Big Data analysis with Python

State of the art and overview of the applications in the area of the diploma thesis

Supervisor: Dr hab. inż. Dariusz Król
Presented by: Oleksii Kyrylchuk 223224
13.10.2017

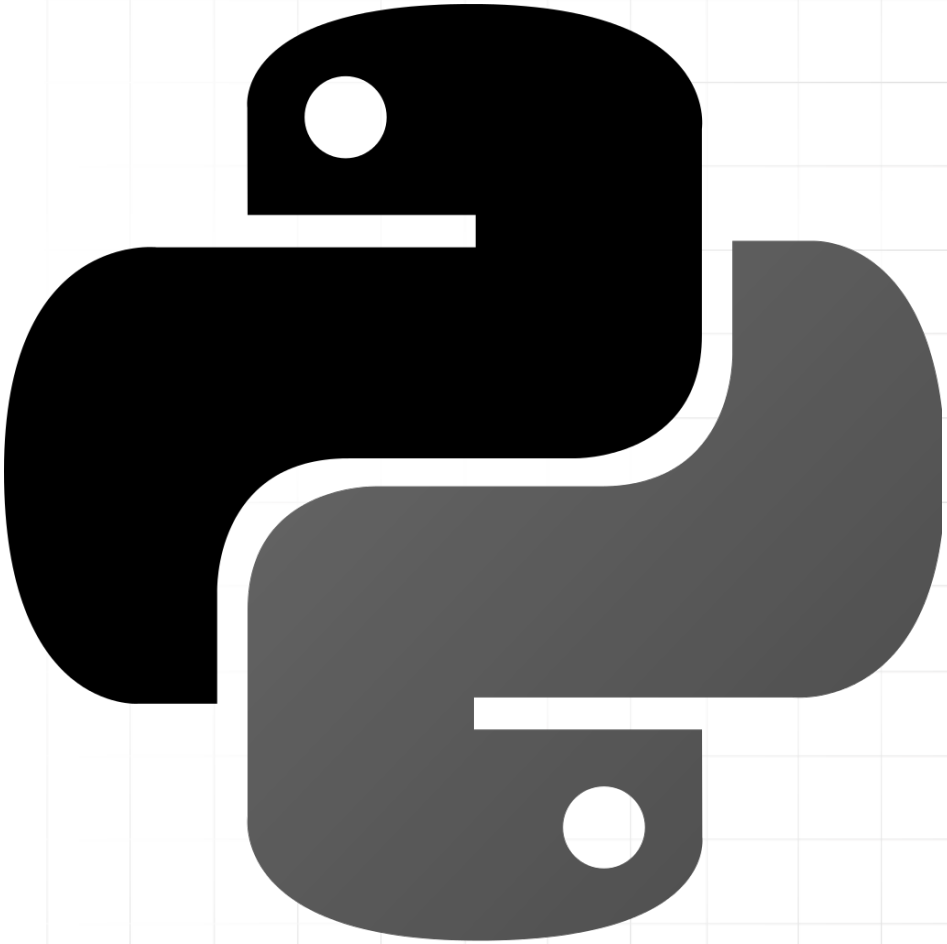
HR EXCELLENCE IN RESEARCH

Wrocław University of Science and Technology

My topic

”Big Data analysis with Python”

“Przetwarzanie dużych zbiorów danych w środowisku Python”



Why did I choose this topic?

Why big data?

- Analyze stock data
  - Predict the future stock price
  - Plot the stock data over time
  - …
- Analyze facebook data
  - Plot the event location data
  - Analyze the popularity of posts
  - …
- Analyze YouTube data
  - Make a computer generated song/video

Why Python?

- Quickly achieve good results
- Wide variety of tools available
- Very popular language in data science

Brief overview of the DS stack

Extract
Gather

- Web scraping, API scraping, Data source packages, Flat files, …

Transform
Clean

- NumPy, pandas, built-in lists, …

Load
Store

- APIs – sqllite3, psycopg2, hadoop, ORMs – SQLAlchemy, peewee …

Analyze

- Tensorflow, matplotlib, scikit-learn, bokeh, pandas, …

Step 1

Data extraction/gathering

Processing files

- CSV reader[1] – A built-in package
- Numpy[2] – A more advanced package, oriented on on scientific computing
- Pandas[3] – A package, that implements R's DataFrame, and has a lot of extra features, including plotting

[1] https://docs.python.org/3/library/csv.html
[2] http://www.numpy.org/
[3] http://pandas.pydata.org/

