# Tutorial 4 - Ioannis Manousaridis

In this classes tutorial the F-MNIST dataset was used and only two overlapping classes, the class number 2 which is for pullovers and the class number 3 which is for dresses. The scope of this project was to use different ensemble learning techniques in the previous described dataset.

| Bagging | | 20% | 40 % | 60 % | 80% | mean |
|---|---|---|---|---|---|---|
| 2 estimators | Accuracy | 0.970929 | 0.972429 | 0.972500 | 0.971786 | 0.971911 |
| | F1 Score | 0.970929 | 0.972429 | 0.972500 | 0.971786 | 0.971911 |
| 3 estimators | Accuracy | 0.971429 | 0.972214 | 0.972786 | 0.971857 | 0.972071 |
| | F1 Score | 0.971428 | 0.972214 | 0.972786 | 0.971857 | 0.972071 |
| 4 estimators | Accuracy | 0.972857 | 0.972214 | 0.972071 | 0.972214 | 0.972339 |
| | F1 Score | 0.972857 | 0.972214 | 0.972071 | 0.972214 | 0.972339 |
| 5 estimators | Accuracy | 0.971357 | 0.972214 | 0.972857 | 0.972071 | 0.972125 |
| | F1 Score | 0.971357 | 0.972214 | 0.972857 | 0.972071 | 0.972125 |

Figure 1: F1 and accuracy scores for classic bagging for different numbers of estimators and different percentages of samples.

| Random Subspaces | | 20% | 40% | 60 % | 80% | mean |
|---|---|---|---|---|---|---|
| 2 estimators | Accuracy | 0.843929 | 0.850143 | 0.568571 | 0.954571 | 0.804304 |
| | F1 Score | 0.843495 | 0.849683 | 0.568571 | 0.954567 | 0.804079 |
| 3 estimators | Accuracy | 0.542500 | 0.537357 | 0.723643 | 0.962643 | 0.691536 |
| | F1 Score | 0.542433 | 0.537264 | 0.722991 | 0.962641 | 0.691332 |
| 4 estimators | Accuracy | 0.961214 | 0.747429 | 0.971571 | 0.963000 | 0.910804 |
| | F1 Score | 0.961210 | 0.746997 | 0.971571 | 0.962996 | 0.910694 |
| 5 estimators | Accuracy | 0.949143 | 0.836000 | 0.970143 | 0.916000 | 0.917821 |
| | F1 Score | 0.949138 | 0.835720 | 0.970142 | 0.915923 | 0.917731 |

Figure 2: F1 and accuracy scores for random subspaces technique for different numbers of estimators and different percentages of samples.

Random Patches (Bagging + Random Subspaces)

| Samples \ Features | | 20% | 40% | 60% | 80% | mean |
|---|---|---|---|---|---|---|
| 20% | Accuracy | 0.966643 | 0.965429 | 0.965357 | 0.967929 | 0.907875 |
| | F1 Score | 0.966640 | 0.965428 | 0.965357 | 0.967928 | 0.907874 |
| 40% | Accuracy | 0.739071 | 0.936929 | 0.960214 | 0.966571 | 0.885768 |
| | F1 Score | 0.738573 | 0.936872 | 0.960211 | 0.966569 | 0.885718 |
| 60% | Accuracy | 0.854571 | 0.922714 | 0.969571 | 0.853357 | 0.909161 |
| | F1 Score | 0.854142 | 0.922641 | 0.969571 | 0.853169 | 0.908995 |
| 80% | Accuracy | 0.961000 | 0.914000 | 0.970571 | 0.938786 | 0.935196 |
| | F1 Score | 0.960994 | 0.913881 | 0.970571 | 0.938727 | 0.935107 |
| 100% | Accuracy | 0.555429 | 0.969857 | 0.870929 | 0.955071 | 0.822250 |
| | F1 Score | 0.555227 | 0.969857 | 0.870613 | 0.955059 | 0.822132 |

Figure 3: F1 and accuracy scores, using random subspaces and classic bagging combined for different numbers of estimators and different percentages of samples.

