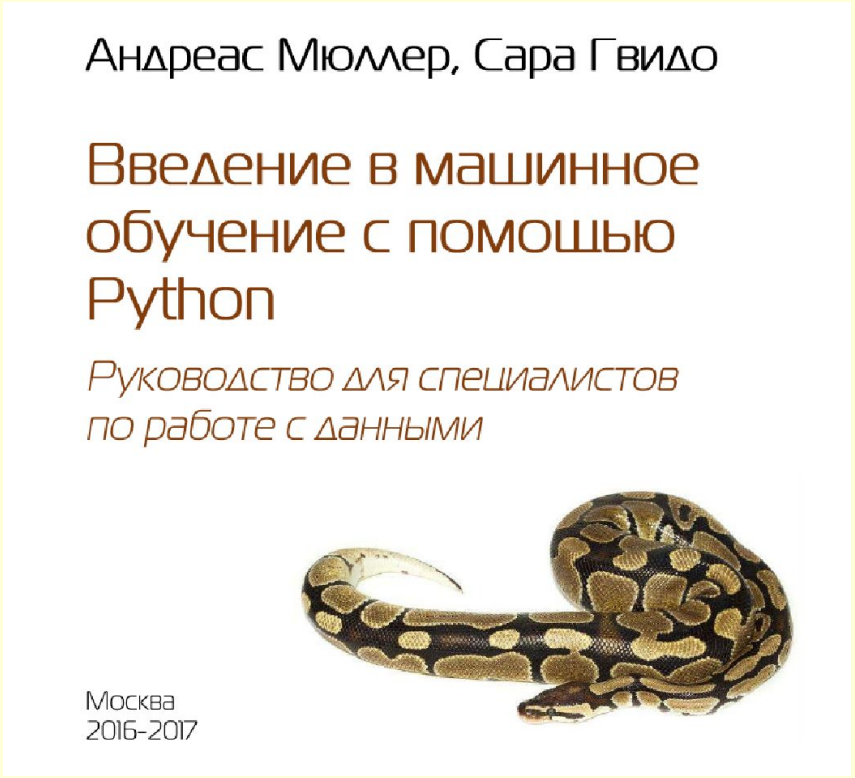
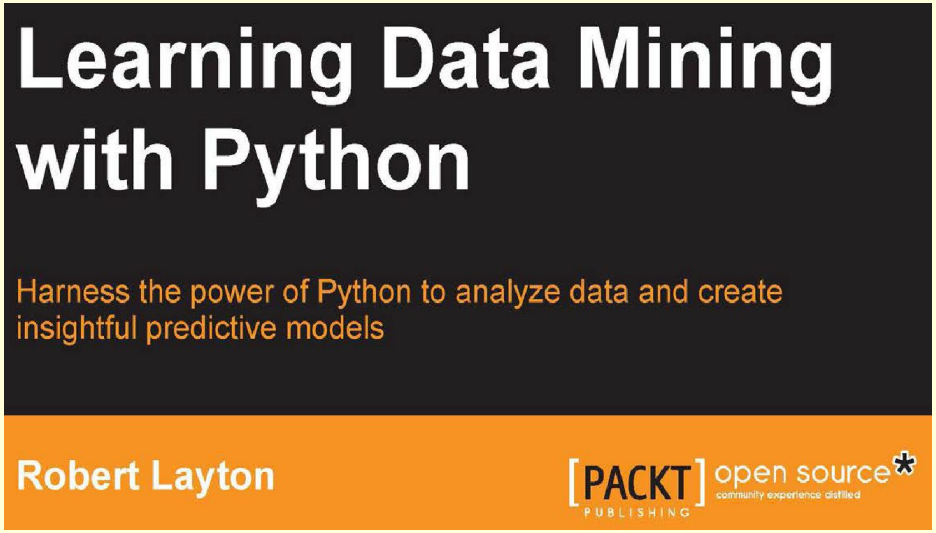
Обработка больших данных

ЛИТЕРАТУРА


Андреас Мюллер, Сара Гвидо
Введение в машинное обучение с помощью Python
Руководство для специалистов по работе с данными
Москва
2016-2017


Learning Data Mining with Python
Harness the power of Python to analyze data and create insightful predictive models
Robert Layton
[PACKT] open source
PUBLISHING


O'REILLY
High Performance Python
PRACTICAL PERFORMANT PROGRAMMING FOR HUMANS

Python 2 и Python 3

Python 3 - более новая версия.
Иногда код, написанный на Python 2, некорректно работает в Python 3.
Будем использовать Python 3.

Python является кросс-платформенным языком программирования
-рекомендуютUbuntu,
-Но работает и из под Windows, Macs, а также других вариантов Linux

Загрузка с сайта https://www.python.org/downloads/ версии Python3.6.8
(под ОС Windows XP и ниже не идет)
Проверить версию!

IPython Notebook

Среды для удобства работы: Jupyter, JupyterLab, Anaconda Python/R и др.