Data source packages

- There are packages, that allow you to download stock data automatically without having to scrape websites/APIs

- yahoo-finance[1]
- googlefinance[2]
- pandas-datareader[3] - fetches data from multiple sources

- Example:

```
import pandas_datareader as pdr

pdr.get_data_yahoo('AAPL') # Returns a DataFrame
```

[1] https://pypi.python.org/pypi/yahoo-finance
[2] https://pypi.python.org/pypi/googlefinance
[3] https://pandas-datareader.readthedocs.io

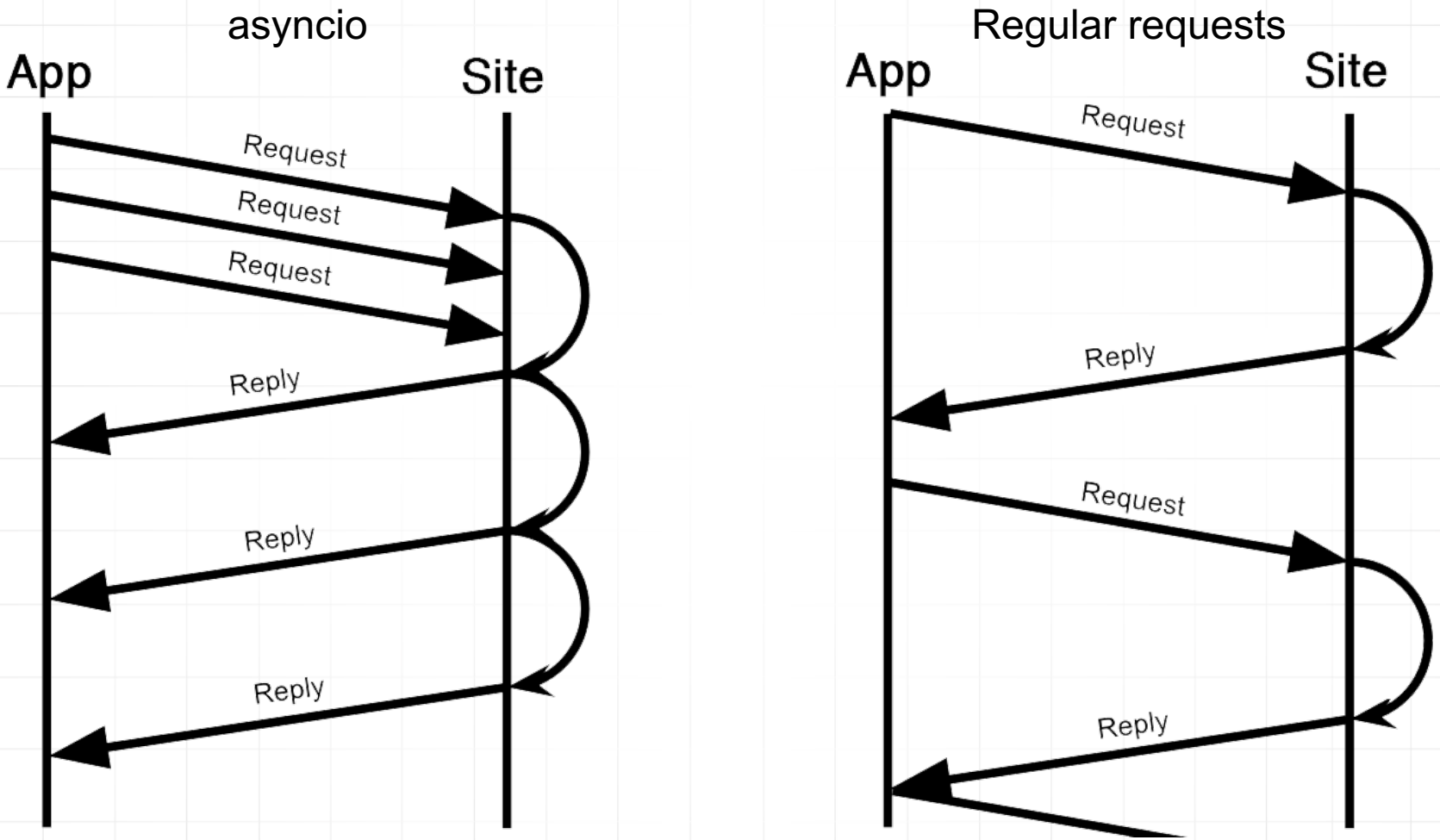
API scraping

- json[1] – a built-in package that allows us to convert json into a python dictionary

