

1. Основы языка Python

Часть 1. Python и интеллектуальный анализ данных (Data Mining) – введение.

В этом разделе:

- Зачем нужен Python при анализе данных?
- Как установить пакет Anaconda и начать с ним работать?

Александр Владимирович Толмачев

axtolm@gmail.com

Определения. Интеллектуальный анализ данных (Data Mining)

1) Data Mining — это процесс обнаружения в «сырых» данных знаний, необходимых для принятия решений в различных сферах человеческой деятельности (Григорий Пятецкий-Шапиро, 1992 г.).

При этом знания должны быть ранее неизвестными, нетривиальными, практически полезными и доступными интерпретации.

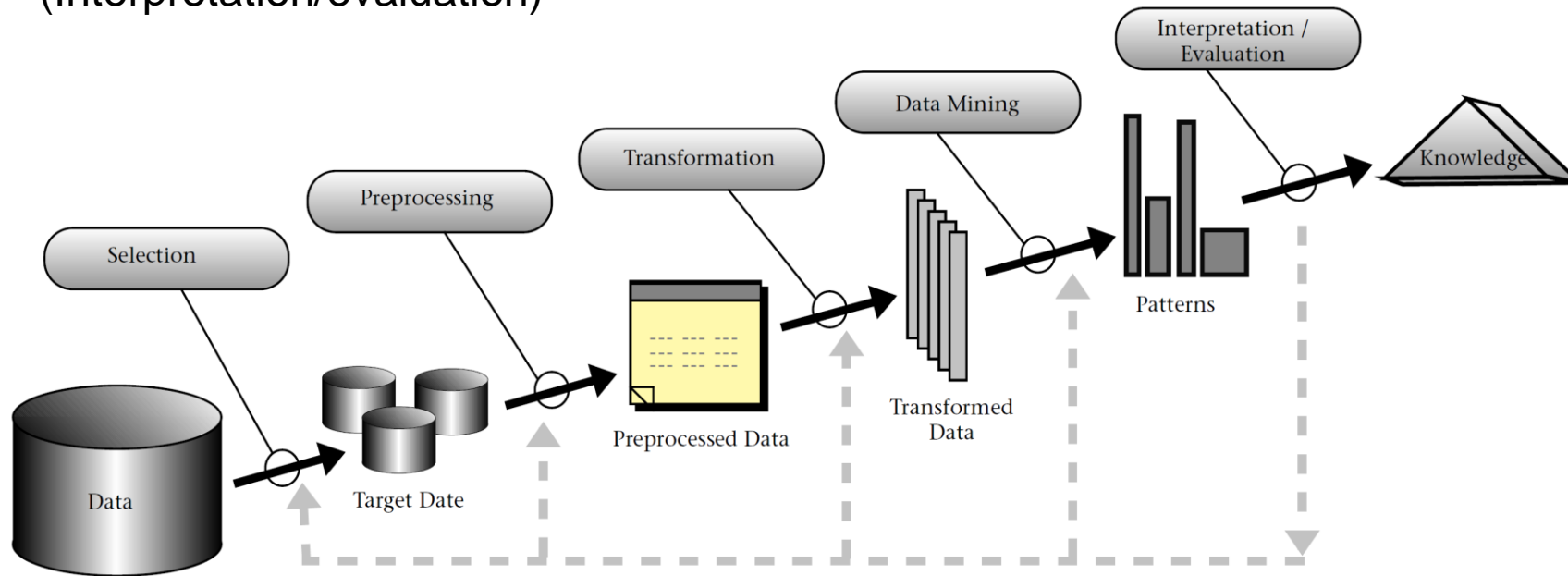
2) Data Mining — это современная концепция анализа данных, предполагающая, что:

- данные могут быть неточными, неполными, противоречивыми, разнородными, и при этом иметь гигантские объёмы;
- алгоритмы анализа данных могут обладать «элементами интеллекта», в частности, способностью обучаться по прецедентам, а их разработка также требует значительных интеллектуальных усилий;
- процессы переработки сырых данных в информацию, а информации в знания не могут быть выполнены вручную и требуют нетривиальной автоматизации.

Data Mining - часть более общего процесса **извлечения знаний из баз данных** («Knowledge Discovery in Databases" или KDD)¹⁾.

Этапы KDD:

1. Отбор данных (Selection)
2. Предварительная обработка данных (Pre-processing)
3. Преобразование данных (Transformation)
4. Интеллектуальный анализ данных (Data Mining)
5. Интерпретация и оценка результатов (Interpretation/evaluation)



Python –
полноценный
инструмент KDD,
используемый на
всех этапах.

Почему Python?

