

# Mobile and Wearable Computing

## Assignment 04

Davide Grandesso

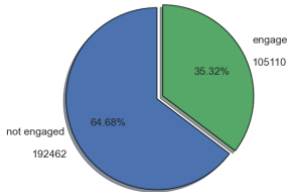
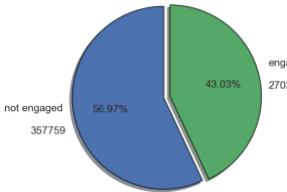
Repository:

[https://github.com/dadegrande99/AssignmentsMWC\\_Grandesso/tree/master/Assignment04](https://github.com/dadegrande99/AssignmentsMWC_Grandesso/tree/master/Assignment04)

### Exercise 0 – Re-run Tutorial Code

Re-run the code from Tutorial 08, but with the data provided for this assignment ("assignment\_data.csv").

Report the new **balanced accuracy** for the 5-fold and Leave One User Out cross validation. Did the results change?

		example_data.csv	assignment_data.csv
Engagement's Distribution			
Samples		297.572	627.973
5-fold	XGBClassifier	68.75 ± 0.78 %	63.40 ± 0.67 %
	DummyClassifier	47.58 ± 0.31 %	50.49 ± 0.24 %
Leave One Group Out	XGBClassifier	45.27 ± 23.23 %	41.92 ± 16.32 %
	DummyClassifier	49.47 ± 1.02 %	49.83 ± 1.21 %

The assignment data are more than twice as large as the example data and are also more balanced, which leads us to think that random approaches may have better results than the example data.

The results between the two cross validation methods remain proportionate to each other.

It is observed that the **Dummy** classifier achieves slight improvements not very significant, this result was to be expected given that the dataset is more balanced.

It is also noted that the **XGBoost** classifier has a worse average balanced accuracy for the new data with however a noticeable improvement in standard deviation in the case of Leave one group out cross validation, these results may depend on the increase in data which may make the model more complex as well as add noise that makes predictions more difficult.

## Exercise 1 – Leave One Day (per user) Out

Now, implement a Leave One Day (per user) Out cross validation. Each user had data for multiple “sessions” (identified by a date).

In the validation paradigms, you’ll have to leave out a single session from a single user.

Write the code to run using the models implemented in class, i.e., XGBoost and the DummyClassifier (with uniform strategy).

Report the balanced accuracy for both models with their standard deviation. Did you obtain better or worse results than in the other validation paradigms? Why do you think it could be the case?

To implement this task the instance we use to subdivide the groups was changed, if before it was created by taking only level 0 of the indices of the working dataset, now it also takes level 1. This means that before we used to do a split by user now by user\_session copying.

Leave one group out:

```
1. groups: np.ndarray = eda_features.index.get_level_values(0).values
```

Leave One Day Out:

```
1. groups_d = eda_features.index.get_level_values(0).astype(str) + '_' +  
eda_features.index.get_level_values(1).astype(str)
```

The table of results is shown below

assignment_data.csv	5-fold	Leave One Group Out	<b>Leave One Day Out</b>
XGBClassifier	63.40 ± 0.67 %	41.92 ± 16.32 %	<b>48.53 ± 16.62 %</b>
DummyClassifier	50.49 ± 0.24 %	49.83 ± 1.21 %	<b>49.68 ± 1.19 %</b>

We still notice how the Dummy classifier reports similar results, instead there are differences regarding the XGB Classifier.

By comparing the mean and standard deviation data with the other two methods it is possible to think that there are instances where we have higher balanced accuracy values; by comparing the last two methods with the 5-Fold cross validation we can also think that these results are due to overfitting.

## Exercise 2 – Statistical test on results

*In class, we have calculated the balanced accuracy, in two validation paradigms (5-fold and Leave One User Out cross validation), for two models.*

