

О Python и Jupyter Notebook

Почему Python?

Язык Python разрабатывался так, чтобы на нем можно было легко писать код. Вы потратите намного больше времени на код на C++, чем на питоновский код. Правда за это приходится платить низкой производительностью — программа на C++ будет в десятки раз быстрее, чем программа на Питоне. Но если вы пишете какой-нибудь простой исследовательский код, то тут вы потратите намного меньше времени и нервов, если напишете код на Питоне.

Так же на Питоне написано огромное количество библиотек для анализа данных (описание некоторых ниже) с очень хорошей документацией.

Для примера посмотрите, как просто можно написать код, который генерирует выборку размера 1000 из двумерного нормального распределения $\mathcal{N}\left(\begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\right)$, считает выборочное среднее и наносит точки на график.

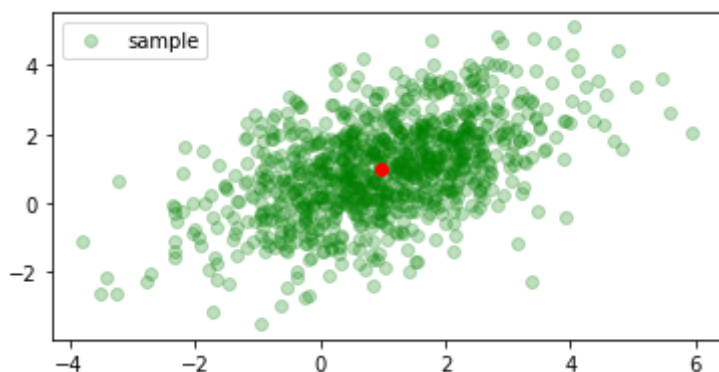
In [1]:

```
# Импортируем всякие библиотеки
import numpy as np
from scipy.stats import multivariate_normal
import matplotlib.pyplot as plt

sample = multivariate_normal.rvs(mean=[1, 1], cov=[[2, 1], [1, 2]], size=1000) # Г
print(np.mean(sample, axis=0)) # Подсчет и печать среднего значения

# Рисуем график
plt.figure(figsize=(6, 3)) # Инициализируем график и задаем его размеры
plt.scatter(sample[:, 0], sample[:, 1], color='green', alpha=0.25, label='sample')
plt.scatter(np.mean(sample[:, 0]), np.mean(sample[:, 1]), color='red') # Наносим ц
plt.legend() # Ставим легенду
plt.show() # Показываем график
```

```
[ 0.98934062  1.00714285]
```



Все еще сомневаетесь в простоте? Попробуйте написать такой же код на C++ или Java.

Python

В настоящее время широко распространены две версии Питона — Python 2 и Python 3. Эти версии похожи, но могут быть не совместимыми. То есть если у вас есть программа на Python 2, то она может не заработать на Python 3, и наоборот. В нашем курсе мы поддерживаем только Python 3, а именно версию 3.5. Это означает, что весь выдаваемый код будет корректно работать на этой версии. Почти весь код, который вы отправляете нам вместе с выполненной работой, мы не запускаем, поэтому использовать можно любую версию Питона. Однако, если в задании сказано, что код будет запускаться, то он должен корректно работать на Python 3.5.

Из операционных систем поддерживается только Ubuntu. Скорее всего больших проблем с Mac быть не должно. Конечно, вы можете использовать Windows, но можете и получить за это целый зоопарк проблем.

Если у вас Ubuntu, то Python скорее всего уже установлен. Попробуйте в терминале следующие команды для запуска интерактивного режима работы:

```
python или python3 или python2
```

Выход: Ctrl+D

Режим работы, в котором выполнится код из файла main.py

```
python main.py
```

Помощь: `help(X)`, где `X` — то, по чему нужна помощь. Выход из помощи: `q`.

Обновить Питон, например, с версии 3.4 на 3.5 можно так

<https://gist.github.com/larainema/a05d2f28cc7d944da6f6>

(<https://gist.github.com/larainema/a05d2f28cc7d944da6f6>)

Установка Jupyter и дополнительных библиотек

Чтобы сохранить свое психическое здоровье, устанавливайте Jupyter Notebook на не слишком старых версиях Ubuntu или на Mac. Инструкция по установке находится тут:

<http://jupyter.readthedocs.org/en/latest/install.html> (<http://jupyter.readthedocs.org/en/latest/install.html>)

Ниже описана инструкция по установке для Ubuntu.

Прежде чем перейти к установке, рекомендуется обновить `pip` командой

```
sudo pip3 install --upgrade pip
```

Затем можно поставить Jupyter командой

```
sudo pip3 install jupyter
```

Так же вам понадобятся несколько достаточно распространенных библиотек, которые можно установить командой

```
sudo pip3 install numpy scipy matplotlib scikit-learn
```

Если вы хотите использовать Python 2 вместо Python 3, то везде пишите `pip2` вместо `pip3`.

