Лабораторная работа №1. Метрический классификатор

Задание. Реализовать метрический классификатор типа k-ближайших соседей – метод парзеновского окна переменной ширины.
Вход: Dataset (размеченная обучающая выборка с категориальными ответами (метки классов))
Выход: k (оптимальное значение); LOO(k) (оценка обобщающей способности); accuracy (доля правильно классифицированных объектов)
Дополнительные условия.
1. Библиотеки: numpy, pandas, matplotlib, seaborn, sklearn (только для получения датасета) и т.д.
1. Dataset: Iris. Можно выбрать другой датасет из репозитория UCI, но с небольшим количеством количественных признаков.
3. Метрика расстояния: евклидовое.
5. Метрика качества классификации: accuracy – доля правильно классифицированных объектов.
6. Функция ядра K(r): ядро Епаничникова:

$$K_1(r) = E(r) = \frac{3}{4}(1-r^2)[|r|<=1],$$

где [ ] - нотация Айверсона: [ложь] =0, [истина]=1.
7. Скользящий контроль для настройки k: Leave-one-out (LOO) - количество ошибок на контроле.

Требования к реализации.
1. Загрузка стандартного датасета:

```
from sklearn.datasets import load_iris
iris = load_iris()
X = iris.data  # массив numpy
Y = iris.target # массив numpy
```

2. Модульная структура программы (набор самостоятельных функций). Рекомендации: функция расстояния (Distance), функция ядра (Kernel), определение класса объекта (Predict), скользящий контроль (LOO). Можно реализовать в виде класса NearestNeighbor с соответствующими методами (ООП вариант приветствуется).
3. Построить график LOO(k).
4. Комментарии к коду.

Материалы.
1. Самоучитель Python: https://pythontutor.ru/
2. Официальный учебник Python на английском: https://docs.python.org/3/tutorial/

- на русском:
https://ru.wikibooks.org/wiki/Python/%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D0%B8%D0%BA_Python_3.1
3. Шпаргалки Python-DataScience:
https://www.dropbox.com/sh/gmfsu39jqsagyq9/AADD2w4M3eUF2s1jn_Fk4AMXa?dl=0
4. Мануал по библиотекам Data science: https://scipy.org/
5. Сто заданий по Numpy (чем больше сделайте, тем лучше для вас):
https://github.com/rougier/numpy-100
6. Лекция Воронцова К.В. «Курс Машинное обучение» 2019:
https://www.youtube.com/watch?v=SZkrxWhI5qM&list=PLJOzdkh8T5krxc4HsHbB8g8f0hu7973fK – Машинное обучение. Метрические методы. К.В. Воронцов, Школа анализа данных, Яндекс
Установка Python. Установить сборку Anaconda со всеми необходимыми библиотеками (numpy, pandas, matplotlib, sklearn и т.д. + редакторы кода):
https://www.anaconda.com/distribution/