# # Intro to Langchain

# ## Analysis vs Analytics

Alright! So…  
Let’s discuss the not-so-obvious differences  
between the terms analysis and analytics.  
Due to the similarity of the words, some people  
believe they share the same meaning, and thus  
use them interchangeably. Technically, this  
isn’t correct. There is, in fact, a distinct  
difference between the two. And the reason  
for one often being used instead of the other  
is the lack of a transparent understanding  
of both.  
So, let’s clear this up, shall we?  
First, we will start with analysis.  
Consider the following…  
You have a huge dataset containing data of  
various types. Instead of tackling the entire  
dataset and running the risk of becoming overwhelmed,  
you separate it into easier to digest chunks  
and study them individually and examine how  
they relate to other parts. And that’s analysis  
in a nutshell.  
One important thing to remember, however,  
is that you perform analyses on things that  
have already happened in the past. Such as  
using an analysis to explain how a story ended  
the way it did or how there was a decrease  
in sales last summer.  
All this means that we do analyses to explain  
how and/or why something happened.  
Great!  
Now, this leads us nicely on to the definition  
of analytics.  
As you have probably guessed, analytics generally  
refers to the future. Instead of explaining  
past events it explores potential future ones.  
Analytics is essentially the application of  
logical and computational reasoning to the  
component parts obtained in an analysis. And  
in doing this you are looking for patterns  
and exploring what you could do with them  
in the future.  
Here, analytics branches off into two areas:  
qualitative analytics – this is using your  
intuition and experience in conjunction with  
the analysis to plan your next business move.  
And quantitative analytics – this is applying  
formulas and algorithms to numbers you have  
gathered from your analysis.  
Here are a couple of examples.  
Say, you are an owner of an online clothing  
store. You are ahead of the competition and  
have a great understanding of what your customer's  
needs and wants are. You’ve performed a  
very detailed analysis from women’s clothing  
articles and feel sure about which fashion  
trends to follow. You may use this intuition  
to decide on which styles of clothing to start  
selling. This would be qualitative analytics.  
But you might not know when to introduce the  
new collection. In that case, relying on past  
sales data and user experience data, you could  
predict in which month it would be best to  
do that. This is an example of using quantitative  
analytics.  
Fantastic!  
To backtrack a little, you can combine these  
areas with analyses also – you could perform  
qualitative analysis – to explain how or  
why a story ended the way it did. And you  
can perform quantitative analysis – working  
with past data to explain how sales decreased  
last summer.  
Perfect!  
Now that we have cleared up the differences  
between analysis and analytics it shouldn’t  
be too difficult to see how terms such as  
‘data analysis’, ‘data analytics’,  
‘business analysis’ and ‘business analytics’  
can have their unique meanings too.  
More of this will be explained in the next  
video which aims to simplify these, as well  
as many more with a fantastic diagram. So,  
let’s move on!

# ## Programming Languages & Software Employed in Data Science - All the Tools You Need

Alright! So…  
How are the techniques used in data, business  
intelligence, or predictive analytics applied  
in real life?  
Certainly, with the help of computers.  
You can basically split the relevant tools  
into two categories—programming languages  
and software.  
Knowing a programming language enables you  
to devise programs that can execute specific  
operations. Moreover, you can reuse these  
programs whenever you need to execute the  
same action.  
As you can see from the infographic, R, and  
Python are the two most popular tools across  
all columns. Their biggest advantage is that  
they can manipulate data and are integrated  
within multiple data and data science software  
platforms. They are not just suitable for  
mathematical and statistical computations.  
In other words, R, and Python are adaptable.  
They can solve a wide variety of business  
and data-related problems from beginning to  
the end.  
Of course, R, and Python do have their limitations.  
They are not able to address problems specific  
to some domains. One example is ‘relational  
database management systems’—there, SQL  
is king. It was specifically created for that  
purpose. SQL is at its most advantageous when  
working with traditional, historical data.  
When preparing your BI analysis, for instance,  
you will surely employ it.  
Okay.  
When it comes to data science, mentioning  
MATLAB is inevitable. It is ideal for working  
with mathematical functions or matrix manipulations.  
That’s why it is present in all categories  
except for ‘big data’. While respectable,  
MATLAB usage is a paid service, and that’s  
one of the reasons why it is losing ground  
to open-source languages like R and Python.  
Either way, R, Python, and MATLAB, combined  
with SQL, cover most of the tools used when  
working with traditional data, BI, and conventional  
data science.  
What about big data?  
Apart from R and Python, people working in  
this area are often proficient in other languages  
like Java or Scala. These two have not been  
developed specifically for doing statistical  
analyses, however they turn out to be very  
useful when combining data from multiple sources.  
All right! Let’s finish off with machine  
learning.  
When it comes to machine learning, we often  
deal with big data. Thus, we need a lot of  
computational power, and we can expect people  
to use the languages similar to those in the  
big data column. Apart from R, Python, and  
MATLAB, other, faster languages are used like  
Java, JavaScript, C, C++, and Scala.  
Cool.  
What we said may be wonderful, but that’s  
not all!  
By using one or more programming languages,  
people create application software or, as  
they are sometimes called, software solutions,  
that are adjusted for specific business needs.  
Their smaller scope does not make them less  
useful, in fact, just the opposite—they  
are a lot easier to learn and be adopted by  
others. You have already heard of several  
of those.  
Because of its ability to do relatively complex  
computations and good visualizations quickly,  
Excel is a tool applicable to more than one  
category—traditional data, BI, and Data  
Science. Similarly, SPSS is a very famous  
tool for working with traditional data and  
applying statistical analysis.  
Among the many applications we have plotted,  
we can say there is an increasing amount of  
software designed for working with big data  
such as Apache Hadoop, Apache Hbase, and Mongo  
DB.  
In terms of big data, Hadoop is the name that  
must stick with you. Hadoop is listed as a  
software in the sense that it is a collection  
of programs, but don’t imagine it as a nice-looking  
application. It’s actually a software framework  
which was designed to address the complexity  
of big data and its computational intensity.  
Most notably, Hadoop distributes the computational  
tasks on multiple computers which is basically  
the way to handle big data nowadays.  
Power BI, SaS, Qlik, and especially Tableau  
are top-notch examples of software designed  
for business intelligence visualizations.  
In terms of predictive analytics, EViews is  
mostly used for working with econometric time-series  
models, and Stata—for academic statistical  
and econometric research, where techniques  
like regression, cluster, and factor analysis  
are constantly applied.  
As a final note, remember the following.  
Should you have the relevant business and  
theoretical knowledge, learning a software  
tool is relatively easy as opposed to learning  
a programming language. More importantly,  
it will be sufficient for your need to create  
quick and accurate analyses.  
However, if your theoretical preparation is  
strong enough, you will find yourself restricted  
by software. Knowing a programming language  
such as R and Python, gives you the freedom  
to create specific, ad-hoc tools for each  
project you are working on.  
Great!  
We hope we gave you a good idea about the  
level of applicability of the most frequently  
used programming and software tools in the  
field of data science.  
Thank you for watching!