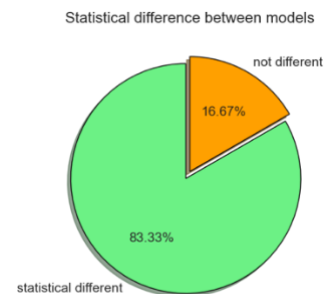
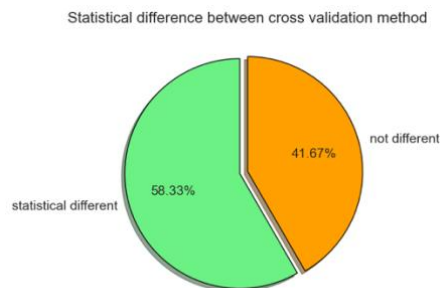
*However, just looking at the results of the balanced accuracy is not enough. Perform a statistical test (t-test) to compare the results obtained from the XGBoost model to those obtained from the DummyClassifier: **are they statistically different?***

*Do this calculation on both 5-fold, Leave One User Out and Leave One Day (per user) Out cross validation.*

*Report the p-value and their statistical significant (with  $\alpha=0.05$ ) for all 3 validation paradigms, and comment on them.*

model	cross	score	p-value	above_threshold
XGBoost	5-Fold vs Leave One Group Out	fit_time	6.144659e-03	True
		score_time	2.844273e-02	True
		test_score	1.355624e-02	True
		train_score	7.680888e-04	True
	5-Fold vs Leave One Day Out	fit_time	4.573978e-04	True
		score_time	1.542416e-09	True
		test_score	6.236852e-02	False
		train_score	2.389205e-08	True
	Leave One Group Out vs Leave One Day Out	fit_time	1.421832e-01	False
		score_time	6.020298e-04	True
		test_score	2.550375e-01	False
		train_score	2.164468e-01	False
Dummy	5-Fold vs Leave One Group Out	fit_time	2.754699e-09	True
		score_time	5.678352e-06	True
		test_score	2.701432e-01	False
		train_score	3.323225e-01	False
	5-Fold vs Leave One Day Out	fit_time	2.679067e-06	True
		score_time	2.634678e-24	True
		test_score	1.531625e-01	False
		train_score	7.790918e-01	False
	Leave One Group Out vs Leave One Day Out	fit_time	2.949902e-02	True
		score_time	7.362071e-06	True
		test_score	7.172678e-01	False
		train_score	6.207283e-02	False

			p-value	above_threshold
cross	model	score		
5-Fold	XGBoost vs Dummy	fit_time	5.061022e-12	True
		score_time	2.887448e-04	True
		test_score	3.643391e-10	True
		train_score	1.066692e-17	True
Leave One Group Out	XGBoost vs Dummy	fit_time	7.289454e-33	True
		score_time	9.176601e-05	True
		test_score	1.068790e-01	False
		train_score	9.287072e-36	True
Leave One Day Out	XGBoost vs Dummy	fit_time	7.141470e-66	True
		score_time	2.010497e-10	True
		test_score	7.267960e-01	False
		train_score	3.681521e-84	True



In this Task, first are compared the differences of each cross validation method with the same model used, then the methods with the same method used are compared with each other.

We can see from the charts, based on the comparisons made, the ratio of statistically similar and non-similar distributions.

The distributions we are most interested in are regarding **test\_score**, in this case we very often find that the distributions of balanced accuracy often cannot be considered statistically different because the value of the p-value is greater than  $\alpha$  (where  $\alpha = 0.05$ )

## Exercise 3 – More models

Implement the 2 validation paradigms shown in class and the one from Exercise 1 for the following models:

- Support Vector Machine
- Random Forest Classifier
- Naïve Bayes Classifier

You can find their implementation in [scikit-learn](https://scikit-learn.org/).

Finally, re-run Exercise 2 on the results from these new models, checking with ones are statistically different from XGBoost and the DummyClassifier. Only compare the new models to these 2: you do not have to perform all possible comparisons.

Report the **balanced accuracy**, the *p-value* for the *t-tests* and their significance for the models mentioned. Comments this results.

assignment_data.csv	5-fold	Leave One Group Out	Leave One Day Out
XGBClassifier	63.40 ± 0.67 %	41.92 ± 16.32 %	48.53 ± 16.62 %
DummyClassifier	50.49 ± 0.24 %	49.83 ± 1.21 %	49.68 ± 1.19 %
SVM	50.05 ± 0.12 %	47.03 ± 36.52 %	51.56 ± 44.47 %
Random Forest	63.60 ± 1.06 %	42.00 ± 16.24 %	48.58 ± 16.69 %
Naïve Bayes	54.81 ± 1.07 %	28.44 ± 22.48 %	36.28 ± 21.99 %

Evaluating the different methods, we can see that generally SVM has a similar average balanced accuracy for the various methods but a very high standard deviation value as the number of groups created increases, in fact this model is known to be sensitive to overfitting so we can think that there are some instance where the model can perform greatly and other instances where it work very bad.

