Практическая работа №7: «Метод *k*-ближайших соседей. Метод *k*-средних».

Оглавление

Цель работы

Получить практические навыки использования метода *k*-ближайших соседей для решения задачи регрессии.

Задачи работы

1. Научиться решать задачу регрессии с помощью библиотеки sklearn, используя метод *k*-ближайших соседей.
2. Научиться подбирать оптимальное значение параметра *k* с помощью среднеквадратичной ошибки.

Перечень обеспечивающих средств

1. ПК.
2. Учебно-методическая литература.
3. Задания для самостоятельного выполнения.

Общие теоретические сведения

Метод *k*-ближайших соседей, классификация

Данные: элементы (x_i, y_i),

где $y_i \in \{Y_1, \ldots, Y_M\}$,

$i = 1, \ldots, N$ (размер набора данных), M – количество классов.

Задача: Найти такую функцию $f(x)$, чтобы $f(x_i) \approx y_i$ для всех *i*.

Общий алгоритм определения класса элемента, не входящего в обучающую выборку:

- Вычислить расстояние до каждого из элементов обучающей выборки.
- Отобрать *k* элементов, расстояние до которых минимально.
- Класс элемента — это класс, чаще всего встречающийся среди отобранных элементов.

Расстояние:

Некая функция $\rho(x_i, x_j)$, которая удовлетворяет гипотезе компактности.

Гипотеза компактности:

Более близкие объекты чаще относятся к одному и тому же классу, чем к разным.

Для любого x элементы обучающей выборки можно упорядочить по увеличению расстояния:

$$\rho(x, \hat{x}_1) \leq \rho(x, \hat{x}_2) \leq \cdots \leq \rho(x, \hat{x}_N)$$

$$f(x) = \arg\max_{j=1,\ldots,M} \sum_{i=1}^{k} I(\hat{y}_i, Y_j)$$

где $I(a, b) = \begin{cases} 0, \text{если } a \neq b \\ 1, \text{если } a = b \end{cases}$

Метод k-ближайших соседей, регрессия

Данные: элементы (x_i, y_i),

где $y_i \in \mathbb{R}$,

$i = 1, \ldots, N$ (размер набора данных).

Задача: Найти такую функцию $f(x)$, чтобы $f(x_i) \approx y_i$ для всех i.

Общий алгоритм определения числового значения для элемента, не входящего в обучающую выборку:

- Вычислить расстояние до каждого из элементов обучающей выборки.
- Отобрать k элементов, расстояние до которых минимально.
- Значение для элемента — это среднее арифметическое значений для отобранных элементов.

Для любого x элементы обучающей выборки можно упорядочить по увеличению расстояния:

$$\rho(x, \hat{x}_1) \leq \rho(x, \hat{x}_2) \leq \cdots \leq \rho(x, \hat{x}_N)$$

$$f(x) = \frac{1}{k} \sum_{i=1}^{k} \hat{y}_i$$

Задание

Пояснение

Для сохранения результатов данной работы вам понадобится два файла: doc/docx – для текста и ipynb – для кода. Назовите их одинаково: «*Фамилия* – задание 7».

Часть 1

- Обновите свой репозиторий, созданный в практической работе №1, из оригинального репозитория: https://github.com/mosalov/Notebook_For_AI_Main.

Часть 2

- Откройте свой репозиторий в Binder (https://mybinder.org/).
- Откройте файл «task7.ipynb».
- Изучите, при необходимости – выполните повторно, приведённый в файле код.
- По аналогии с изученным выполните задание, приведённое в последней ячейке.
- Сохраните код в ipynb-файле. Необходимые пояснения опишите в своём docx/doc-файле.

Требования к отчету

Оба файла (doc/docs и ipynb) загрузите в свой репозиторий, созданный в практическом задании №1 по пути: «Notebook_For_AI_Main/2020 Осенний семестр/Практическое задание 7/» и сделайте пул-реквест.

Литература

- www.machinelearning.ru/wiki/index.php?title=Метод_ближайшего_соседа
- https://learnmachinelearning.wikia.org/ru/wiki/Метод_ближайших_соседей_(kNN)
- https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html