- urllib3[2] – a built-in http request package (synchronous)
- requests[3] – an external request package, that simplifies the code
- aiohttp[4] – a package, that allows you to asynchronously send http requests, which greatly improves the performance (asyncio/await syntax accessible in python 3.5+)

[1] https://docs.python.org/3/library/json.html
[2] https://urllib3.readthedocs.io/en/latest/
[3] http://docs.python-requests.org/en/master/
[4] http://aiohttp.readthedocs.io/en/stable/

API/web scraping - asyncio



Web scraping

- We use the same packages from the API scraping to download the data

- re[1] – a built-in package that implements regular expressions
- BeautifulSoup[2] – a package that allows us to navigate HTML much more easily
- scrapy[3] – a package for making web spiders, that download webpages recursively

[1] https://docs.python.org/3/library/re.html
[2] https://www.crummy.com/software/BeautifulSoup/
[3] https://scrapy.org/

Step 2

Data transformation, cleaning and storage

Transformation & cleaning

- Built-in lists[1]
- numpy[2] – a lower level package that gives you access to array and matrix classes with a lot of useful methods
- pandas[3] – one of the best packages for dealing with table data. Has an interface to import/export data to/from numpy, csv, zip files, etc.

[1] https://docs.python.org/3.6/tutorial/datastructures.html
[2] http://www.numpy.org/
[3] http://pandas.pydata.org/

Data Storage – APIs vs ORMs

- APIs (drivers)
  - sqlite3[1]
  - psycopg2[2]
- Use the database API to do everything
- Use SQL for querying
- Harder to code & maintain
- Usually faster

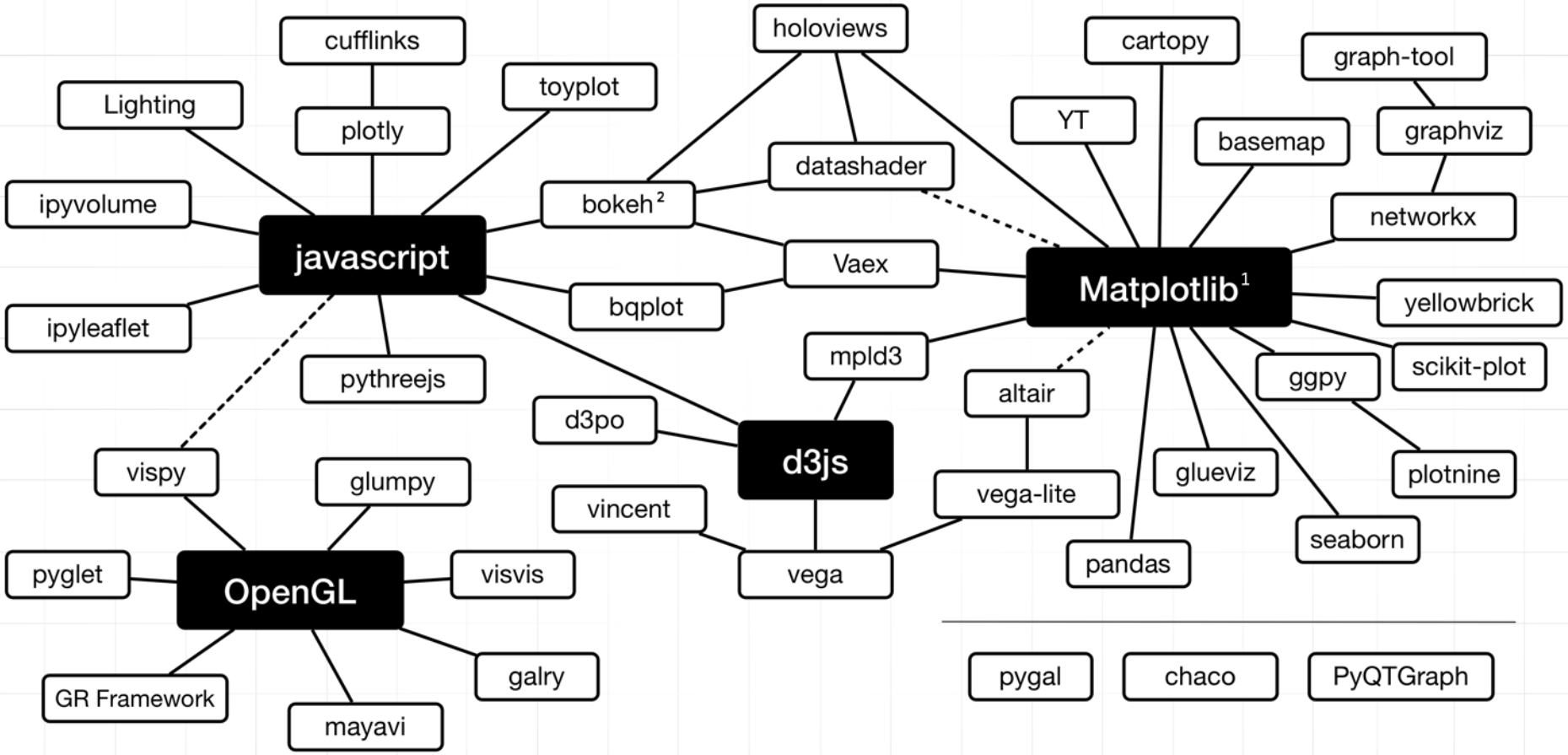
- ORMs
  - SQLAlchemy[3]
  - peewee[4]
- Easier to code & maintain
- They use the API packages to send the SQL to the database
- Sometimes slower

