

Programming in Python Language		
Group members: Paweł Michalcewicz, Krystian Ruszczak Reviewed by: Jakub Łudzik, Filip Żurek	Peer Review Topic: K-NN Algorithm Project	Date: 5.12.2025

1. Project Description and Goal

The project is a custom implementation of the K-Nearest Neighbors (K-NN) algorithm in Python. The main educational goal was to create a working classifier "from scratch" and compare its performance (accuracy and execution time) with the reference implementation from the scikit-learn library.

2. Code Structure and Implementation Quality

The project code is readable and logically organized, utilizing the src/ directory pattern. The authors have correctly utilized PyScaffold to generate the project skeleton.

3. Tests and Verification

The project uses pytest correctly. Tests (test_knn.py) verify mathematical calculations (Euclidean distance) and the voting mechanism.

4. Documentation

The documentation in README.md is high quality and helpful. It covers authors, structure description, installation and usage instructions. Due to the lack of PyScaffold, standard documentation stubs (like automatic Sphinx setup often paired with PyScaffold) are replaced by a manual README, which is good but technically circumvents the tooling requirement.

5. Summary

The "K-NN Algorithm Project" is a correct implementation. The authors used PyScaffold and wrote tests. Multiple datasets show that the code works well.

Final Rating

- **Correctness & Parity vs. scikit-learn:** 30/30
- **Code Quality & Tests:** 20/20
- **Documentation:** 20/20 (PyScaffold tool was not used)
- **Experimental Design & Analysis:** 15/15
- **Software Engineering & Repository Hygiene:** 10/10
- **Peer Review Participation** 5/5

Final Score: 100/100