Witold Drzewakowski
XAI, mimuw, 23.11.2023

Homework 6.

In this task I explain the predictions of XGBoost (`xgb`) on dataset `pc1` from OpenML using Permutation-based Variable Importance. I compare it with the ranked importance of variables of the following models:

- a random forest (`forest_50`) with 50 estimators,
- a random forest (`forest_100`) with 100 estimators,
- a decision tree (`tree`)
- a neural network (`nn`)

as well as with `xgb`'s ranked gini coefficients and ranked shap values. Since shap values are local, I average them over the whole validation dataset.

In Fig. 1 we can see that the variables in `pc1` are highly correlated, from which we can expect significant differences in the variable importance. As discussed during the lecture, permuting a highly correlated column leads to examples being out of distribution, and a model can behave randomly there. It is also possible that because of this correlation, a model will learn only to depend on the (fixed or varying) subset of features, so it is possible that adding noise to any feature may not decrease performance. In Fig. 2 I present the balanced accuracy on the validation set of the models that I trained.

In Fig. 3 we can see the aggregated results for subproblems 1, 2 and 3: first five columns contain the Permutation-based Variable Importance ranks of models. The last two columns contain ranked gini coefficients and shap values. Rows represent which feature has noise applied. We can observe several things:

- As anticipated, probably due to high correlation, the variable importance in different models is very different,
    - `I` is the least important feature in `tree` and almost the most in `forest`s and `xgb-shap`.
- `forest_100` and `forest_50` both have the same results for most of the features. It is probably not surprising, because they are ensembles of random trees, and during training, at a certain level the subset of features to consider is randomly sampled, so a large enough forest will not over rely on any specific feature.
- Table 4 represents the Variable Importance before ranking. We can see that some of the values exceed 1, which means that a model can perform better on a dataset with a permuted column than on the dataset with no noise added. This is surprising, but it can happen due to pure luck.
- It is not surprising that also shap and gini produce different ranks, as those are completely different methods. Shap for example is inherently local, so the nature of those explanations will be different.

Fig. 1.

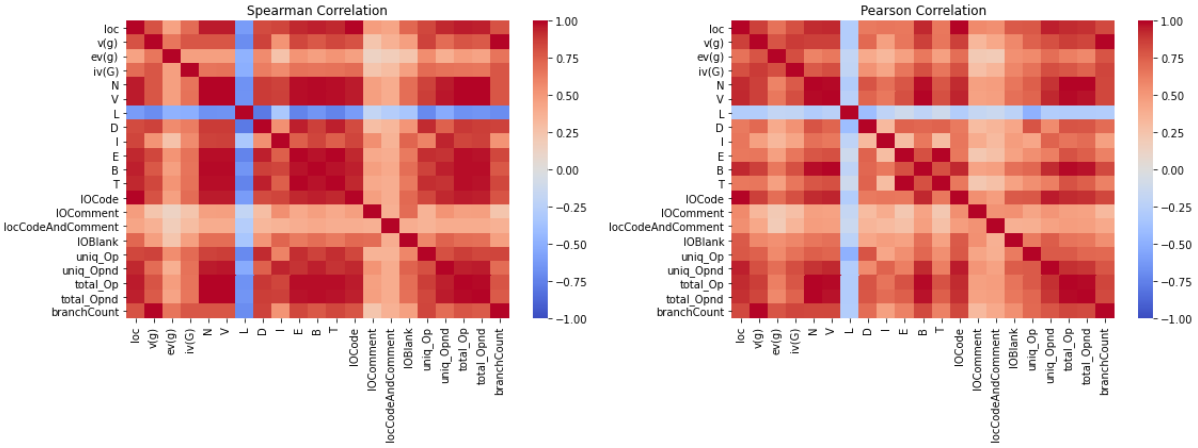
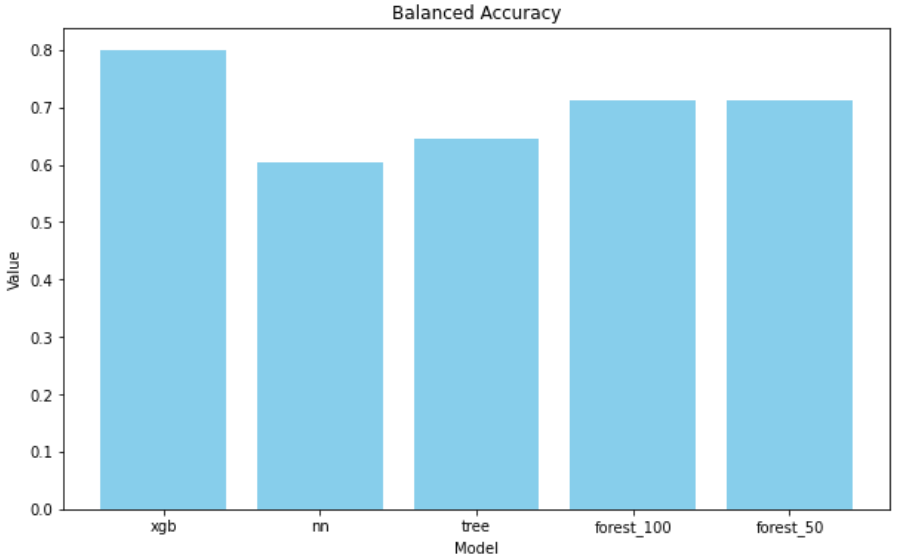

