

УДК 004.04Python

ББК 32.372

Г90

Груздев А. В.

Г90 Предварительная подготовка данных в Python: Том 1. Инструменты и валидация. – М.: ДМК Пресс, 2023. – 816 с.: ил.

ISBN 978-5-93700-156-6

В двухтомнике представлены материалы по применению классических методов машинного обучения в различных промышленных задачах. Первый том посвящен инструментам Python – основным библиотекам, классам и функциям, необходимым для предварительной подготовки данных, построения моделей машинного обучения, выполнения различных стратегий валидации. В конце первого тома разбираются задачи с собеседований по SQL, Python, математической статистике и теории вероятностей.

Издание рассчитано на специалистов по анализу данных, а также может быть полезно широкому кругу специалистов, интересующихся машинным обучением.

УДК 004.04Python

ББК 32.372

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но, поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

Оглавление

Введение	10
ЧАСТЬ 1. НЕМНОГО МАТЕМАТИКИ	11
1.1. Функция	11
1.2. Производная	12
1.3. Дифференцирование сложных функций	15
1.4. Частная производная	16
1.5. Градиент	17
1.6. Функция потерь и градиентный спуск	18
Часть 2. Инструменты	23
1. Введение	23
1.1. Структуры данных	23
1.1.1. Кортеж (tuple)	23
1.1.2. Список (list)	24
1.1.3. Словарь (dictionary)	27
1.1.4. Множество (set)	31
1.2. Функция	34
1.3. Полезные встроенные функции	35
1.3.1. Функция enumerate()	35
1.3.2. Функция sorted()	36
1.3.3. Функция zip()	36
1.4. Класс	38
1.5. Знакомство с Anaconda	43
2. IPython и Jupyter Notebook	44
3. NumPy	50
3.1. Создание массивов NumPy	50
3.2. Обращение к элементам массива	55
3.3. Получение краткой информации о массиве	57
3.4. Изменение формы массива	58
3.5. Конкатенация массивов	61
3.6. Функции математических операций, знакомство с правилами транслирования	65
3.7. Обработка пропусков	70
3.8. Функция np.linspace()	72
3.9. Функция np.logspace()	74

3.10. Функция np.digitize()	75
3.11. Функция np.searchsorted()	76
3.12. Функция np.bincount()	78
3.13. Функция np.apply_along_axis()	79
3.14. Функция np.insert()	80
3.15. Функция np.repeat()	81
3.16. Функция np.unique()	82
3.17. Функция np.take_along_axis()	84
3.18. Функция np.array_split()	86
4. Библиотеки Numba, datatable, bottleneck для ускорения вычислений	88
4.1. Numba	88
4.2. Datatable	94
4.3. Bottleneck	98
5. SciPy	99
6. pandas	111
6.1. Почему pandas?	111
6.2. Библиотека pandas построена на NumPy	111
6.3. pandas работает с табличными данными	111
6.4. Объекты DataFrame и Series	111
6.5. Задачи, выполняемые pandas	113
6.6. Кратко о типах данных	113
6.7. Представление пропусков	114
6.8. Какую версию pandas использовать?	115
6.9. Подробно знакомимся с типами данных	115
6.9.1. Типы данных для работы с числами и логическими значениями	115
6.9.2. Типы данных для работы со строками	126
6.10. Чтение данных	136
6.11. Получение общей информации о датафрейме	137
6.12. Изменение настроек вывода с помощью функции get_options()	139
6.13. Знакомство с индексаторами [], loc и iloc	140
6.14. Фильтрация данных	147
6.14.1. Одно условие	147
6.14.2. Несколько условий	148
6.14.3. Несколько условий в одном столбце	148
6.14.4. Использование метода .query()	149
6.15. Агрегирование данных	151
6.15.1. Группировка и агрегирование с помощью одного столбца	151
6.15.2. Группировка и агрегирование с помощью нескольких столбцов	153
6.15.3. Группировка с помощью сводных таблиц	156
6.16. Анализ частот с помощью таблиц сопряженности	166
6.17. Выполнение SQL-запросов в pandas	169