- прост в освоении,
- широко распространен,
- много библиотек для решения прикладных задач,
- open source проект,
- кроссплатформенный, работает с CPU и GPU,
- активно развивается,
- есть своя философия.

¹⁾ Fayyad U., Piatetsky-Shapiro G., & Smyth P. From data mining to knowledge discovery in databases. Ai Magazine, vol. 17, no. 3, pp. 37-54, 1996

Подробнее о Data Mining

Шесть классов задач, которые решают с помощью Data Mining:

1. Классификация (Classification)
2. Регрессия (Regression)
3. Кластеризация (Clustering)
4. Обобщение (Summarization)
5. Моделирование зависимостей (Dependency modelling)
6. Обнаружение аномалий (Change and deviation detection)

**Вручную их не выполнить, а для автоматизации нужны инструменты.
Мы будем использовать Python.**

Работать с Python будем с помощью пакета Anaconda Individual Edition для Windows

- Популярность у 25M+ пользователей в мире.
- Установка на Linux, Windows, Mac OS.
- Основные библиотеки для работы с данными (250+) идут в комплекте поставки.
- 7.5K+ библиотек доступны в облаке.
- Open-source для Data Mining и Machine Learning.



Установка Anaconda Individual Edition:

Скачать установщик <https://www.anaconda.com/products/individual> и запустить его.

Подробности процесса установки: <https://docs.anaconda.com/anaconda/install/>

Anaconda Individual Edition - когда и что из компонентов будем использовать в работе



IDE Spyder

Интегрированная среда разработки

- код, требующий отладки
- относительно объемный код
- код с длительным временем исполнения



Для отладки есть все необходимое:

- ✓ менеджер переменных
- ✓ точки останова и пошаговое выполнение
- ✓ профайлер



Jupyter Notebook

Web приложение для работы с документами (текст + код на Python + результаты его выполнения)

- алгоритмы с пошаговым выводом результатов
- код и результаты, которыми нужно поделиться
- отчеты для программистов и непрограммистов



Conda

Менеджер для управления библиотеками и окружением

- установка библиотек
- создание виртуальных окружений



IDE Spyder – интегрированная среда разработки в составе Anaconda Distribution

<https://www.spyder-ide.org/>

кнопки запуска
и отладки кода

точка останова
для отладки

редактор кода

менеджер
переменных

КОНСОЛЬ

The screenshot displays the Spyder IDE interface with the following components:

- Code Editor:** Contains Python code for data extraction and outlier detection using pandas.
- Variable Explorer:** Shows a table of variables and their values.
- Console:** Displays the output of the executed code.

Code Editor Content:

```
1 # -*- coding: utf-8 -*-
2
3 DATA_PATH = 'D:/YandexDisk/5_python/_python_learning/_py_bachelors_09_2021/'
4 FILE_LA_F2017 = 'lesson5_data_ENGM_F2017_train_atolm092021' # Имя файла
5 FILE_EXT = '.csv'
6
7 print(DATA_PATH+FILE_LA_F2017+FILE_EXT)
8
9 import pandas as pd
10 LA_F2017 = pd.read_csv(DATA_PATH+FILE_LA_F2017+FILE_EXT, ";") # выкачиваем данные из файла
11
12 # Посмотрим описательную статистику по выборке
13 LA_F2017.describe()
14
15 # Пропуски есть в AVG и RAVG. Найдём строки с пропусками в исходной базе
16 # Воспользуемся DataFrame.isna() и DataFrame.isin() - whether each element in the DataFrame is contained in values.
17 na_df = LA_F2017[LA_F2017.isna()['AVG'].isin([True]) | LA_F2017.isna()['RAVG'].isin([True])]
18 # na_list = LA_F2017[LA_F2017.isna()['AVG'].isin([True]) | LA_F2017.isna()['RAVG'].isin([True])].index.tolist()
19 na_list = na_df.index.tolist()
20
21 # выбросы
22 outlier_df = LA_F2017.query('AVG>1 or RAVG>1')
23 outlier_list = LA_F2017.query('AVG>1 or RAVG>1').index.tolist()
24
25 LA_F2017_Copy1 = LA_F2017.copy() # Сделаем копию, чтобы удалять в ней
26 # получим в виде списка индексы строк, которые надо удалить - ПРОПУСКИ
27 del_list = list(set(na_list+outlier_list)) # уберем подтопы
28
29 LA_F2017_Copy1 = LA_F2017_Copy1.drop(del_list) # удаляем строки с индексами из списка
30 ddf = LA_F2017_Copy1.describe()
31
32 LA_F2017_Copy2 = LA_F2017.copy()
33 LA_F2017_Copy2 == LA_F2017
34
```

Variable Explorer Table:

Name	Type	Size	Value
DATA_PATH	str	62	D:/YandexDisk/5_python/_python_learning/_py_bachelors_09_2021/
ddf	DataFrame	(8, 3)	Column names: ID_student, AVG, RAVG
del_list	list	5	[1, 2, 4, 5, 8]
FILE_EXT	str	4	.csv
FILE_LA_F2017	str	41	lesson5_data_ENGM_F2017_train_atolm092021
LA_F2017	DataFrame	(831, 4)	Column names: ID_student, AVG, RAVG, gender
LA_F2017_Copy1	DataFrame	(826, 4)	Column names: ID_student, AVG, RAVG, gender
LA_F2017_Copy2	DataFrame	(831, 4)	Column names: ID_student, AVG, RAVG, gender
na_df	DataFrame	(3, 4)	Column names: ID_student, AVG, RAVG, gender
na_list	list	3	[1, 4, 5]
outlier_df	DataFrame	(3, 4)	Column names: ID_student, AVG, RAVG, gender
outlier_list	list	3	[1, 2, 8]

Console Output:

```
...: outlier_list = LA_F2017.query('AVG>1 or RAVG>1').index.tolist()
...:
...: LA_F2017_Copy1 = LA_F2017.copy() # Сделаем копию, чтобы удалять в ней
...: # получим в виде списка индексы строк, которые надо удалить - ПРОПУСКИ
...: del_list = list(set(na_list+outlier_list)) # уберем подтопы
...:
...: LA_F2017_Copy1 = LA_F2017_Copy1.drop(del_list) # удаляем строки с индексами из списка
...: ddf = LA_F2017_Copy1.describe()
...:
...: LA_F2017_Copy2 = LA_F2017.copy()
...: LA_F2017_Copy2 == LA_F2017
D:/YandexDisk/5_python/_python_learning/_py_bachelors_09_2021/
lesson5_data_ENGM_F2017_train_atolm092021.csv
Out[1]:
  ID_student  AVG  RAVG  gender
0      True  True  True   True
1      True False  True   True
2      True  True  True   True
3      True  True  True   True
4      True  True False  True
..      ...   ...   ...   ...
826     True  True  True   True
827     True  True  True   True
828     True  True  True   True
829     True  True  True   True
830     True  True  True   True

[831 rows x 4 columns]

In [2]:
```



<https://jupyter.org/>

The Jupyter Notebook – это open-source **web приложение**, которое позволяет создавать **документы**, содержащие **тексты и рисунки**, код на **Python** и **результаты его выполнения**; конвертировать документы в HTML и другие форматы.

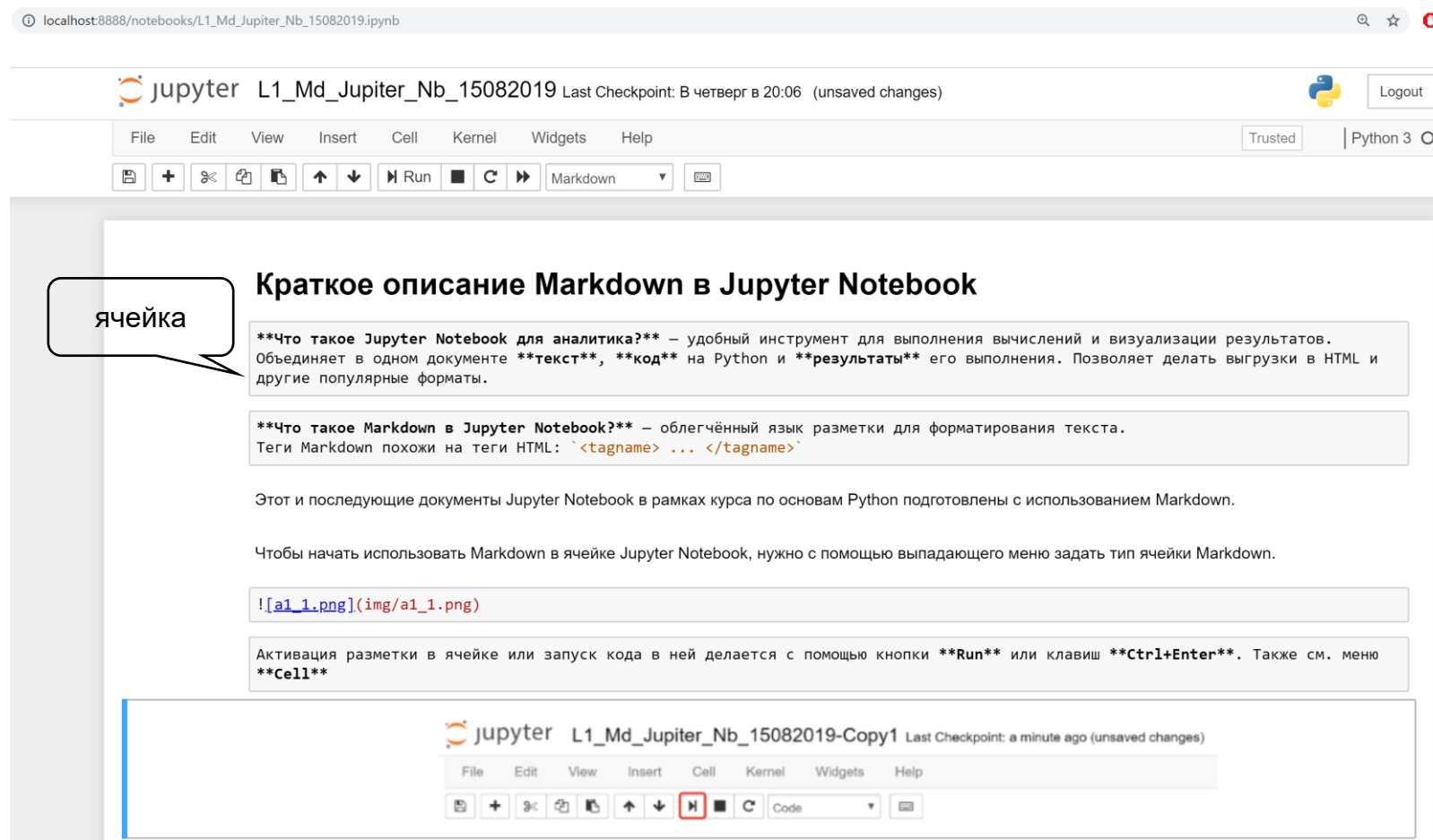
Поставляется в составе
Anaconda Distribution