Установка из командной строки Windows cmd: **pip install ipython[all]**
Запуск cmd: **ipython3 notebook**
Запустится в веб-браузере Jupyter
[Удобнее работать в JupyterLab: инсталл. **pip install jupyter lab**,
запуск: **jupyter lab**
Запустится JupyterLab]

scikit-learn

Библиотека алгоритмов, данных, утилит, frameworks и др. библиотеки
Установка из командной строки Windows cmd: **pip install sklearn**

Пример 1. Качественный анализ данных

Скопировать файл affinity_dataset.txt

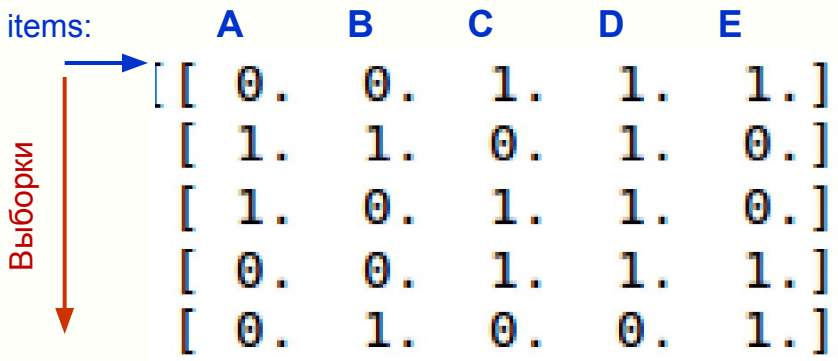
[в папку python (по умолчанию при установке C:\Users\User), или указывать путь]

```
import numpy as np
dataset_filename = "affinity_dataset.txt“
X = np.loadtxt(dataset_filename)                    # X = np.loadtxt(r “d:\ п у т ь \affinity_dataset.txt“)
```

Это двумерный массив – матрица 100×5:

0 = присутствует в выборке
1 = отсутствует

Печать первых 5 строк: print(X[:5])



Задача: Определить, есть ли зависимость между items в выборках

(если появляется какое-то item X1 из набора X={A,B,C,D,E}, то будет ли, как правило, присутствовать в этой же выборке item X2 из того же набора X?

[если покупают фрукты россыпью, то обычно покупают и пакет]

Пример 1. Качественный анализ данных

- Открыть файл данных в Excel. Посмотреть структуру. Посчитать кол-во 1 по каждому item. Решить задачу в Excel для D+E, используя правила выделения ячеек.

- Загрузить файл. Вывести на печать количество выборок и количество items (features) , распечатать первые 5 строк матрицы.

- Задать названия items={A,B,C,D,E}, посчитать, сколько всего раз выпадает D ?

Пример для D (item [3], т.к. индексы с 0!!)

```
num_D = 0
for sample in X:
    if sample[3] == 1:
        num_D +=1
        print("{0} raz vypalo D".format(num_D))
36 raz vypalo D
```

- Посчитать, сколько всего раз одновременно выпадают D и E ? Решение методом перебора

Пример 1. Качественный анализ данных

Пример реализации для произвольной пары

```
#Составляем правила: если X1=1 и X2=1 – то valid, иначе – invalid; считаем количество совпадений X1=X2=1
from collections import defaultdict
valid_rules = defaultdict(int)
invalid_rules = defaultdict(int)
num_X1vsego = defaultdict(int)

for sample in X:                          #Цикл делаем по выборкам
    for itemX1 in range(4):               #цикл от 0 до 4 по items
        if sample[itemX1] == 0: continue  #не интересно, продолжаем
        num_X1vsego[itemX1] += 1          #считаем общее число выпаданий X1=1, для расчета вероятности совпадений с X2

for itemX2 in range(n_features):          #Если выпало X1=1, то проверяем второе правило, что X2=1.
    if itemX1 == itemX2: continue         #НО надо НЕ учитывать X1=X1!!, «перескочить» X1
if sample[itemX2] == 1:                   #учитываем совпадение X1=X2=1:
   valid_rules[(itemX1, itemX2)] += 1
    else:
     invalid_rules[(itemX1, itemX2)] += 1
```

- Рассчитать статистические показатели: сколько раз выпала пара {D,E}, какая вероятность появления E при наличии D? Решение методом перебора

Пример 1. Качественный анализ данных

```
# Статистика, вероятность совпадений X1=X2=1 относительно общего числа выпаданий только X1
# (т.е. когда X1=1, а X2 не выпало, X2=0)
support = valid_rules
probabilityX12 = defaultdict(float)
for itemX1, itemX2 in valid_rules.keys():
    rule = (itemX1, itemX2)
      probabilityX12[rule] = valid_rules[rule] / num_X1vsego[itemX1]
```

- Сделать код для любой пары X1, X2 из X={A,B,C,D,E}. Вывод на печать для всех возможных пар

- Создать функцию расчета и вывода на печать для любой пары {X1, X2} из X={A,B,C,D,E}.

```
# Функция
def print_rule(itemX1, itemX2, support, probabilityX12, features):
    premise_name = features[itemX1]
    conclusion_name = features[itemX2]
    print("Rule: If X1 ravno {0} to X2 ravno {1}".format(premise_name, conclusion_name))
    print(" - Support: {0}".format(support[(itemX2, itemX2)]))
    print(" - Confidence: {0:.3f}".format(probabilityX12[(itemX1, itemX2)]))
```

```
# Вызов функции, проверка кода
itemX1 = 1
itemX2 = 3
print_rule(itemX1, itemX2, support, probabilityX12, features)
```

Пример 1. Качественный анализ данных

Визуализация и анализ данных:

-сортировать по парам по убыванию их совместных реализаций;

-сортировать по вероятности появления X2 у тех случаях, когда выпало X1;

В чем разница этих сортировок? Пояснить суть.

-Импортировать результаты в Excel и создать «Отчет» по результатам анализа в наглядной и легко воспринимаемой форме (графики, таблицы и т.п.)

Придумать, как упростить представление полученных результатов для их лучшего визуального восприятия.