Big Data:
Spark Practical Work
First Semester
2016/2017

Table of Contents

Introduction

The objective of this work is to help students to put into practice the concepts learnt during the theory lessons, and to get proficiency in the use of Spark and other related Big Data technologies. In this exercise, the students are required to develop a Spark application that creates a machine learning model for a real-world problem, using real-world data: Predicting the arrival delay of commercial flights.

The Problem

The basic problem of this exercise is to create a model capable of predicting the arrival delay time of a commercial flight, given a set of parameters known at time of take-off. To do that, students will use publicly available data from commercial USA domestic flights. The main result of this work will be a Spark application, programmed to perform the following tasks:

- Load the input data, previously stored at a known location.
- Select, process and transform the input variables, to prepare them for training the model.
- Perform some basic analysis of each input variable.
- Create a machine learning model that predicts the arrival delay time.
- Validate the created model and provide some measure of its accuracy.

The Data

For this exercise, students will use data published by the US Department of Transportation. This data can be downloaded from the following URL:

http://stat-computing.org/dataexpo/2009/the-data.html

The dataset is divided into several independent files, to make download easier. You do not need to download and use the entire dataset. A small piece should be sufficient, one that fits in your development environment and does not take too long to process. The Spark application you develop, however, should be able to work with any subset of this dataset, and not be limited to a specific piece.

Forbidden variables

The dataset consists of a single table with 29 columns. Some of these columns must not be used, and therefore need to be filtered at the beginning of the analysis. These are:

- ArrTime
- ActualElapsedTime
- AirTime
- TaxiIn
- Diverted
- CarrierDelay
- WeatherDelay
- NASDelay
- SecurityDelay
- LateAircraftDelay

These variables contain information that is unknown at the time the plane takes off and, therefore, cannot be used in the prediction model.

Allowed variables

Any other variable present in the dataset, and not included in the previous list, can be used for the model.

Target variable

The target variable for the prediction model will be **ArrDelay**.

Exercise Requirements

Students have to develop a Spark application capable of loading and processing the input data, and create and validate the prediction model. Each one of these steps are described in detail in the following sections.

Loading the data

Input data files should be placed somewhere accessible to the Spark application, preferably, an HDFS path or Hive table. This initial placement can be done by an external software, like the hadoop command line tool. Once the data is stored, its location must be provided to the Spark application as an input parameter. The application should load the data from the specified location. The application should expect to find the data as it is downloaded from the previous url. No initial pre-processing should be done on the data before feeding it to the Spark application.

Processing the data

After the data is loaded, the application should perform the following operations:

- Selecting the variables that are going to be used.
- Transforming the variables that cannot be used in its raw form.
- Creating new, derived variables that could be helpful to create a better model.

To perform these operations, the students need to analyze and understand each of the variables involved. The dataset public documentation provides a lot of useful information that can help to design these steps. Some situations that may be encountered include:

- Several variables may not contain useful information or are forbidden. These need to be filtered out.
- Several variables may contain information in a way that is difficult to understand/process. These need to be transformed into something meaningful.
- Several variables may provide better information when combined with others. In these cases, new variables could be derived from them.

All these decisions are left to the good judgement of the students. The target variable, ArrDelay, must, of course, be left alone during this process.

Creating the model

After generating the definitive set of variables, the prediction model must be trained, using ArrDelay as the target variable. The students can select any machine learning technique provided by the Spark API they wish to create this model. This selection must be properly justified in the report delivered.

Validating the model

Finally, the model created must be validated, and some measure of its accuracy should be provided. The validation procedure and accuracy metric used is to be decided by the students. As in the previous case, the selection of the evaluation technique and accuracy measure must be sufficiently justified in the report.

Other requirements

Aside from what has already been described, the Spark application developed should meet the following requirements:

- All data processing, model training and validation must be done with Spark.
- No additional machine learning libraries can be used.
- The application must be developed in Scala, Java or Python.

Deliverables

As results of this work, students are required to deliver the Spark application source code and a short report. These two deliverables are now described.

Spark application

The source code for the Spark application must be delivered in a way that is easy to compile and execute. The preferable way is a regular Maven project. If an alternative format is chosen, detailed instructions on how to compile and execute it should be included in the report.

Report

The report should be a short document, describing the work done. No source code should be included in the report.The document must contain the following information:

1. List of variables selected for the model, and the reason to select them.
2. List of variables excluded from the model, and the the reason not to use them.
3. Detailed description and justification of the variable transformations performed.
4. Detailed description and justification of the new variables created.
5. Description of the machine learning technique selected and its parameters, explaining why this technique was selected.
6. Description of the validation process, including justification of the decisions taken.
7. Final evaluation of the prediction model, based on the validation results.
8. Instructions on how to compile and execute the application.
9. Instructions on where to place the input data.
10. Final conclusions and personal remarks.

Packaged deliverable

The work must be delivered in a single ZIP file, containing three elements:

- authors.txt: A plain text file with the names of the authors, one per line.
- report.pdf: A single pdf file with the report document.
- app: A directory containing the application source code (Maven project or equivalent).

The file must be delivered using moodle. No email deliveries will be accepted.

Final remarks

The problem presented in this exercise is the construction of a prediction model. This is a typical Data Science problem, and the application requirements try to cover the basic aspects of most problems of this sort. The focus of the students, however, should be on developing the necessary Big Data processes required to create this model, and not on producing a high-quality predictor. The problem presented (estimating arrival delay in commercial flights) is not a trivial one. The key aspect here is the correct use of the Big Data technologies (mainly Spark, in this case).