Exercício de Laboratório EBD13

Prof. Lucas Mello Schnorr (INF/UFRGS)

A disciplina EBD13 trata da gerência de dados e computação em nuvem. Para exercitar os conceitos de maneira prática, o exercício de laboratório consiste em criar uma infraestrutura computacional para análise de dados na nuvem, utilizando Apache Spark em modo standalone, e pacotes R tais como *sparklyr*, *dplyr* para enviar as diretivas de processamento de dados. Toda a condução do exercício será realizada nas máquinas *Linux* do laboratório (usuário: *aluno*, senha: *aluno*).

A demonstração em laboratório será conduzida utilizando recursos disponíveis na versão gratuita (free tier) da Amazon EC2. Espera-se que os alunos tenham uma conta na Amazon para conduzir o exercício. Caso negativo, outras nuvens podem ser utilizadas caso o aluno assim desejar.

O portal Amazon EC2 é acessível através do link: *https://aws.amazon.com/pt/ec2/*

1. Instanciar três máquinas virtuais (com 1 core virtual e pelo menos 512 MBytes)
   - Procure no AWS MarketPlace a AMI (imagem): "Debian GNU/Linux 9 (Stretch)"
   - Identificaremos cada uma das máquinas virtuais da seguinte forma
     - Master: máquina onde será executada o processo master do Spark
     - Worker: máquina onde será executada o processo worker do Spark
     - Cliente: máquina onde será executada o cliente (*sparklyr* + *dplyr*)
   - Anote o endereço IP de cada uma das máquinas virtuais
2. Utilizar *ssh* para acessar cada uma das máquinas virtuais
   - Quarde o arquivo *PEM* para isso
   - Instalação Java, execute em *Master* e *Worker*

     ```
     sudo apt install default-jre
     ```

   - Instalação R e dependências, execute no *Cliente*

     ```
     sudo apt install libssl-dev libcurl4-openssl-dev
     sudo apt install r-base
     ```

3. Instalar a versão Apache Spark 2.1.0 em cada uma das máquinas virtuais
   - *https://spark.apache.org/downloads.html*
   - Utilize o comando *wget* para baixar diretamente na nuvem
4. Lançar os seguintes serviços

- Master

```
$HOME/spark-2.1.0-bin-hadoop2.7/sbin/start-master.sh -p 7077
```

  – Obter o endereço do master nos logs, veja em

- Worker

```
$HOME/spark-2.1.0-bin-hadoop2.7/sbin/start-worker.sh ENDERECO
```

  onde *ENDERECO* é algo na forma *spark://IP:PORTA* e representa o endereço do master

5. Na máquina Cliente, conectar com o pacote *sparklyr* na master

```
library(sparklyr)
ENDERECO.MASTER = "spark://IP:PORTA"
sc <- spark_connect(master = ENDERECO.MASTER)
```

   Verificar que a conexão foi bem sucedida

```
connection_is_open(sc)
```

6. Carregar dados na nuvem utilizando

   - Conecte-se com *ssh* na máquina Cliente
   - Execute o seguinte comando para copiar os dados

```
wget http://www.inf.ufrgs.br/~schnorr/acidentes-2016.csv
```

     – Ou utilize dados disponíveis no moodle
     – Ou qualquer outro dado tabular (CSV) disponível

7. Ainda no Cliente, realizar os passos

   Primeiro ler os dados para a variável *df.local*:

   - Ler os dados em R (com *readr*)

```
library(sparklyr)
library(tidyverse)
df.local <- read_delim("acidentes-2016.csv", delim=";");
```

   Em seguida carregá-lo no Apache Spark

   - Assumindo que a variável *sc* contém o resultado da chamada à *spark_connect*()

```
df.remoto <- sdf_copy_to (sc, df.local)
```

   A variável *df.remoto* será uma referência para os dados na nuvem.

8. Realizar a análise dos dados com verbos *dplyr*

   - Use no final *collect()* para trazer os dados para o Cliente