fold partitions using the `create_folds` function. Then, we

adjust the number of neighbors by training a k-NN classifier on the training partition and testing it on the validation partition (using the same cross-validation approach explained in the previous exercise). We repeat this process for several number of neighbors and keep the one that obtains the best kappa in the validation data. Once the best parameter is chosen, we test the model by training it again on the concatenation of the training and validation partitions (using the number of neighbors previously estimated as the best) and testing it on the test partition (using again the cross-validation methodology).

"Fig. 3" shows the tuning process of the no. of neigbors for both datasets. It can be seen how the value that works best is **1 in both cases**.



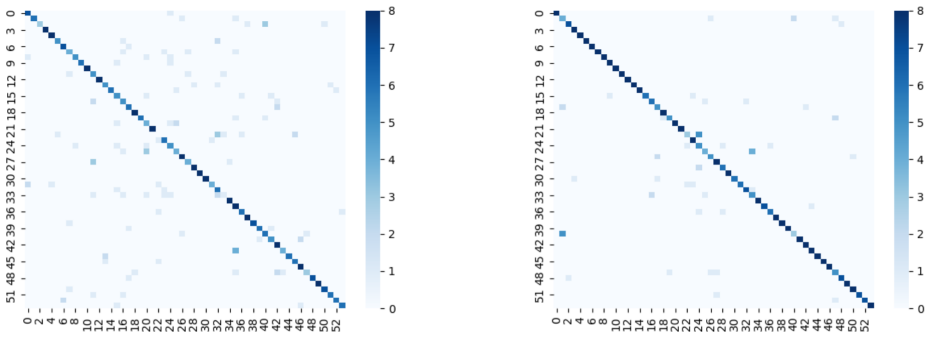
CooCur LBP

Figure 3: No. of neighbors tuning. k-NN, 4-folds cross validation. 1 is the best in both

Finally, "Tab. 3" shows the testing results and "Fig. 4" the confusion matrices.

Table 3: k-NN, 4-folds cross validation. Estimated best K=1

Dataset	**No. features**	**Accuracy (%)**	**Kappa (%)**
Co-occurrence matrices	12	71.88	71.34
Local binary patterns	10	86.57	86.32



CM. CooCur CM. LBP

Figure 4: Confusion matrix for CooCur and LBP. K-NN 4-folds cross validation. Best estimated k=1