RandomForest - subsets of samples

| Trees \ Samples | | 20% | 40% | 60% | 80% | mean |
|---|---|---|---|---|---|---|
| 20 | Accuracy | 0.980571 | 0.989857 | 0.995143 | 0.998429 | 0.991000 |
| | F1 Score | 0.980571 | 0.989857 | 0.995143 | 0.998429 | 0.991000 |
| 60 | Accuracy | 0.982929 | 0.991071 | 0.996643 | 0.999500 | 0.992536 |
| | F1 Score | 0.982928 | 0.991071 | 0.996643 | 0.999500 | 0.992536 |
| 100 | Accuracy | 0.982500 | 0.991214 | 0.997500 | 0.999643 | 0.992714 |
| | F1 Score | 0.982500 | 0.991214 | 0.997500 | 0.999643 | 0.992714 |
| 140 | Accuracy | 0.982857 | 0.991214 | 0.997929 | 0.999929 | 0.992982 |
| | F1 Score | 0.982857 | 0.991214 | 0.997929 | 0.999929 | 0.992982 |
| 180 | Accuracy | 0.982929 | 0.991357 | 0.998286 | 1.000000 | 0.993143 |
| | F1 Score | 0.982929 | 0.991357 | 0.998286 | 1.000000 | 0.993143 |

Figure 4: F1 and accuracy scores for different number of trees and different percentages of samples.

RandomForest - subsets of features

| Trees \ Features | | 20% | 40% | 60% | 80% | mean |
|---|---|---|---|---|---|---|
| 20 | Accuracy | 0.99921 | 0.99914 | 0.998714 | 0.998714 | 0.998946 |
| | F1 Score | 0.99921 | 0.99914 | 0.998714 | 0.998714 | 0.998946 |
| 60 | Accuracy | 1 | 1 | 0.999857 | 0.999857 | 0.999929 |
| | F1 Score | 1 | 1 | 0.999857 | 0.999857 | 0.999929 |
| 100 | Accuracy | 1 | 1 | 1 | 1 | 1 |
| | F1 Score | 1 | 1 | 1 | 1 | 1 |
| 140 | Accuracy | 1 | 1 | 1 | 1 | 1 |
| | F1 Score | 1 | 1 | 1 | 1 | 1 |
| 180 | Accuracy | 1 | 1 | 1 | 1 | 1 |
| | F1 Score | 1 | 1 | 1 | 1 | 1 |

Figure 5: F1 and accuracy scores for different number of trees and different percentages of features.

Adaboost

| Trees | 20 | 60 | 100 | 140 | 180 | mean |
|---|---|---|---|---|---|---|
| Accuracy | 0.96771 | 0.9725 | 0.97393 | 0.97614 | 0.9775 | 0.973557 |
| F1 Score | 0.96771 | 0.9725 | 0.97393 | 0.97614 | 0.9775 | 0.973557 |

Figure 6: F1 and accuracy scores for different number of trees for the adaboost method.

## **Conclusions**:

- In all the techniques the F1 score and the Accuracy do not differ a lot. This is most probably to the fact that two classes were used used and binary classification was done instead of multilabel classification.

- The classic bagging method provide excellent results around 97%. The changes in the number of estimators and the samples of data did not improve significantly the results. Thus, the best choice is to use the less information and power as possible, 2 estimators and 20% data.

- In the random subspaces small samples of data gave lower results. More stable results were given with 80% of data. The highest results were for 4 estimators with 60% of data.

- By combining the two previous methods the results were very good but still not higher than the classic bagging method.

- From the figure 4, it can be seen that as the number of trees increases, the performance increases too. The improvement though is not that high in order to justify complexity that is being added by the increasing of the number of the trees.

- From figure 5, it can be conclude that by using 100 trees and more, all the results are 100% despite the percentage of features used.

- Finally, regarding the adaboost method, all the results were around 97% and the increasement of the number of trees, improved the results, but again not enough in order to justify the addition complexity that they add in the program.

- To conclude, the ensemble learning methods provided very good results for the classes 2 and 3 of the FMNIST dataset. It has to be noted though that the different parameters (number of trees, percentage of data samples, percentage of feature) do not require to have very high values. The increment of those parameters make the program more complex without improving the accuracy and the F1-score accordingly.