Objective of the Big Data course

This main objective of this course is to give you some brief insight into performing data analysis using Big Data technologies.

Data generation has changed massively in the last decade. The 4 V's of Big Data best describe this change -

- Volume
- Velocity
- Variety
- Veracity

To process this type of data, a new type of infrastructure and software was developed. This type of software processes data using methodologies that are different from the ones you have learnt till here. All of these fall under the broad umbrella of 'big data technologies'. The goals of big data analyses are similar to that of conventional data science, but the methodologies are different.

We are introducing you to some of the important technologies involved in big data analytics. Following are the technologies covered in this course:

- Apache Hadoop
- Apache Hive
- Apache Sqoop
- Apache Spark

The purpose of this module is to get you some hands-on experience with big data technologies.

Working with big data involves a whole lot of engineering activities. However, as an analyst, you will be using big data technologies, and not configuring them. You will be what is commonly referred to as a "consumer" of big data technology. For this reason, we will not be covering the engineering details involved during this course. The engineering aspects will often be handled by a big data engineer or architect.

First Steps

The first module that involves working hands-on in the Big Data Analytics course is the one on Apache Hive. Before you start working on the Hive lab questions, make sure you have familiarised yourself with the lab environment.

The lab comprises of a central dashboard, for which you will receive a username and password. Once on the dashboard, you will find links to take you to all the lab components involved.

Before anything else, try opening all the links present on the dashboard. If any of these don't open, please inform your student mentor.

Some things to keep in mind

- Try finishing large chunks of work in one sitting, as opposed to many sittings with smaller chunks of work. Any active lab will have a timeout feature if you leave it idle for too long. If you are planning to leave your workstation for some time, make sure you reach a checkpoint where you can save your work
- While creating your tables in Hive, make sure you use your own database, don't use the "default" database.

Debugging

Necessity of Debugging

A large part of any coding task is debugging. As data science professionals, you will spend considerable amount of your time figuring out what's wrong with your code.

When you discover an error in any of the sections of the lab, it is a useful skill to learn to look up the errors. Most errors will have some documentation on forums, and this documentation will give you some clue of the source of this error.

Debugging in Big Data technologies

This task is only amplified with Big Data technologies, since there are many more moving parts involved. Learning how to debug situations involved in Big Data technologies is one of the key learning of this course. In this regard, this course is different from the other courses involved. You may end up spending a larger proportion of your time debugging errors, but this is a course feature, not a bug. We want you to get well acquainted with the forums where people take their Big Data errors.

Besides, Big Data workflows and practices are not as well documented as data analysis in R that you learnt through the program. Here are some forums that do have lots of activity and are mostly reliable sources.

- Stack Overflow
- Cloudera Community Forums
- DataBricks Forums

Other than this, there are several Big Data websites and blogs that are updated regularly:

- Big on Data | ZDNet

What are some of the scenarios where debugging will be necessary?

- Spark session won't be initialised

Here is an example of the error message:

```
Spark package found in SPARK_HOME: /usr/local/spark
Launching java with spark-submit command
/usr/local/spark/bin/spark-submit   --driver-memory "1g" sparkr-shell
/tmp/RtmpeYeWCg/backend_port56b355e85f45
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use
setLogLevel(newLevel).
Error in if (len > 0) { : argument is of length zero
```

- Hive queries not running

Here is an example of the error message:

```
hive> show tables;
FAILED: Error in metadata: java.lang.RuntimeException: Unable to
instantiate org.apache.hadoop.hive.metastore.HiveMetaStoreClient
FAILED: Execution Error, return code 1 from
org.apache.hadoop.hive.ql.exec.DDLTask
```

When encountered with any type of errors, it will help to google the first clause of the error. This Stack Overflow link pops up when you Google the error.

While you're working in a role doing data science, you will extensively use forums to discuss issues you might be facing at work.

Queries

How do you go about debugging? In your professional life, debugging will be an integral part of your full-time job. However, during this course, owing to the limited time you have, we will assist you through this debugging process.

- The discussion forum will be regularly updated by our TAs with useful information on how to proceed with the course.
- We will compile the commonly found errors and frequently asked questions, and proactively send the whole batch answers to these questions.

Academic vs technical queries

In general, all the questions you might have about the program will be of two types - academic or technical.

Debugging academic queries

These are queries about the concepts and hands-on practicals that are present in the module.

The best place for resolving your academic queries is the discussion forum. While posting on the discussion forum, it will be helpful if you post some Stack Overflow link along with your question (refer to previous section for more details on this). Please add screenshots of your errors to get quick resolutions.

You can also use the “Report a Mistake” button. However, use this only when you want to report errors. For getting your queries resolved, the discussion forum is definitely the best option.

Debugging technical queries

A technical query is a problem with the regular functioning of the lab environment. Following are some examples of these types of queries:

- The lab page is not loading
- 502 error appearing
- Unable to login to the lab (Web Console, RStudio, etc.)

For these queries, send an email to support@corestack.io and a ticket will be created about your problem. Please add screenshots of your errors to get quick resolutions.

We hope this course gives you a good start into your journey in Big Data technology.