BIG DATA VS ANALYTICS VS DATA SCIENCE: WHAT'S THE DIFFERENCE?

There is much confusion from people who do not work with the technology what the difference is between big data and analytics. Often you see the names **big data analytics**, **big data**, **analytics**, or **data science**. What do these mean?

In brief, **big data is the infrastructure that supports analytics**. **Analytics** is applied mathematics. Analytics is also called **data science**.

That said, you can use big data without using analytics, such as simply a place to store logs or media files. And you can use analytics without a big data database, using, for example, Microsoft Excel.

Analytics

Operations Research is an old word that means to apply mathematics to industrial and other processes to fine tune them. Mathematicians and statisticians prior to, say, the 1990s did this using the tools available to them at that time. That includes Microsoft Excel and statistical packages like SPSS for the PC or SaaS for the mainframe. Before that they used calculators and slide rules with pencil and paper.

The other use of applied mathematics is to evaluate clinical trials in the pharmaceutical industry and research hospitals. Statisticians there are looking to evaluate the efficacy of new drugs. They are also looking to find the correlation between things like smoking the incidence of lung cancer.

Finding the **correlation** between variables, and doing **classification**, is what neural networks and other data structures and other machine algorithms are designed to do.

It is difficult to process medical and industrial data like this for several reasons. First, the statistician/mathematician (data scientist) does know know COBOL, SQL, and other mainframe programming languages and databases. Thus the data scientist has to wait for the programmer to load data into the mainframe. Conversely, the programmer has limited or no knowledge of applied mathematics. It would be better if the data scientist could program this him or herself.

Second, without machine learning SDKs, the programmer would have to write statistical algorithms themselves. Of course, those are built into products like SaaS. But COBOL does not do that unless you program it to do that.

Third, SaaS is proprietary software. So is IBM SPSS. it is not open source. So academics and practitioners cannot contribute their own work to the systems they work with, like scikit-learn, TensorFlow, Keras, Spark Machine Language, and other modern tools. Imagine if you could only use the tools provided by the Microsoft (former), IBM (former), and Oracle (current) monopolies. That would limit their value.

The next difficulty is that in order to put data into a SQL database that data must be put into a specific, rigid row-column format. It does not support **unstructured data**.

Finally, such systems cannot scale almost without limit, as can distributed databases like Spark, ElasticSearch, and Hadoop.

Big Data

Spark, ElasticSearch, Hadoop, etc. are tools that were written mainly to handle the enormous data demands at Yahoo, Google, and Facebook. Those companies wrote that software, with assistance from academics at places like Stanford, and then gave it away as open source software.

These systems can run across a network (cluster) of low cost commodity PCs. They can process more data can even the largest mainframe because you can just add as many additional machines to the cluster as you wish.

PCs are what Amazon EC2, in-house IT shops, and others use in data centers for computers. Those have replaced, for the most part, mainframes, Sun Solarix, IBM AIX, and other large, expensive computers. This had made such systems inexpensive.

Hadoop is not a database. It is a distributed file system, meaning you can use that to store data that spans multiple machines.

This is where the phrase **big data** comes from. Big data is data that will not fit onto the disk drive of a single machine.

As for the limits on memory, Apache Mesos and similar systems abstract memory so that it can be larger than the confines of one machine. To add more memory you simply add another machine. When adding x machine increases memory by an equal amount (x), such a system is said to **scale linearly**.

ElasticSearch and MongoDB are **unstructured big data databases**. This means you can store data in the relatively-free format JSON (JavaScript Object Notation) format. Also you do not have to define the relationship between entities as you would with SQL. In other words, there is no need to define

foreign keys etc.

For example, a MongoDB JSON record could be:

```
{ name:  "Walker", age: "secret"}
```

While the next one could be different, such as having additional fields:

```
{ name:  "Stephen", age: "young", employer: "BMC"}
```

And lots of tools exist to convert data of many different formats to JSON. They same cannot be said for SQL.

Cassandra too is a new idea in computing. This is a **column-oriented database**. Such a database can have missing values. For example, one employee record could include a social security number for Americans but leave it out for foreigners. Such a database does not waste space as does Oracle SQL by reserving space for empty values. It also runs faster by grouping columns with the same value next to each other (rather, alphabetical order).

So that is a basic introduction to the difference between big data and analytics. Know that programmers can specialize in big data programming by being, for example, a big data engineer or architect. But only engineers with knowledge of applied mathematics can do data science.