

MACIEJ WIELGOSZ, MARCIN PIETROŃ

PROGRAMOWANIE W JĘZYKU PYTHON

September 2019

This work is licensed under a [Creative Commons Attribution
4.0 International License](https://creativecommons.org/licenses/by/4.0/). 

PROJEKTY ZALICZENIOWE

ZASADY PRACY PROJEKTOWEJ

Należy wykorzystać

- repozytorium git (udostępnione prowadzącemu),
- [PyScaffold](#),
- [VirtualEnv](#)

Finalny projekt

- wyniki pracy (własna implementacja) mają zostać porównane z implementacją dostępną w *scikit-learn*,
- gotowy projekt studencki powinien zawierać pełną dokumentację dotyczącą implementowanego algorytmu oraz testy,
- każdy zespół studencki otrzyma do przetestowania i oceny kod innego zespołu.

NOTATKI

TEMATY PROJEKTÓW

Dane uczące oraz testowe	<ul style="list-style-type: none">• DB scan,• Birch,• SVD,• Random Projection,• SVM,• Learning Vector Quantization (ograniczona dostępność w <i>scikit-learn</i>),• Random Forest,• Cart,• Hierarchical Clustering,• k-NN,• Affinity propagation,• Agglomerative Clustering,• Gaussian mixtures,• K-Means,• Mean-shift,• Nearest Neighbors,• Naive Bayes,• Decision Trees. <p>Przykładowe implementacje algorytmów są dostępne w bibliotece scikit-learn.</p> <ul style="list-style-type: none">• Tekstowe• Obrazy
--------------------------	--

NOTATKI

ETAPY PRACY PROJEKTOWEJ

1. sformowanie zespołu 3 osobowego (sugerowane: 2 programistów, 1 tester oraz administrator repo),
2. wybranie tematu projektu,
3. założenie repo oraz ustalenie strategii rozwoju projektu w jego ramach,
4. analiza wybranego algorytmu,
5. implementacja,
6. testowanie - obejmujące porównanie wyników pracy zaimplementowanego modułu z wersją dostępną w *scikit-learn*,
7. opracowanie dokumentacji projektowej (zarówno algorytmu jak i kodu)

Częściowo dwa ostatnie etapy (testowanie oraz dokumentacja kodu) są dostępne w ramach *PyScaffold*: wymagana jest odpowiednia dokumentacja kodu (komentarze we właściwym formacie) oraz napisanie testów.

NOTATKI