SPARK AND PYTHON For BIG DATA

TTT « PERSONAL WORK »

First Name : Sara

Family Name : El-Metwally

Objectives:

You have to choose two end-to-end projects with different levels of difficulty (*Low, Medium or High*). Your goal is to design a data pipeline with Pyspark so that the students know:

- How to setup the cluster environment.
- The data retrieval system (why technology, why and how to setup).
- The storage system of the data (which database, why, and how to setup).
- The data analysis system: Data Cleaning, Feature Engineering**,** Model Training...
- The visualization system (What are the result which need to be displayed, why and how).
- The monitoring system of the cluster (What is important to monitor, why and how).

The students will focus on technology of Spark and Pyspark ecosystem but they could use others stacks for different steps (for instance: visualization and data retrieval).

For each of these two problems, you are asked to give:

1. An abstract
2. A description
3. And your "evaluation" of the problem (see below).
4. You can also give, if necessary, a Jupyter Notebook.

Ps: Don't forget to validate your submissions on Teams. Please use this file for your answers.

Problem 1

1.1 Abstract (minimum 100 characters & maximum 300 characters[1]):

Python Libraries like **Pandas** and **scikit-learn** are suitable for mid-size data processing. Machine learning projects always deal with big data sizes that can not fit into one computer memory. The solution for big data processing is distributing its computation among different computing machine. Each machine will have a running code on subsets of data items and the results will be aggregated at the end in order to deliver a complete solution. **PySpark** is a python API that can handle the parallelization of data processing and provide an easy way to manipulate different issues related to distributing data, code, and collecting results. In this simple example we will explain how to use PySpark and use it for data analysis and processing.

[1] We will not count blank characters on all answers.

1.2 Description[2] (minimum 2000 characters & maximum 3000 characters):

1. First, you need to initialize a **SparkContext** in order to establish a connection with clusters and run any operation.
2. You will need to create **SQLContext** in order to connect the Spark engine with different data sources and allow the Spark SQL commands over these data sources.
3. After setup our working environment, it is time to specify your working data set.
4. It is our data exploration time: Read a CSV file, and tell the Spark to automatically determine the data type by set **inferSchema = true** and print it using **printSchema**
5. In order to adjust some variables data types (i.e. converting integer to float). We will create a function called **convertToFloat** and pass the columns names to it. We will use **withColumn** to inform the Spark which columns need a transformation into a float data type. The columns names are: **age, fnlwgt, capital-gain, educational-num , capital-loss, hours-per-week.**
6. We will explore our data step further in order to gain some insights.
7. The first data pre-processing step is compute the square of age features and adds it as a new column in our data set. From the above **age-income table** , we saw that age and income variables has a non-linear relationship. Young people have low income compared to the mid-age people. Also, the retired people have a fixed retirement low income. We will squared the ages and add its values in a new column called **age_squared** in order to capture this non-linearity feature.
8. Removing a single observation is another pre-processing step since it has no added value to the model.
9. We will create a data processing pipeline using **PySpark** . Your data will input to the pipline from one side to have additional data analysis and transformation and the resulted transformed data will be the output from the other side of the pipline tunnel.

We will create a pipline that has the following stages for each categorical feature (i.e. column) we would like to transform it:

- The **StringIndexer** will used to encode the string values in the columns using their corresponding frequencies numerical indexes.
- The resulted numerical representation of a categorical column will be encoded using **OneHotEncoder** .
- Aggregate the transformations of the encoded columns.

10. To build a LogisticRegression model using PySpark, the input features and the new income label will be passed to the model.

1.3 Your Evaluation of this problem (minimum 500 characters & maximum 1000 characters):

1. Level of difficulty of this problem ? Simple
2. Interest of this problem ? It is a starting step to understand how to use PySpark.
3. Other Remarks ? [...]
4. References ?
    a. https://www.guru99.com/pyspark-tutorial.html#10

[2] You can add pictures of course.

b. https://github.com/SaraEl-Metwally/TOT-ITI-EPITA/blob/main/Big%20Data%20and%20Cloud%20assignment-Final.ipynb

Your Personal Comments

Since I am newbie with a Spark :D, I took this assignment as a first step towards learning PySpark for the first time ever :D. This link (https://www.guru99.com/pyspark-tutorial.html#10) was very useful and help me to understand many concepts. Also, I found some many coding errors there, so my notebook will be useful for others too.