7. scikit-learn	179
7.1. Основы работы с классами, строящими модели предварительной подготовки данных и модели машинного обучения	179
7.2. Строим свой первый конвейер моделей	198
7.3. Разбираемся с дилеммой смещения–дисперсии и знакомимся с бутстрепом	210
7.4. Обработка пропусков с помощью классов MissingIndicator и SimpleImputer	228
7.5. Выполнение дамми-кодирования с помощью класса OneHotEncoder и функции get_dummies(), знакомство с разреженными матрицами	235
7.6. Автоматическое построение конвейеров моделей с помощью класса Pipeline	246
7.7. Знакомство с классом ColumnTransformer	250
7.8. Класс FeatureUnion	263
7.9. Выполнение перекрестной проверки с помощью функции cross_val_score(), получение прогнозов перекрестной проверки с помощью функции cross_val_predict(), сохранение моделей перекрестной проверки с помощью функции cross_validate()	264
7.10. Виды перекрестной проверки для данных формата «один объект – одно наблюдение» (отсутствует ось времени)	273
7.10.1. Обычная нестратифицированная k -блочная перекрестная проверка с помощью класса KFold	274
7.10.2. Обычная стратифицированная k -блочная перекрестная проверка с помощью класса StratifiedKFold	281
7.10.3. Повторная нестратифицированная k -блочная перекрестная проверка с помощью класса RepeatedKFold	283
7.10.4. Повторная стратифицированная k -блочная перекрестная проверка с помощью класса RepeatedStratifiedKFold	286
7.10.5. k -кратное случайное разбиение на обучающую и тестовую выборки (перекрестная проверка Монте-Карло)	288
7.10.6. Перекрестная проверка со случайными перестановками при разбиении с помощью класса ShuffleSplit	294
7.10.7. Стратифицированная перекрестная проверка со случайными перестановками при разбиении с помощью класса StratifiedShuffleSplit	296
7.10.8. Перекрестная проверка с исключением по одному с помощью класса LeaveOneOut	297
7.10.9. Перекрестная проверка с исключением p наблюдений с помощью класса LeavePOut	299
7.11. Виды перекрестной проверки для данных формата «один объект – несколько наблюдений» и стратифицированных данных (отсутствует ось времени)	301
7.11.1. Перекрестная проверка, учитывающая группы связанных наблюдений, с помощью классов GroupKFold	301
7.11.2. Перекрестная проверка, учитывающая группы связанных наблюдений с исключением из обучения одной группы, с помощью класса LeaveOneGroupOut	302

7.11.3. Перекрестная проверка, учитывающая группы связанных наблюдений с исключением из обучения p групп, с помощью класса <code>LeavePGroupsOut</code>	304
7.11.4. Перекрестная проверка, учитывающая группы связанных наблюдений и распределение классов, с помощью класса <code>StratifiedGroupKFold</code>	305
7.11.5. Перекрестная проверка со случайными перестановками при разбиении и учитывающая группы связанных наблюдений с помощью класса <code>GroupShuffleSplit</code>	307
7.12. Обычный и случайный поиск наилучших гиперпараметров по сетке с помощью классов <code>GridSearchCV</code> и <code>RandomizedSearchCV</code>	309
7.12.1. Обычный поиск оптимальных значений гиперпараметров моделей предварительной подготовки и модели машинного обучения	312
7.12.2. Обычный поиск оптимальных значений гиперпараметров моделей предварительной подготовки и модели машинного обучения с добавлением строки прогресса	318
7.12.3. Случайный поиск оптимальных значений гиперпараметров моделей предварительной подготовки и модели машинного обучения	320
7.12.4. Обычный поиск оптимальных значений гиперпараметров для <code>CatBoost</code> при обработке категориальных признаков «как есть» (заданы индексы категориальных признаков)	321
7.12.5. Отбор оптимальной модели предварительной подготовки данных в рамках отдельного трансформера	324
7.12.6. Отбор оптимального метода машинного обучения среди разных методов машинного обучения (перебор значений гиперпараметров с отдельной предобработкой данных под каждый метод машинного обучения)	329
7.13. Вложенная перекрестная проверка	335
7.14. Классы <code>PowerTransformer</code> , <code>KBinsDiscretizer</code> и <code>FunctionTransformer</code>	341
7.15. Написание собственных классов предварительной подготовки для применения в конвейере	350
7.16. Модификация классов библиотеки <code>scikit-learn</code> для работы с датафреймами	375
7.17. Полный цикл построения конвейера моделей в <code>scikit-learn</code>	381
7.17.1. Первая задача	381
7.17.2. Вторая задача	393
7.18. Калибровка модели	404
7.18.1. Актуальность калибровки	404
7.18.2. Функция <code>calibration_curve()</code>	406
7.18.3. Оценка Брайера	413
7.18.4. Оценка качества калибровки моделей до применения калибратора	415
7.18.5. Класс <code>CalibratedClassifierCV</code>	420
7.18.6. Оценка качества калибровки моделей после применения калибратора	421

