

Задание 2. Основы numpy + matplotlib + pandas

Курс по методам машинного обучения, 2022-2023, Крыжановская Светлана

1 Характеристики задания

- **Длительность:** 2 недели
- **Кросс-проверка:** 13 баллов; в течение 1 недели после дедлайна; нельзя сдавать после жесткого дедлайна
- **Юнит-тестирование:** 27 баллов; можно сдавать после дедлайна со штрафом в 40%; Публичная часть
- **Почта:** ml.cmc@mail.ru
- **Темы для писем на почту:** ВМК.ML[Задание 2][peer-review], ВМК.ML[Задание 2][unit-tests]

Кросс-проверка: После окончания срока сдачи, у вас будет еще неделя на проверку решений как минимум **3х других студентов** — это **необходимое** условие для получения оценки за вашу работу. Если вы считаете, что вас оценили неправильно или есть какие-то вопросы, можете писать на почту с соответствующей темой письма

2 Описание задания

Задание состоит из трех частей, посвященных работе с табличными данными с помощью библиотеки **pandas**, визуализации с помощью библиотек **matplotlib**, **seaborn**, **plotly** и векторным вычислениям с помощью библиотеки **numpy**. В каждой части Вам необходимо выполнить несколько заданий. По **numpy** и визуализации есть отдельные tutorиалы, ссылки на которые вы найдете ниже, в которых можно найти информацию по библиотекам и попрактиковаться в их применении.

Задание состоит из двух различных видов подзаданий: **кросс-рецензирование** и **юнит-тестирование**.

3 Кросс-проверка (13 баллов)

- **Тьюториал по numpy:** [ссылка тут](#)
- **Тьюториал по визуализации:** [ссылка тут](#)
- **Ссылка на задание (ноутбук, который вы будете отправлять на кросс-проверку):** [ссылка тут](#)

Внимание! Отправлять на кросс-рецензирование в систему нужно **ТОЛЬКО** заполненный ноутбук `numpy-pandas-matplotlib-2022.ipynb`. Тьюториалы отправлять никуда не нужно!

Внимание! Отправлять задание нужно в систему во вкладку `numpy-pandas-matplotlib (notebook)`.

Внимание! Отправлять задание нужно только с расширением `ipynb`! После отправки проверьте корректность загруженного задания в систему, просмотрев глазами загруженное решение (оно автоматически конвертируется в `html`). Как это сделать, можно найти в [тьюториале тут](#)

3.1 pandas

В ноутбуке `numpy-pandas-matplotlib.ipynb` необходимо ответить на 10 вопросов по анализу табличных данных с помощью библиотеки **pandas**. Многие из заданий можно выполнить несколькими способами. Не существуют единственно верного, но для решения так или иначе должен быть задействован арсенал **pandas**. Все задания в этой части оцениваются по системе **кросс-рецензирования**.

3.2 matplotlib

Перед выполнением этой части советуем вам заглянуть в [тutorial по визуализации](#).

В ноутбуке `numpy-pandas-matplotlib.ipynb` необходимо построить 3 визуализации по табличным данным. При желании в решении заданий допустимо пользоваться любыми средствами для визуализации в ноутбуке — **главное, проверьте, что при конвертации ноутбука в html, а также при открытии сданного ноутбука из проверяющей системы, все графики по-прежнему видны**. Все графики будут оцениваться по системе **кросс-рецензирования** на содержательность и соответствие правилам, описанным в ноутбуке.

3.3 numpy

Перед выполнением этой части советуем вам заглянуть в [тutorial по numpy](#).

В файлах `functions.py` и `functions_vectorised.py` находятся шаблоны шести функций, которые необходимо реализовать в рамках задания. Формулировки заданий прописаны в ноутбуке `numpy-pandas-matplotlib.ipynb`. Библиотеками, не объявленными в импорте в файлах с шаблонами функций, пользоваться запрещено. Модули с реализованными функциями необходимо **сдать в систему для автоматической проверки**. Все тесты находятся в открытом доступе и предварительное тестирование может быть запущено локально на компьютере.

Помимо реализации функций, необходимо провести сравнение скорости работы функций в ноутбуке `numpy-pandas-matplotlib.ipynb`. Графики и выводы будут оцениваться по системе **кросс-рецензирования**.

4 Юнит-тестирование (27 баллов)

Уже знакомый вам формат, в котором необходимо реализовать какие-либо функции. В данном задании вам необходимо реализовать функции, находящиеся в файлах `functions.py` и `functions_vectorised.py`. После реализации ваш код можно протестировать локально, а затем его необходимо сдать в проверяющую систему (вкладка `numpy-pandas-matplotlib (unit-tests)`).

Замечание: Запрещается пользоваться библиотеками, импорт которых не объявлен в файле с шаблонами функций.

Замечание: Задания, в которых есть решения, содержащие в каком-либо виде взлом тестов, дополнительные импорты и прочие нечестные приемы, будут автоматически оценены в 0 баллов без права передачи задания.

5 Тестирование

Для локального тестирования необходимо скачать из тестирующей системы папку `public_tests` с публичными тестами и скрипт для тестирования `run.py` и положить их в одну директорию вместе с файлами `functions.py` и `functions_vectorised.py` с реализованными функциями. Для тестирования необходимо запустить команду

```
$ python run.py public_tests.
```

Каждая функция тестируется на 4-6 тестах на правильность, а функции из модуля `functions_vectorised.py` дополнительно тестируются на время выполнения. Входные тестовые данные для функций лежат в папках `[0-9]_task[1-6]_[vectorised/unvectorised]_input`, а правильные решения в папках `[0-9]_task[1-6]_[vectorised/unvectorised]_gt`. Входные тестовые данные для функций хранятся в NumPY файлах, а правильные результаты в формате `.pkl`. Примеры чтения входных данных и правильных ответов:

```
1 import numpy as np
2 X = np.load('public_tests/07_task1_vectorised_input/input_0/X.npy')
```

```
1 import pickle
2 with open('public_tests/07_task1_vectorised_gt/output_0.pkl', 'rb') as f:
3     data = pickle.load(f)
```