Ячейка – ключевой элемент

В ячейке может быть:

- **текст**, который можно форматировать с помощью Markdown
- **код на Python** и результаты его выполнения
- **рисунок**

Есть еще **JupyterLab** – Web IDE для Jupyter notebooks





Conda – это open source менеджер для управления библиотеками и окружением, который работает на Windows, macOS, Linux.

Поставляется в составе
Anaconda Distribution

Используем для установки
библиотек с командной
строки в терминале:

conda install PKGNAME

PKGNAME – имя
устанавливаемого пакета

```
C:\Windows\system32\cmd.exe

(base) C:\Users\atolm>conda info

      active environment : base
      active env location : C:\Anaconda3
            shell level : 1
      user config file : C:\Users\atolm\.condarc
 populated config files : C:\Users\atolm\.condarc
         conda version : 4.7.10
    conda-build version : 3.18.8
         python version : 3.7.3.final.0
    virtual packages : __cuda=10.0
      base environment : C:\Anaconda3 (writable)
        channel URLs : https://repo.anaconda.com/pkgs/main/win-64
                      https://repo.anaconda.com/pkgs/main/noarch
                      https://repo.anaconda.com/pkgs/r/win-64
                      https://repo.anaconda.com/pkgs/r/noarch
                      https://repo.anaconda.com/pkgs/msys2/win-64
                      https://repo.anaconda.com/pkgs/msys2/noarch
         package cache : C:\Anaconda3\pkgs
                        C:\Users\atolm\.conda\pkgs
                        C:\Users\atolm\AppData\Local\conda\conda\pkgs
      envs directories : C:\Anaconda3\envs
                        C:\Users\atolm\.conda\envs
                        C:\Users\atolm\AppData\Local\conda\conda\envs
         platform : win-64
        user-agent : conda/4.7.10 requests/2.22.0 CPython/3.7.3 Windows/10 Windows/10.0.18362
       administrator : False
           netrc file : None
       offline mode : False
```

Запуск приложений Spyder, Jupyter Notebook, Conda из пакета Anaconda

Запуск Spyder

Windows menu Start

- Anaconda3 (64-bit)
- **Spyder** (anaconda3)

Запускается как самостоятельное приложение в своем окне

Запуск Conda

Windows menu Start

- Anaconda3 (64-bit)
- **Anaconda Powershell Prompt (anaconda3)**

Запускается терминал с командной строкой

Запуск Jupyter Notebook

Windows menu Start

- Anaconda3 (64-bit)
- **Jupyter Notebook** (anaconda3)

Запускается в окне браузера

Подведем итоги:

- Мы рассмотрели, что такое Data Mining и какие задачи можно решать с его помощью.
- Узнали, какое место занимает Data Mining в более общем процессе извлечения знаний KDD.
- Выяснили, зачем при анализе данных нужен язык Python.
- Установили пакет Anaconda Individual Edition на персональный компьютер.
- Научились запускать приложения Spyder, Jupyter Notebook, Conda.