ВМК МГУ

Задание 2. Основы numpy + matplotlib + pandas
Base & Research

Курс по методам машинного обучения, 2024-2025, Крыжановская Светлана

1 Характеристики задания

- **Длительность:** 9 дней
- **Base**
  - **Кросс-проверка:** 7 баллов + 0.5 бонусных балла; в течение 1 недели после дедлайна; нельзя сдавать после жесткого дедлайна
  - **Юнит-тестирование:** 3 балла; можно сдавать после дедлайна со штрафом в 40%; Публичная часть
- **Research**
  - **Кросс-проверка:** 5 баллов; в течение 1 недели после дедлайна; нельзя сдавать после жесткого дедлайна
  - **Юнит-тестирование:** 5 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

Это общее описание задания как для практического задания уровня base, так и для практического задания уровня research. Оба уровня устроены практически идентично с точки зрения заданий, поэтому все дальнейшее применимо для всех уровней 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 туториалы, ссылки на которые вы найдете в системе проверки, в которых можно найти информацию по библиотекам и попрактиковаться в их применении.

3 Кросс-проверка

Внимание! Отправлять на кросс-рецензирование в систему нужно **ТОЛЬКО** приложенные заполненные ноутбуки. Туториалы отправлять никуда не нужно!

Внимание! Отправлять задание нужно в систему во вкладку с пометкой (notebook).

Внимание! Отправлять задание нужно только с расширением ipynb! После отправки проверьте корректность загруженного задания в систему, просмотрев глазами загруженное решение (оно автоматически сконвертируется в html). Как это сделать, можно найти в туториале по проверяющей системе на сайте курса.

Внимание!: Перед сдачей проверьте, пожалуйста, что не оставили в ноутбуке где-либо свои ФИО, группу и так далее — кросс-рецензирование проводится анонимно.

3.1 pandas

В приложенном к заданию ноутбуке необходимо ответить на несколько 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 туториал по визуализации, приложенный в системе сдачи задания.

В ноутбуке необходимо построить несколько визуализаций 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 туториал по numpy, приложенный в системе проверки.

В файлах base_functions.py (и research_functions.py) и base_functions_vectorised.py (и research_functions_vectorised.py) находятся шаблоны нескольких функций, которые необходимо реализовать в рамках задания. Формулировки заданий прописаны в прилагаемых ноутбуках. 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 research-уровня. Графики и выводы будут оцениваться по системе **кросс-рецензирования**.

4 Юнит-тестирование

Уже знакомый вам формат, в котором необходимо реализовать какие-либо функции. В данном задании вам необходимо реализовать функции, находящиеся в файлах base_functions.py (и research_functions.py) и base_functions_vectorised.py (и research_functions_vectorised.py). После реализации ваш код можно протестировать локально, а затем его необходимо сдать в проверяющую систему во вкладку с пометкой (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 пересдачи задания.

5 Тестирование

В cv-gml можно скачать все файлы, необходимые для тестирования, одним архивом. Для этого просто скачайте zip-архив во вкладке **шаблон решения** соответствующего задания и разархивируйте его. Для тестирования необходимо запустить команду

$ python run.py public_tests

Каждая функция тестируется на 4-6 тестах на правильность, а функции из модулей *_vectorised.py дополнительно тестируются на время выполнения. Входные тестовые данные для функций лежат в папках с окончанием _input, а правильные решения в папках с окончанием _gt. Входные тестовые данные для функций

хранятся в NumPY файлах, а правильные результаты в формате .pkl. Примеры чтения входных данных и правильных ответов:

```
import numpy as np
 X = np.load('public_tests/01_test_task1v_input/input_0/X.npy')
```

```
import pickle
with open('public_tests/01_test_task1v_gt/output_0.pkl', 'rb') as f:
        data = pickle.load(f)
```