Exploratory Data Analysis with Apache Spark

Hossein Falaki
@mhfalaki
About Databricks

Founded by creators of Apache Spark from UC Berkeley

We are dedicated to open source Spark
  › Largest organization contributing to Apache Spark
  › Drive the roadmap

We offer Spark as a service in the cloud
We wish all the people in the Bay Area who were affected by the earthquake a speedy recovery and a good night's sleep.
Large data
“Visualization is critical to data analysis.”

William S. Cleveland

But we often skip exploratory visualization with large data
Challenges

1. Interactivity with large data is challenging

2. Visual medium cannot accommodate as many pixels as data points
Solutions

1. Interactivity
   - In-memory computation
   - High parallelism
Fast & General distributed computing engine: batch, streaming, iterative

Capable of handling petabytes of data

Even faster by caching data in-memory

Versatile programming interfaces
Spark: Versatile programming interface

Data visualization is like programming.
  › Point and click doesn’t really cut it
  › Requires an API (grammar): ggplot, matplotlib, bokeh, etc.

Spark has SQL, Scala, Python, Java and (experimental) R API

Libraries for distributed statistics and machine learning
// Query an existing table and get results back as Schema RDD
rdd = hiveContext.sql("select article, text from wikipedia")

// Perform transformations
words = rdd.flatMap(lambda r: r.text.split())

// Sample data and download to driver machine
sampled_words = words.sample(fraction = 0.001).collect()
Reducing interaction latency with Spark

1. In-memory computation
   - Significantly reduces latency

2. High parallelism
   - Get more executors with Mesos or Yarn: a challenge in itself
   - Click a button to increase cluster size in Databricks Cloud
Solutions

1. Interactivity
   - In-memory computation
   - High parallelism

2. Visual medium
   - In-browser collaborative notebooks
   - Summarizing, Sampling and Modeling
Summarize and visualize
Sample and visualize

Sometimes we need to visualize (feel) individual data points

Sampling is extensively used in statistics

Spark offers native support for:

- Approximate and exact sampling
- Approximate and exact stratified sampling

Approximate sampling is faster and is good enough in most cases
Model and visualize

MLLib supports a large (and growing) set of distributed algorithms

- Clustering: k-means
- Classification and regression: LM, DT, NB
- Dimensionality reduction: SVD, PCA
- Collaborative filtering: ALS
- Correlation, hypothesis testing
About Databricks Cloud

Databricks Workspace

- Notebooks
- Dashboards
- Job launcher

Databricks Platform

- Start clusters in seconds
- Dynamically scale up & down
Demo
We saw that

With new big data tools we can resume interactive visual exploration of data

Using Spark we can manipulate large data in seconds
  › Cache data in memory
  › Increase parallelism

To visualize millions of data points we can
  › Summarize
  › Sample
  › Models