SCALE FOR PROJECT PISCINE PYTHON DATA SCIENCE (/PROJECTS/PISCINE-PYTHON-DATA-SCIENCE) / DAY 09 (/PROJECTS/PISCINE-PYTHON-DATA-SCIENCE-DAY-09)

You should evaluate 1 student in this team

★

Git repository

`git@vogsphere.msk.21-school.ru:vogsphere/intra-uuid-f5ac893`

Introduction

The methodology of School 21 makes sense only if
peer-to-peer assessments are done seriously.
This document will help you to do it properly.

- Please, stay courteous, polite, respectful and constructive in all
communications during this assessment. The bond of trust
between community 21 and you depends on it.

- Highlight possible malfunctions of the work done by the person
and take the time to discuss and debate it.

- Keep in mind that sometimes there can be differences
in interpretation of the tasks and the scope of features.
Please, stay open-minded to the vision of the other.

Guidelines

- Evaluate only the files that are on the GIT
repository of the student or group.

- Doublecheck that the GIT repository is the one
corresponding to the student
or the group as long as to the project.

- Meticulously check that nothing malicious has been used

to mislead you and have you assess something except
the content of the official repository.

- If you have not finished the project yet, it is compulsory
to read the entire instruction before starting the review.

- Use the special flags in the scale to report an empty
or non-functional solution as long as a case of cheating.
In these cases, the assessment is completed
and the final grade is 0 (or in a case of cheating is -42).
However, except for a case of cheating, you are encouraged to continue
reviewing the project to identify the problems that caused the situation
in order to avoid them for the next assessment.

- You must stop giving points from the first wrong exercise
even if the following exercises are correct.

Attachments

subject.pdf (https://cdn.intra.42.fr/pdf/pdf/49859/en.subject.pdf)

attachments.txt (/uploads/document/document/8775/attachments.txt)

Preliminaries

Respect the rules

- The repository contains the work of the student (or group).
- The student is able to explain their work at any time during the assessment.
- The general rules and any rules specific to the day are respected throughout the assessment.

Yes No

Piscine Python | Data Science D09

Any hardcoded result is worth zero for the exercise.

Exercise 00 – Regularization

- Run all the cells in the notebook, they should work without errors
- logreg baseline model: average accuracy on cross-validation is 0.60165, the standard deviation is 0.02943?
- %%time is used to evaluate the time to run the code for fitting the model
- SVM baseline model: average accuracy on cross-validation is 0.65871, the standard deviation is 0.04359?

- Decision tree baseline model: average accuracy on cross-validation is 0.72551, the standard deviation is 0.03562?
- Random forest baseline model: average accuracy on cross-validation is 0.88722, the standard deviation is 0.02204?
- The final model is evaluated on the test dataset?
- The distribution of errors among the classes for the final model is calculated? In all other cases, the test is failed.

Yes No

Exercise 01 – Gridsearch

- Run all the cells in the notebook, they should work without errors
- The best SVM model has the following parameters: {'C': 10, 'class_weight': 'None', 'gamma': 'auto', 'kernel': 'rbf', 'probability': 'true', 'random_state': 21}?
- The best decision tree model has the following parameters: {'class_weight': 'balanced', 'criterion': 'gini', 'max_depth': 21, 'random_state': 21}?
- The best random forest model has the following parameters: {'class_weight': 'balanced', 'criterion': 'entropy', 'max_depth': 24, 'n_estimators': 100, 'random_state': 21}?
- All the resulting datafames are sorted ascendengly by the rank_test_score?
- There is a progress bar with tqdm.notebook (not just ordinary tqdm) for random forest? In all other cases, the test is failed.

Yes No

Exercise 02 – Metrics

- Run all the cells in the notebook, they should work without errors
- The SVM model has the following metrics: accuracy is 0.88757, precision is 0.89267, recall is 0.88757, roc_auc is 0.97878?
- The decision tree has the following metrics: accuracy is 0.88462, precision is 0.88765, recall is 0.88462, roc_auc is 0.93528?
- The random forest has the following metrics: accuracy is 0.92604, precision is 0.92754, recall is 0.92604, roc_auc is 0.98939?
- Use the written function for any of the models – does it create the same results but in a dict? In all other cases, the test is failed.

Yes No

Exercise 03 – Ensembles

- Run all the cells in the notebook, they should work without errors
- The best voting classifier ensemble has the following parameters: {'voting': 'soft', 'weights': [4,1,4]}? where 4 – SVM, 1 – tree, 4 – random forest
- The best bagging classifier ensemble has n_estimators equal to 50?
- The best stacking classifier ensemble has the following parameters: {'passthrough': True} and n_splits=4?

- The best model is the voting classifier with accuracy on the test dataset: 0.90533 and precision: 0.90881? In all other cases, the test is failed.

Yes No

Exercise 04 – Pipelines and OOP

- Run all the cells in the notebook, they should work without errors
- All the classes and methods from the subject exist?
- ModelSelection.choose() returns the info in the exact formatting as in the subject?
- BestVotingClassifier.choose() and BestStackingClassifier.choose() return the info in the exact formatting as in the subject?
- Finalize.final_score() returns the info in the exact formatting as in the subject?
- Finalize.save_model() returns the info that the model was successfully saved?
- Load the saved model and use it in the Finalize() class – does it produce the same score as it was before the model was saved?
- Pass to the MyOneHotEncoder() a categorical column as the target column – does the method perform the transformation to it? In all other cases, the test is failed.

Yes No

Ratings

Don't forget to check the flag corresponding to the defense

Ok Outstanding project

Empty work No author file Invalid compilation Norme Cheat Crash

Leaks Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation