Spark Lightning-fast cluster computing

Here we will download **Spark** binaries on our AWS instance, fire up some **Spark** clusters, and try out **RStudio Server**.

- Spark Documentation - Exploring the Faithful Dataset

Let's get started by connecting to our launching instance on AWS, firing up 1 master and 2 dependent clusters.

- get the **SSH** address and run it on your terminal:

```
ssh -i "udemy.pem" ec2-user@52.33.12.163
```

- cd to folder

```
/home/ec2-user/spark-1.5.1-bin-hadoop2.6/ec2
```

- export both AWS keys into new session

```
export AWS_ACCESS_KEY_ID=xxxxxxxxxxxxxxxxxxxx
export AWS_SECRET_ACCESS_KEY=xxxxxxxxxxxxxxxx
```

- fire up clusters

```
./spark-ec2 -k udemy -i udemy.pem -r us-west-2 -s 1 -t
m1.small launch --copy-aws-credentials my-spark-cluster
```

- log into new master instance from same folder:

```
ssh -i "udemy.pem" root@ec2-54-184-155-154.us-west-
2.compute.amazonaws.com
```

- create RStudio user:

```
sudo adduser ...
sudo passwd ...
```

Once you're up and running enter the following code to get Spark up and running in RStudio

Spark Documentation - Exploring the Faithful Dataset

We'll start with a simple example straight out of the Spark documentation

```
# spark ------------------------------------------------------
--------------

library("SparkR", lib.loc="/root/spark/R/lib")
Sys.setenv(SPARK_HOME="/root/spark")

sc <- sparkR.init()
sqlContext <- sparkRSQL.init(sc)

# faithful ---------------------------------------------------
-----------------

# Create the DataFrame
faithful_spark_df <- createDataFrame(sqlContext, faithful)

# Get basic information about the DataFrame
faithful_spark_df

# Select only the "eruptions" column
head(select(faithful_spark_df, faithful_spark_df$eruptions))

# You can also pass in column name as strings
head(select(faithful_spark_df, "eruptions"))

# Filter the DataFrame to only retain rows with wait times
shorter than 50 mins
head(filter(faithful_spark_df, faithful_spark_df$waiting <
50))

# get the schema
printSchema(faithful_spark_df)
```