[1] https://docs.python.org/3/library/sqlite3.html
[2] http://initd.org/psycopg/
[3] https://www.sqlalchemy.org/
[4] http://docs.peewee-orm.com/en/latest/

Step 3

Data analysis

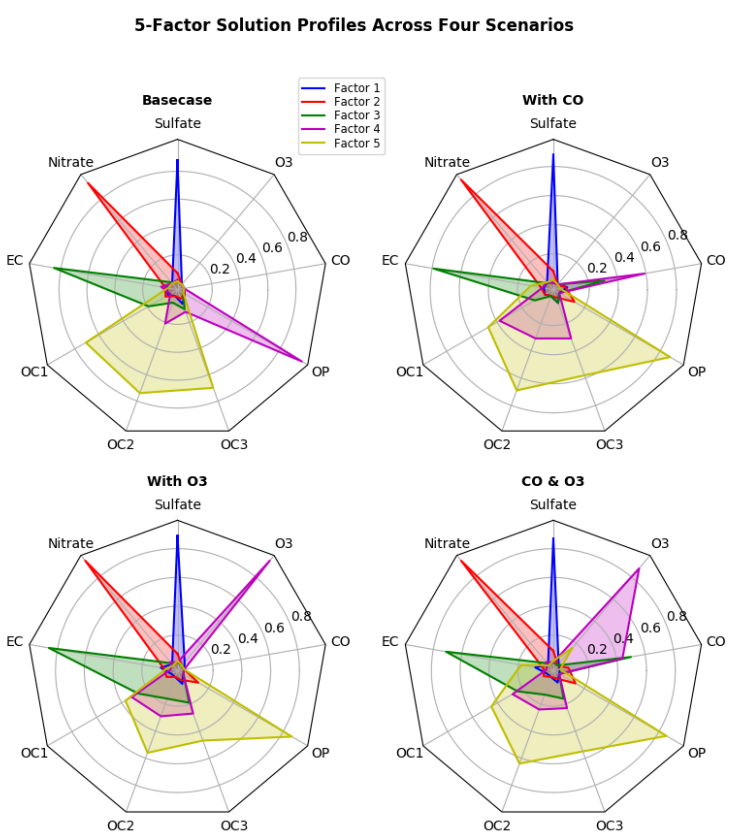
Data analysis – Visualization



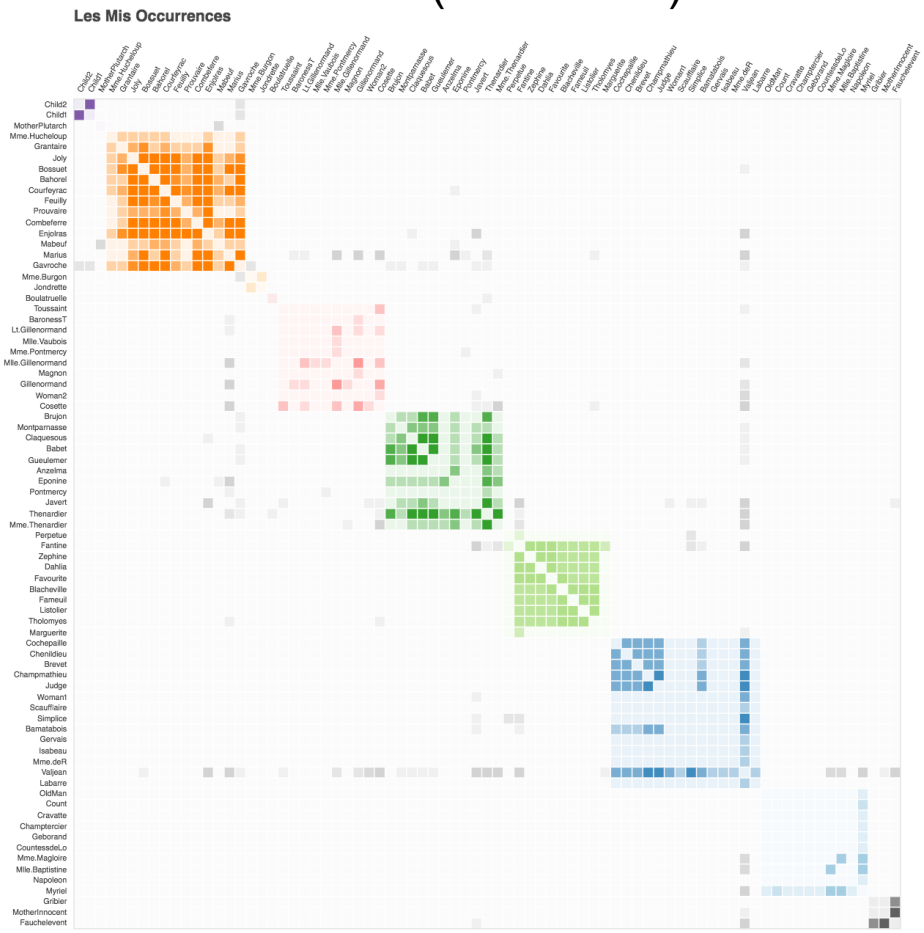
[1] https://matplotlib.org/
[2] https://bokeh.pydata.org/en/latest/

Data analysis – Example visualizations

Matplotlib[1]



Bokeh[2] (Interactive)



1 https://matplotlib.org/gallery.html

2 https://bokeh.pydata.org/en/latest/docs/gallery.html

Data analysis – Machine learning

- Scikit-learn [1] – the easies way to get the results
- Numpy [2], Scipy [3] – if we want to do everything from scratch
- Tensorflow [4] – a more efficient way to do everything from scratch
- Theano [5] – a competitor to Tensorflow
- Keras [6] – A high level package that can use either Tensorflow or Theano as its backend

[1] http://scikit-learn.org
[2] http://www.numpy.org/
[3] https://www.scipy.org
[4] https://www.tensorflow.org
[5] http://deeplearning.net/software/theano
[6] https://keras.io/

Similar applications

- There are a lot of apps that implement a similar structure, but most of them are corporate and operate on corporate datasets
- Example projects include
  - Analyzing stock data [1]
  - Analyzing medical data
  - Computer-generated art/art style [2] [3]
  - Computer-generated music
  - …

[1] http://sentdex.com/
[2] https://deepart.io/
[3] http://nightmare.mit.edu/

My thesis plan – A Roadmap

Task	Status	Month	March		April				May					June		J - S	October					November				Dec
		Week	13	14	15	16	17	18	19	20	21	22	23	24	25-27	28-40	41	42	43	44	45	46	47	48	49	50
Organization																										
Paperwork (Registering, etc.)	Completed														Session	Internship										
Specification	Completed																									
Studying available tools	In progress																									
Development																										
Implementing the facebook analyzer	In progress														Session	Internship										
Implementing the stock analyzer	In progress																									
Implementing the youtube analyzer	Not started																									
Writing																										
Introduction	In progress														Session	Internship										
Available technologies	Not started																									
Part 1 - Stock analysis	Not started																									
Part 2 - Facebook analysis	Not started																									
Part 3 - Youtube analysis	Not started																									

References

- Wes McKinney (2012) Python for Data Analysis – O'Reilly Media, Inc.
- Joel Grus (2015) Data Science from Scratch – O'Reilly Media, Inc.
- Jake VanderPlas (2017) The Python Visualization Landscape - PyCon 2017 https://www.youtube.com/watch?v=FytuB8nFHPQ
- Nikola Milosevic (2016) Equity forecast: Predicting long term stock price movement using machine learning