MA-INF 4223-Lab Distributed Big Data Analytics

Spark Fundamentals II (Spark ML)

23/05/2017

Dr. Hajira Jabeen, Gezim Sejdiu, Prof. Dr. Jens Lehmann
After completing this lesson, you should be able to:

- Understand the difference between Dense and Sparse Data Types, and how they apply to LabeledPoints
- Have a general understanding of each of the algorithm that will be discussed in the course and how they work.
- *(Hajira is not well! Sorry for missing the lecture)*
A unified analytics stack
Overview

- **MLlib: Machine Learning in Apache Spark**
Spark ML

Spark Core Engine

APIs & Libraries
- Spark SQL & Data Frames
- Spark Streaming
- MLlib
- GraphX

Core

Deploy
- Local: Single JVM
- Cluster: (Standalone, Mesos, YARN)
- Containers: docker-compose
Machine learning are separated in two major types of algorithms:

- Supervised - labeled data in which both, input and output are provided to the algorithm.
- Unsupervised - do not have the outputs in advance.

### Supervised Learning
- **Classification**
  - Naive Bayes
  - SVM
  - Random Decision Forests
- **Regression**
  - Linear
  - Logistic

### Supervised Learning
- **Clustering**
  - K-means
- **Dimensionality reduction**
  - Singular value decomposition (SVD)
  - Principal component analysis (PCA)
MLlib is a standard component of Spark providing machine learning primitives on top of Spark.

- It is scalable machine learning, statistics, math libraries

- Supports out-of-the-box most popular machine learning algorithms like Linear regression, Logistic regression, Decision Trees

- Is available in Scala, Java, Python, and R.
Spark ML-pipelines

- Uniform set of APIs for creating and tuning data processing/machine learning pipelines.

- Core concepts:
  - DataFrame: RDD with names columns. SQL-like syntax and other core RDD operations.
  - Transformer: DataFrame => DataFrame. Eg., features to predictions.
  - Estimator: DataFrame => Transformer. Eg., supervised learning algo.
  - Param: map of params.
  - Pipeline: Chain of Transformers and Estimators. Specifies the data flow.
○ **Estimator**
  - An Estimator abstraction uses an algorithm which is fitted into a DataFrame returning a model.
  - It implements a method `fit()`:
Transformer
- A Transformer is an abstraction which uses an algorithm to transforms one DataFrame to another
- It implements a method `transform()`:
Split text into words => convert numerical features => generate a prediction model.

```scala
val tokenizer = new Tokenizer().setInputCol("text").setOutputCol("words")
val hashingTF = new HashingTF().setNumFeatures(1000).setInputCol(tokenizer.getOutputCol).setOutputCol("features")
val lr = new LogisticRegression().setMaxIter(10).setRegParam(0.01)
val pipeline = new Pipeline().setStages(Array(tokenizer, hashingTF, lr))
val model = pipeline.fit(training.toDF)
val test = sc.parallelize(Seq(
  Document(4L, "spark i j k"),
  Document(5L, "l m n"),
  Document(6L, "mapreduce spark"),
  Document(7L, "apache hadoop")))
val predictions = model.transform(test.toDF)
```
References