После установки запуск осуществляется командой

```
jupyter notebook
```

Через несколько секунд должна открыться страница в браузере со списком файлов. Если этого не произошло, перейдите по ссылке, которая будет указана при запуске. Обычно это <http://localhost:8888/> (<http://localhost:8888/>)

Что делать, если пакет установлен, а Jupyter его не видит?

В этом случае ему нужно прописать путь до пакета. Для этого сначала откройте Питон в терминале и выполните (например, для пакета `numpy`):

```
import numpy
```

```
numpy
```

Напечатается путь, например, такой `/usr/local/lib/python3.5/dist-packages/numpy/__init__.py`. Теперь в Jupyter нужно написать такой код

In [2]:

```
import sys
sys.path.append('/usr/local/lib/python3.5/dist-packages/numpy')
import numpy
```

Онлайн Jupyter Notebook

Можно ничего не устанавливать и работать с сайтом <https://try.jupyter.org/> (<https://try.jupyter.org/>)

На сайте можно создать новый файл (New -> Python 2/3) или загрузить файл с компьютера с помощью Upload.

Будьте внимательны! **Сайт не сохраняет ваши файлы** и удаляет их после закрытия страницы. Чтобы загрузить файл себе откройте его нажмите File -> Download as -> ipynb.




Можно так же пользоваться сервисом <https://cloud.sagemath.com/> (<https://cloud.sagemath.com/>), который немного тормознущий, но сохраняет все файлы.


Работа с Python в Jupyter Notebook

Чтобы создать новый файл, кликните New -> Python.

Ноутбук состоит из ячеек, которые бывают текстовыми (Markdown) и кодовыми (Code). Выбрать тип ячейки можно на панели управления.

Работа с ячейками:

- Выбор ячейки — нажмите на нее мышкой.
- Редактирование — нажмите на нее два раза.
- Запуск ячейки — SHIFT+ENTER или нажмите на кнопку  на панели.
- Добавление новой ячейки — нажмите на кнопку  на панели.
- Удаление ячейки — нажмите на кнопку  на панели.
- Перемещение ячейки — нажмите на вертикальные стрелки.

Кодовые ячейки содержат в себе код на Python. Если код должен что-то печатать, то оно отобразится сразу под ячейкой. Перед каждой ячейкой с кодом есть обозначение In [N] :, которое означает, что код в этой ячейке выполнен N-ым по счету. В этой ячейке доступны все переменные предыдущих выполненных ячеек, даже если вы их удалите. Выполнение ячеек происходит в том порядке, в котором вы их запускаете, а не в том, в котором они расположены в ноутбуке. Чтобы перезагрузить ноутбук, стерев все данные в памяти, нажмите на restart. Чтобы сохранить ноутбук нажмите  на панели. Все отображаемые данные при этом сохраняются, включая вывод, но после перезагрузки среды (или загрузки в онлайн сервис) для дальнейшей работы придется заново произвести все вычисления.

Текстовые ячейки содержат в себе обычный текст, который может включать в себя формулы LaTeX и html-команды.

Щелкните два раза на эту ячейку, чтобы посмотреть, как работать с формулами

$$E\xi = \int_{\Omega} \xi dP$$

Щелкните два раза на маркированный список выше, чтобы посмотреть, как работать с html.

Изучение Python

Кроме наших питон-ноутбуков можете воспользоваться также следующими ресурсами.

<http://docs.scipy.org/doc/scipy/reference/stats.html> (<http://docs.scipy.org/doc/scipy/reference/stats.html>) (англ.)
— полное описание модуля stats

Scikit-Learn — библиотека алгоритмов машинного обучения с множеством примеров и наборами данных. Она вам очень пригодится в курсе машинного обучения, но что-то (например, метод наименьших квадратов) может пригодиться уже сейчас.

<http://scikit-learn.org/> (<http://scikit-learn.org/>) (англ.) — очень подробное описание библиотеки

<http://habrahabr.ru/company/mlclass/blog/247751/> (<http://habrahabr.ru/company/mlclass/blog/247751/>) — немного примеров на русском

Matplotlib — библиотека для визуализации данных

<http://matplotlib.org/> (<http://matplotlib.org/>) (англ.)

<http://matplotlib.org/gallery.html> (<http://matplotlib.org/gallery.html>) (англ.) — тысячи примеров

<http://jenyay.net/Matplotlib/Matplotlib> (<http://jenyay.net/Matplotlib/Matplotlib>) и

<http://pyviy.blogspot.ru/2009/09/matplotlib.html> (<http://pyviy.blogspot.ru/2009/09/matplotlib.html>) — хорошие описания на русском

Так же желающие могут изучить пакет **Pandas** для работы с табличными данными, но нам он скорее всего не понадобится.