Spearman Correlation
Pearson Correlation
loc
v(g)
ev(g)
iv(G)
N
V
L
D
I
E
B
T
lOCode
lOComment
locCodeAndComment
lOBlank
uniq_Op
uniq_Opnd
total_Op
total_Opnd
branchCount
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Fig. 2


Balanced Accuracy
Value
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
xgb
nn
tree
forest_100
forest_50
Model

.

Fig. 3.

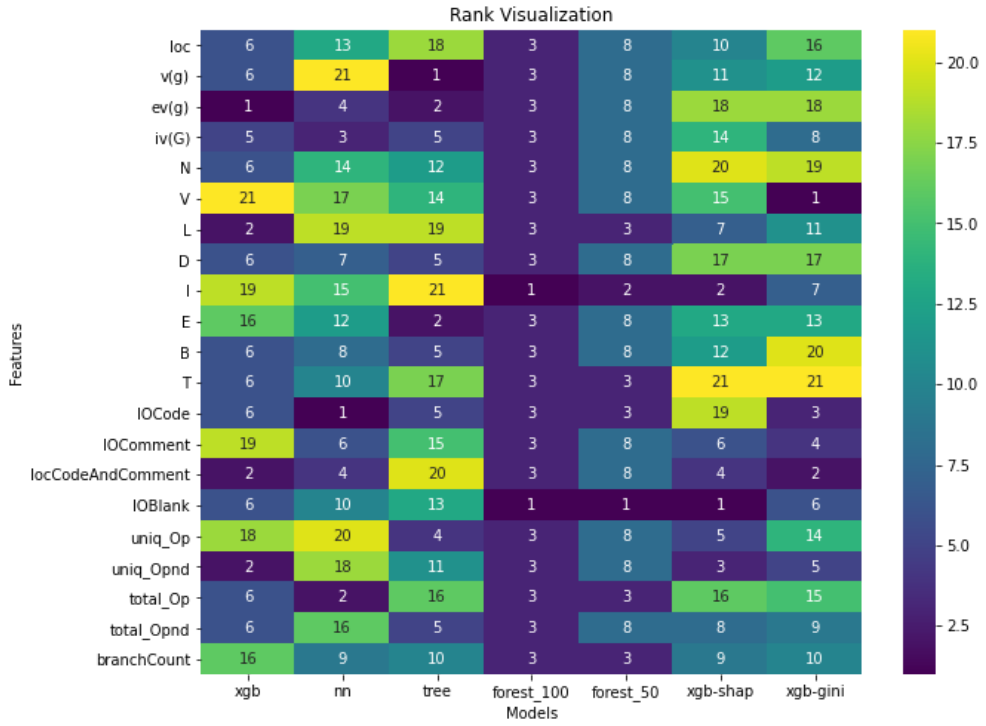


Fig. 4.

	xgb	nn	tree	forest_100	forest_50
loc	1.000000	0.956549	0.912310	1.000000	1.000000
v(g)	1.000000	0.757495	1.082621	1.000000	1.000000
ev(g)	1.083679	1.095318	1.028393	1.000000	1.000000
iv(G)	1.032523	1.101605	1.000000	1.000000	1.000000
N	1.000000	0.942583	0.976294	1.000000	1.000000
V	0.860257	0.877753	0.947377	1.000000	1.000000
L	1.049701	0.850792	0.910748	1.000000	1.073311
D	1.000000	1.045797	1.000000	1.000000	1.000000
I	0.892754	0.925750	0.879376	1.117455	1.117455
E	0.940170	0.970096	1.028393	1.000000	1.000000
B	1.000000	1.024264	1.000000	1.000000	1.000000
T	1.000000	1.000000	0.923915	1.000000	1.073311
lOCode	1.000000	1.264822	1.000000	1.000000	1.073311
lOComment	0.892754	1.071191	0.935479	1.000000	1.000000
locCodeAndComment	1.049701	1.095318	0.883845	1.000000	1.000000
lOBlank	1.000000	1.000000	0.955976	1.117455	1.179032
uniq_Op	0.895246	0.824541	1.019406	1.000000	1.000000
uniq_Opnd	1.049701	0.860885	0.979230	1.000000	1.000000
total_Op	1.000000	1.138713	0.926000	1.000000	1.073311
total_Opnd	1.000000	0.897807	1.000000	1.000000	1.000000
branchCount	0.940170	1.024033	0.982253	1.000000	1.073311