ВМК МГУ

Задание 3. Sklearn-scaling-cv

Курс по методам машинного обучения, 2024-2025, Драгунов Никита

1 Характеристики задания

- **Длительность:** 1 неделя
- **Base**
  - **Кросс-проверка:** 6 баллов; в течение 1 недели после дедлайна; нельзя сдавать после жесткого дедлайна
  - **Юнит-тестирование:** 4 балла; можно сдавать после дедлайна со штрафом в 40%; Публичная часть
- **Почта:** ml.cmc@mail.ru
- **Темы для писем на почту:** ВМК.ML[Задание 3][peer-review], ВМК.ML[Задание 3][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 вопросы, можете писать на почту с соответствующей темой письма

2 Описание задания

В настоящем задании вы познакомитесь с библиотекой sklearn, основными способами ее использования на примере алгоритма машинного обучения kNN. Также вы подробнее изучите, что такое и зачем нужна нормализация (скейлинг) данных, а также можно проверять качество работы модели и выбирать наилучшую с помощью метода кросс-валидации.

3 Кросс-проверка

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.

4 Юнит-тестирование

- В ходе выполнения задания из jupyter-notebook вам необходимо реализовать два класса в модуле `scalers.py`. Первый вид нормализации StandardScaler выполняет нормализацию к распределению с нулевым матожиданием и единичной дисперсией. Второй вид нормализации MinMaxScaler отображает признаковые описания в отрезок [0, 1].

Файл `scalers.py` можно найти в архиве из **шаблона решения** из соответствующего задания в системе проверки. Более подробное описание входных и выходных данных вы найдете в этих файлах. После реализации ваш код можно протестировать локально (описано в конце pdf), а затем его необходимо сдать в проверяющую систему.

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 Стиль программирования

Внимание! Обновление!!!

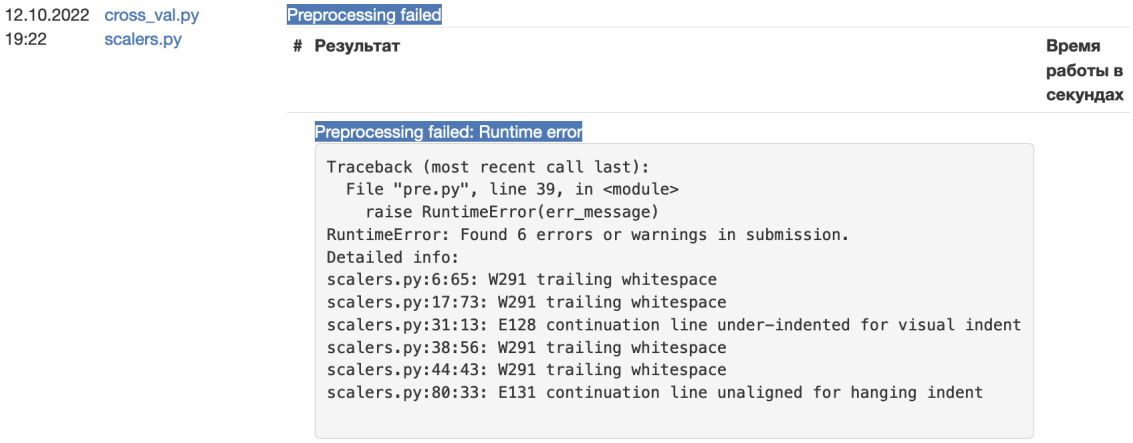
Начиная с этого задания при выполнении задач типа unit-tests, ML-задания вам необходимо будет соблюдать определенный стиль программирования (codestyle). В данном случае мы выбирали PEP8 как один из популярных стилей для языка Python. Зачем мы это вводим? Хорошая читаемость кода – не менее важный параметр, чем работоспособность кода :) Единый стиль позволяет быстрее понимать код сокомандников (в командных проектах, например), упрощает понимание кода (как другим, так и вам). Также, привыкнув к какому-либо стилю программирования, вам будет проще переориентироваться на другой.

Полезные при изучении PEP8 ссылки, если что-то непонятно, дополнительный материал можно найти самостоятельно в интернете:

- Официальный сайт PEP8, на английском
- Небольшое руководство по основам на русском

Требования к PEP8 мы вводим только для заданий с авто-тестами, требований к такому же оформлению ноутбуков нет. Но улучшение качества кода в соответствии с PEP8 в них приветствуется!

Внимание!!! В проверяющей системе, при несоответствии прикрепляемого кода PEP8, будет высвечиваться вердикт `Preprocessing failed`. Более подробно посмотреть на ошибки можно, нажав на них:



Также посылки, упавшие по code style, считаются за попытку сдачи и идут в счет общего количества посылок за день.

Проверить стиль программирования локально можно при помощи утилиты pycodestyle (в окружении, которое вы ставили, эта утилита уже есть) с параметром максимальной длины строки (мы используем 160 вместо дефолтных 79):

```
pycodestyle --max-line-length=160 your_file_with_functions.py
```

6 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. Далее следуйте инструкциям по запуску тестирования.

Тесты запускаются с помощью команд:

```
python run.py unittest scalers
```