7.18.7. Оценка качества калибровки моделей после применения калибратора с уже обученным классификатором.....	423
7.18.8. Калибровка на основе сплайнов.....	426
7.19. Полезные классы CountVectorizer и TfidfVectorizer для работы с текстом.....	436
7.20. Сравнение моделей, полученных в ходе поиска по сетке, с помощью статистических тестов.....	450
7.20.1. Простое сравнение всех построенных моделей.....	451
7.20.2. Сравнение двух моделей: частотный подход.....	454
7.20.3. Сравнение двух моделей: байесовский подход.....	458
7.20.4. Парное сравнение всех моделей: частотный подход.....	463
7.20.5. Парное сравнение всех моделей: байесовский подход.....	465
7.20.6. Итоговые выводы.....	467
7.21. Разбиение на обучающую, проверочную и тестовую выборки с учетом временной структуры для валидации временных рядов.....	468
7.22. Виды перекрестной проверки для данных формата «один объект – одно наблюдение» (присутствует ось времени).....	521
7.22.1. Перекрестная проверка расширяющимся окном.....	525
7.22.2. Перекрестная проверка скользящим окном.....	542
7.22.3. Перекрестная проверка расширяющимся/скользящим окном с гэпом.....	552
7.23. Перекрестная проверка для данных формата «один объект – несколько наблюдений» (присутствует ось времени).....	563
7.24. Многоклассовая классификация: подходы «один против всех», «один против одного» и «коды, исправляющие ошибки».....	567
7.24.1. Подход «один против остальных» или «один против всех» («one versus rest», «one versus all»).....	568
7.24.2. Подход «один против одного» («one versus one»).....	573
7.24.3. Подход «коды, исправляющие ошибки» («error-correcting output codes»).....	592

ЧАСТЬ 3. ДРУГИЕ ПОЛЕЗНЫЕ БИБЛИОТЕКИ.....602

1. Библиотеки визуализации matplotlib, seaborn и plotly.....602

1.1. Matplotlib.....	602
1.2. Seaborn.....	621
1.3. Plotly.....	629

2. Библиотека прогнозирования временных рядов ETNA.....634

2.1. Общее знакомство.....	634
2.2. Создание объекта TSDataset.....	641
2.3. Визуализация рядов объекта TSDataset.....	645
2.4. Получение сводки характеристик по объекту TSDataset.....	646
2.5. Модель наивного прогноза.....	647
2.6. Модель скользящего среднего.....	654

2.7. Модель сезонного скользящего среднего	658
2.8. Модель SARIMAX.....	662
2.9. Модель Хольта–Винтерса (модель тройного экспоненциального сглаживания, модель ETS).....	671
2.10. Модель Prophet.....	677
2.11. Модель CatBoost	689
2.12. Модель линейной регрессии с регуляризацией «эластичная сеть»....	709
2.13. Объединение процедуры построения модели, оценки качества и визуализации прогнозов в одной функции	714
2.14. Перекрестная проверка нескольких моделей	717
2.15. Ансамбли	722
2.16. Стекинг	724
2.17. Создание собственных классов для обучения моделей.....	725
2.18. Импутация пропусков	741
2.19. Работа с трендом и сезонностью	751
2.20. Обработка выбросов	766
2.21. Собираем все вместе.....	772
2.22. Модели нейронных сетей	787
2.23. Оптимизация гиперпараметров с помощью Optuna от разработчиков	789
Ответы на вопросы с собеседований.....	794