How to make XGBoost available in the Spark Notebook

Bogdan Cojocar Follow
Feb 7, 2018 · 2 min read ★

This is a step by step tutorial on how to install XGBoost (an efficient implementation of gradient boosting) on the Spark Notebook (tool for doing Apache Spark and Scala analysis and plotting graphs, similar to the Jupyter notebook).

If you don't have the Spark Notebook installed you can follow this quick guide.

Step 1: Build XGBoost

For this step we need to clone the repository from github and build the project:

```
git clone --recursive https://github.com/dmlc/xgboost
```

Next we need to go into the newly cloned repository and build the project:

```
cd xgboost
make -j4
```

For Mac users you have to do an additional step, before building with `make` :

```
cp make/config.mk ./config.mk
```

Step 2: Create the XGBoost jar

Before we are able to build the jar we need to make sure we have `JAVA_HOME` set up and pointing to the JDK directory. We also need maven installed. Next we need to run maven in the `xgboost` directory and publish the artifact on your local maven repository:

```
mvn install
```

This command will also run some tests, but we can skip that to make the installation faster. If you wish to do that simply run this command instead:

```
mvn -DskipTests install
```

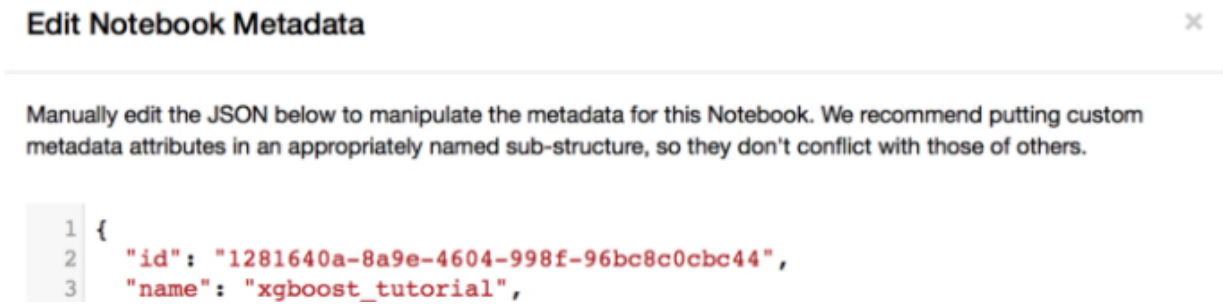
Step 3: Include the XGBoost jar into the Spark Notebook

As we now have the jar available in our local repository we can include it as a dependency in the notebook. In order to do that we need to go to the notebook and open the metadata window in the menu:

```
Edit -> Edit Notebook Metadata
```

A window will open containing a configuration JSON file. We will add the xgboost dependency in the `customDeps` property as shown in the screenshot below:

```
"customDeps": [
   "ml.dmlc:xgboost4j-spark:0.8-SNAPSHOT"
]
```



Please note that at the moment this tutorial was created I was using `xgboost` version 0.8.

The last step is to restart the kernel of the notebook and we should have the dependency available. To restart the kernel go in the menu at the kernel tab and run restart:

```
Kernel -> Restart
```

Now we should be able to import XGBoost:

When we add `import ml.dmlc.xgboost4j.scala.spark` into a cell of the notebook and we run the code (in the menu `Cell -> Run`) it should run succefully.

Xgboost | Machine Learning | Apache Spark | Scala | Big Data

WRITTEN BY

Bogdan Cojocar

Big data consultant. I write about the wonderful world of data.
If you enjoy my articles:
https://www.buymeacoffee.com/paxdata. Cheers!

More From Medium

How to run a real-time pipeline in AWS Kinesis using PySpark Structured Streaming

Bogdan Cojocar in Towards Data Science

Delta lake with Spark: What and Why?

Jyoti Dhiman in Towards Data Science

Predict Customer Churn using PySpark Machine Learning

Marvin Lüthe in Towards Data Science

Logistic Regression with PySpark

Gülcan Öğündür in The Startup

PySpark Under the Hood: RandomSplit() and Sample() Inconsistencies Examined

Meltem Tutar in Udemy Tech Blog

Binary Classifier Evaluation made easy with HandySpark

Daniel Godoy in Towards Data Science

Deploy a Python model (more efficiently) over Spark

Schaun Wheeler in Towards Data Science

Want to Master Your Data? Here's Why You Should Care About Metadata

Nam Nguyen in Towards Data Science