4. pySpark for Data Science (VL training)
by Mikołaj Kromka

bio: a software engineer and Apache Spark trainer at VirtusLab with background in Computer Science and Statistics, focused on theoretical and technical aspects of leveraging Big Data in Machine Learning. As a PhD candidate he conducts his research in the field of large, dynamic Complex Networks. In his spare time likes to climb, take photos and explore Cracow art museums.

Level: Medium - minimal knowledge of Python and SQL is required. No knowledge of Spark is needed.
Length: 1 day: 7 hours of training + 1 hour break
Approach: Learn by doing: have access to all materials and solve exercises on your own with a help from experienced trainer. Discuss new concepts and solutions with other participants.
Requirements: Students must bring their own laptops with Java 8 and conda

next sessions: to be announced in July

About

https://github.com/VirtusLab/pyspark-workshop

TODO

Participants
Data Scientists or Data Engineers with proficiency in Python and SQL
Data Analysts who want to gain an understanding on getting insights from Big Data
Benefits
Getting to know most popular Big Data technology - Apache Spark
Getting used to data wrangling and feature generation in scale using pySpark SQL
Learning how to develop and deploy ML models from Spark's ML library
Scaling and parallelizing custom Python models with pySpark
Learning caveats of pySpark from Engineers and Scientists with professional experience in the topic gained in the largest companies.
Outline

- Intro
- Apache Spark Overview
  - Practical aspects of using RDDs
  - Monitoring using Spark UI
  - pySpark as a Python API
- pySpark Basics (all parts are first introduced and 1 or more exercises follow)
  - Getting to know your Dataframe and Spark's Execution Model
  - SQL support
  - Spark Dataframe API for SQL-like operations (grouping, filtering, joining)
  - Extending Dataframes (adding new columns, using built-in UDFs, wide/narrow data transformations)
  - Date operations

    - Combining multiple queries together
    - Window functions
    - Complex aggregations (e.g. creating time series)
- Spark ML
    - Creating ML pipelines
    - Running models in production
    - Reusing old models
- Running custom Python models with pySpark
    - Dependency management
    - Custom Python code as UDF - benefits and caveats