Pre-requisites

Python

Basics of Spark

Day 1

1) What is Bigdata?

2) Hadoop Yarn Architecture

3) An Introduction to Apache Spark

    a) Spark’s Basic Architecture

    b) Spark’s Language APIs

    c) Starting Spark

    d) The SparkSession

    e) Data Frames

        i) Partitions

    f) Transformations

        i) Lazy Evaluation

    g) Actions

    h) Spark UI

2) Spark’s Toolset

3) Structured API

    a) Data Frames and Datasets

    b) Schemas

    c) Overview of Structured Spark Types

        i) Data Frames Versus Datasets

        ii) Columns

        iii) Rows

        iv) Spark Types

d) Overview of Structured API Execution

i) Logical Planning

ii) Physical Planning

iii) Execution

4) Basic Structured Operations

a) Schemas

b) Columns and Expressions

c) Records and Rows

d) Data Frame Transformations

5) Executing on Spark

a) Run job from command line

b) Go through execution plan / DAG on spark for existing model

6) Using logging

Day 2

8) Details of the Case Study which we will use in our hands-on

9) Aggregations

    a) Aggregation Functions

        i) count

        ii) min and max

        iii) sum

        iv) avg

    b) Grouping

    c) Window Functions

    d) User-Defined Aggregation Functions

10) Joins

    a) Join Expressions

    b) Join Types

    c) Inner Joins

    d) Outer Joins

    e) Left Outer Joins

11) Right Outer Joins

12) Data Sources

    a) The Structure of the Data Sources API

    b) CSV Files

    c) JSON Files

    d) ORC Files

    e) AVRO Files

    f) Parquet Files`

    g) Advanced I/O Concepts

        i) Splitable File Types and Compression

        ii) Reading Data in Parallel

Day 3

15) Resilient Distributed Datasets (RDDs)

    a) What Are the Low-Level APIs?

    b) About RDDs

        i) Types of RDDs

        ii) When to Use RDDs?

    c) Creating RDDs

    d) Manipulating RDDs

    e) Transformations

        i) distinct

        ii) filter

        iii) map

        iv) sort

        v) Random Splits

    f) Actions

        i) reduce

        ii) count

        iii) first

        iv) max and min

        v) take

    g) Saving Files

        i) saveAsTextFile

        ii) SequenceFiles

        iii) Hadoop Files

16) Caching

17) Persistence

        i) Different types of persistence

        ii) When to use which kind of persistence.

18) Checkpointing

    i) What is checkpointing?

    ii) localCheckpoint

    iii) When and where to use?

19) Distributed Shared Variables

  a) Broadcast Variables

  b) Accumulators

    i) Basic Example

    ii) Custom Accumulators

Day 4

20) Configuring Applications

21) PySpark UDF Introduction

    a) What is UDF?

    b) Why do we need it?

22) Create PySpark UDF (User Defined Function)

    a) Create a Data Frame

    b) Create a Python function

    c) Convert python function to UDF

23) Using UDF with Data Frame

    a) Using UDF with Data Frame select()

    b) Using UDF with Data Frame withColumn()

    c) Registering UDF & Using it on SQL query

24) Introducing Apache Parquet file format

    a) What is Apache Parquet?

    b) Parquet Format vs. CSV

    c) Advantages of Parquet Columnar Storage

    d) Primitive data types in Parquet format

    e) Apache Parquet Spark Example

        i) Spark Write Data Frame to Parquet file format

        ii) Spark Read Parquet file into Data Frame

        iii) Append to existing Parquet file

        iv) Using SQL queries on Parquet

        v) Spark parquet partition – Improving performance

        vi) Spark Read a specific Parquet partition

25) Apache Arrow in PySpark

a) What is Apache Arrow?

b) Apache PyArrow with Apache Spark

c) What is the problem with existing Pandas/Spark conversion without PyArrow?

d) How to use PyArrow in Spark to optimize?

e) Enabling for Conversion to/from Pandas

f) Pandas UDFs

g) Compatibility Setting for PyArrow

h) Converting Pandas Data frame to Apache Arrow Table

i) PyArrow Table to Pandas Data Frame

j) How does the PyArrow enabled conversion work internally?

Day 5

26) Performance Tuning

   a) Indirect Performance Enhancements

      i) Design Choices

      ii) Object Serialization in RDDs

      iii) Cluster Configurations

      iv) Scheduling

      v) Data at Rest

      vi) Shuffle Configurations

      vii) Memory Pressure and Garbage Collection

   b) Direct Performance Enhancements

      i) Parallelism

      ii) Improved Filtering

      iii) Repartitioning and Coalescing

      iv) User-Defined Functions (UDFs)

      v) Temporary Data Storage (Caching)

      vi) Joins

      vii) Aggregations

      viii) Broadcast Variables

Day 6

29) Machine Learning Basic Concepts

    a) Importing the Libraries

    b) Importing the Dataset

    c) Summary of Object-oriented programming: classes & objects

    d) Missing Data Treatment

    e) Categorical Data

    f) Splitting the Dataset into the Training set and Test set

    g) Feature Scaling

30) Analytics and Machine Learning

    a) What Is Spark’s MLlib?

    b) High-Level MLlib Concepts

    c) MLlib in Action

        i) Feature Engineering with Transformers

        ii) Estimators

        iii) Pipelining Our Workflow

        iv) Training and Evaluation

        v) Persisting and Applying Models

31) Preprocessing and Feature Engineering

    a) Formatting Models According to Use Case

    b) Transformers

    c) Estimators for Preprocessing

        i) Transformer Properties

    d) High-Level Transformers

        i) SQL Transformers

        ii) VectorAssembler

    e) Working with Continuous Features

        i) Bucketing

    ii) Scaling and Normalization

    iii) StandardScaler

f) Working with Categorical Features

    i) StringIndexer

    ii) Converting Indexed Values Back to Text

    iii) Indexing in Vectors

    iv) One-Hot Encoding

g) Feature Manipulation

h) Feature Selection

Day 7

32) Pipeline

   a) Building of pipeline

   b) Saving it

   c) Use on a different dataset

33) Classification

   a) Use Cases

   b) Types of Classification

   c) Classification Models in MLlib

   d) Logistic Regression

      i) Logistic Regression Intuition

      ii) Sigmoid Function

      iii) Model Hyperparameters

      iv) Training Parameters

      v) Prediction Parameters

      vi) Example

   e) Decision Trees

      i) Decision Tree Regression Intuition

      ii) Pruning

      iii) Overfitting in Decision Tree

      iv) Entropy

      v) Information Gain

      vi) Model Hyperparameters

      vii) Training Parameters

      viii) Prediction Parameters

34) Regression

   a) Use Cases

b) Regression Models in MLlib

c) Linear Regression

    i) Simple Linear Regression Intuition

    ii) RMSE

    iii) Model Hyperparameters

    iv) Training Parameters

    v) Example

    vi) Training Summary

d) Decision Trees

    i) Model Hyperparameters

    ii) Training Parameters

    iii) Example