We also notice a similar result for the other two methods except that the standard deviation does not have such a high growth, in fact these two do not suffer as much from overfitting as SVM.

			p-value	above_threshold
cross	model	score		
5-Fold	XGBoost vs SVM	fit_time	2,67E-18	TRUE
		score_time	6,7E-20	TRUE
		test_score	1,97E-10	TRUE
		train_score	1,59E-15	TRUE
	XGBoost vs Random Forest	fit_time	1,59E-16	TRUE
		score_time	9,94E-14	TRUE
		test_score	0,755653	FALSE
		train_score	2,82E-14	TRUE
	XGBoost vs Naive Bayes	fit_time	5,62E-12	TRUE
		score_time	0,000348	TRUE
		test_score	8,33E-07	TRUE
		train_score	9,09E-15	TRUE
	Dummy vs SVM	fit_time	1,81E-18	TRUE
		score_time	5,96E-20	TRUE
		test_score	0,009763	TRUE
		train_score	0,267627	FALSE
	Dummy vs Random Forest	fit_time	1,06E-16	TRUE
		score_time	4,05E-19	TRUE
		test_score	9,04E-09	TRUE
		train_score	1,66E-23	TRUE
	Dummy vs Naive Bayes	fit_time	2,27E-11	TRUE
		score_time	0,001929	TRUE
		test_score	4,99E-05	TRUE
		train_score	1,07E-08	TRUE
Leave One Group Out	XGBoost vs SVM	fit_time	1,83E-20	TRUE
		score_time	8,11E-08	TRUE
		test_score	0,662059	FALSE
		train_score	3,27E-26	TRUE
	XGBoost vs Random Forest	fit_time	1,81E-28	TRUE
		score_time	6,23E-09	TRUE
		test_score	0,990607	FALSE
		train_score	6,36E-28	TRUE
	XGBoost vs Naive Bayes	fit_time	1,04E-32	TRUE
		score_time	9,06E-05	TRUE
		test_score	0,105737	FALSE
		train_score	5,88E-26	TRUE
	Dummy vs SVM	fit_time	1,05E-20	TRUE
		score_time	7,76E-08	TRUE
		test_score	0,792626	FALSE
		train_score	0,016067	TRUE
	Dummy vs Random Forest	fit_time	7,69E-29	TRUE
		score_time	8,97E-10	TRUE
		test_score	0,108652	FALSE

Leave One Day Out	Dummy vs Naive Bayes	train_score	1,02E-48	TRUE
		fit_time	4,51E-29	TRUE
		score_time	0,980156	FALSE
		test_score	0,003072	TRUE
	XGBoost vs SVM	train_score	5,34E-08	TRUE
		fit_time	1,34E-65	TRUE
		score_time	4,16E-24	TRUE
		test_score	0,746317	FALSE
	XGBoost vs Random Forest	train_score	8,43E-77	TRUE
		fit_time	7,32E-84	TRUE
		score_time	1,29E-29	TRUE
		test_score	0,992227	FALSE
	XGBoost vs Naive Bayes	train_score	1,71E-68	TRUE
		fit_time	1,68E-65	TRUE
		score_time	1,03E-09	TRUE
		test_score	0,027555	TRUE
	Dummy vs SVM	train_score	1,11E-62	TRUE
		fit_time	4,19E-66	TRUE
		score_time	3,7E-24	TRUE
		test_score	0,830243	FALSE
	Dummy vs Random Forest	train_score	1,94E-05	TRUE
		fit_time	1,03E-84	TRUE
		score_time	1,06E-31	TRUE
		test_score	0,738207	FALSE
	Dummy vs Naive Bayes	train_score	6,7E-109	TRUE
		fit_time	5,68E-56	TRUE
		score_time	0,23389	FALSE
		test_score	0,003086	TRUE
		train_score	1,71E-25	TRUE

Analyzing the results of the t-test we can see how these 3 models rarely have distributions that are statistically different compared to the XGBoost and Dummy classifiers.