

# WhiteHat Session 3 - Tools of the Trade

6 Hours

## Overview

In this module we will be covering various tools apprentices will need to enhance their analytics performance in their roles. In the first half of the session we will give an overview of Big Data, considering how it works and what the benefits and drawbacks are of using it. This is followed by a discussion around data analytics platforms and comparing their usage to coding it yourself.

In the second part of the module we will be looking into the statistical programming language R, briefly looking at how many of the processes we learned in Python can be performed in R.

## Prerequisites

- Apprentices will need to have R installed on their computer ([link](#))
- They should also install RStudio ([link](#))

## Learning objectives

- Understand fundamental concepts of **Big Data**
- Justify the use (or lack of) Big Data technologies **in your analysis**
- Critically evaluate the differences in **using a data platform** and **coding it yourself**
- Understand how **R** can be utilised in your analysis

## Assignment

### Task

Justify the use (or lack of) Big Data technologies in your role

### What we are looking for

A written report (max 1500 words) detailing how Big Data technologies are used in your role and the business impact this decision has. Consider also how Big Data technology could be used in your role and what the potential business implications would be. Additionally, you should consider how use of these technologies comply with GDPR and other policies your workplace has around data protection.

## ***Technical Knowledge***

- Identify, collect and migrate data to/from a range of internal and external systems (TC1)
- Interpret and apply the organisation's data and information security standards, policies and procedures to data management activities (TC3)
- Perform routine statistical analyses and ad-hoc queries (TC6)
- Use a range of analytical techniques such as data mining, time series forecasting and modelling techniques to identify and predict trends and patterns in data (TC7)
- Apply the tools and techniques for data analysis, data visualisation and presentation (TC8)
- Works with organisation's data architecture (TC11)
- Assist with data quality checking and cleansing (TC12)

## ***Skills, Attributes and Behaviours***

- Logical and creative thinking skills (SAB1)
- A thorough and organised approach (SAB5)
- Maintain productive, professional and secure working environment (SAB8)

## ***KM1 Data Analysis Tools Syllabus***

- Explain the nature and challenges of data volumes being processed through integration activities and how a programming approach can improve this:
  - Big Data (1.5)
- Capabilities and functions of statistical programming language: R (2.1)

## ***Recordings (for coach reference)***

<https://drive.google.com/drive/folders/1uXw2RfTfzVR1QEMRoAiX6hvWiZn4xzR8?usp=sharing>

Session Overview	Timing (approximate)
<a href="#">Class Introduction</a>	10 Minutes
<a href="#">Understanding Big Data</a>	20 Minutes
<a href="#">The Five V's</a>	25 Minutes
<a href="#">Developing the Future</a>	20 Minutes
<a href="#">How Does it Work?</a>	10 Minutes

<i>Break</i>	10 Minutes
<a href="#"><u>Using Big Data in Your Role</u></a>	10 Minutes
<a href="#"><u>Advantages and Disadvantages</u></a>	20 Minutes
<a href="#"><u>Big Data Products</u></a>	5 Minutes
<a href="#"><u>Data Platforms</u></a>	20 Minutes
<a href="#"><u>Platform vs Coding Yourself</u></a>	20 Minutes
<a href="#"><u>Setting up R</u></a>	5 Minutes
<a href="#"><u>Session 1 Recap</u></a>	5 Minutes
<a href="#"><u>Session 2 Intro</u></a>	10 Minutes
<a href="#"><u>Features of R</u></a>	30 Minutes
<a href="#"><u>Control Flow in R</u></a>	30 Minutes
<a href="#"><u>EDA in R</u></a>	35 Minutes
<i>Break</i>	15 Minutes
<a href="#"><u>Visualisation in R</u></a>	35 Minutes
<a href="#"><u>Linear Modelling in R</u></a>	15 Minutes
<a href="#"><u>RStudio</u></a>	5 Minutes
<a href="#"><u>Session 2 Recap</u></a>	5 Minutes

## *Additional Resources*

- [Basic R Tutorial](#)

Topic	Class Introduction	Duration	10 minutes
-------	--------------------	----------	------------

Objectives
● To provide an overview of the class agenda and the expected learning objectives

Section No	Section	Notes	Timing
1	<b>Session 1: Tools of the Trade</b> <p>Data Analytics is taking over the world and many companies are taking the opportunity to develop products and software to help companies leverage their data more efficiently without having to upskill their staff. In this session we will be learning about the fundamentals of Big Data and how it can impact your analysis. We will also consider the various data tools available to help you analyse them in your portfolio.</p> <p><b>Session Outline</b></p> <ul style="list-style-type: none"> <li>• Understanding Big Data</li> <li>• The Data Stack</li> <li>• Data Science The Future</li> <li>• How It Works</li> <li>• An Overview of Data Architectures</li> <li>• Big Data Products</li> <li>• Data Platforms</li> <li>• Platform vs Coding Yourself</li> <li>• Setting Up A Big Data Environment</li> <li>• Basics</li> </ul> <p><b>Learning Objectives</b></p> <ul style="list-style-type: none"> <li>• Understand fundamental concepts of Big Data</li> <li>• Explain the use of various Big Data technologies in your analysis</li> <li>• Critically analyse the differences in using a data platform and coding it yourself</li> </ul>	Coach welcome to session, run an ice breaker from <a href="#">here</a> . Run through session plan and learning objectives	

Topic	Understanding Big Data	Duration	20 Minutes
-------	------------------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>• Define what we mean by 'Big Data'</li> </ul>

Section No	Section	Notes	Timing
1	 <p>The first thing we will be covering today is the fundamentals of Big Data and why it is important to you and your organisation.</p>		
2	<h2>What is Big Data?</h2> <p>Ask apprentices what their understanding of Big Data is (through annotate, chat, etc).</p> <p>Some follow up questions:</p> <ul style="list-style-type: none"> <li>• How would you use it in a working environment?</li> <li>• What type of data do we mean by 'Big Data'? (i.e. structured/unstructured/quantitative/qualitative etc)</li> </ul>		

		<ul style="list-style-type: none"> <li>• Does your organisation have a Big Data strategy?</li> </ul>	
3	<p><b>Big Data is data whose scale, distribution, diversity and or timeliness require the use of new technical architectures and analytics to enable insights that unlock new sources of business value.</b></p> <p><small>McKinsey &amp; Co., Big Data: The Next Frontier for Innovation, Competition, and Productivity</small></p>	<p>McKinsey's definition of Big Data implies that organisations will need new data architectures, new tools, methods and an integration of multiple skills to be competitive when it comes to recruiting talent and generating business insights.</p> <p>Data is created constantly and at an ever-increasing rate. Mobile phones, social media, imaging technologies- all these and more create new data and that must be stored somewhere.</p> <p>Merely keeping up with this huge influx of data is difficult, but what is even more challenging is analysing the vast amounts of it, especially when it does not conform to traditional notions of data structure. This makes identifying meaningful patterns and extracting useful information harder.</p>	
4	 <p><small>Big Data is data but in huge volume that is also growing exponentially with time. There is no (real) limit to how large a 'Big Data' set can be, except from storage. There is no standard definition of how large a data set must be for it to be defined as 'big data', but typically it is so large and complex that none of the traditional data management tools are able to store or process it efficiently.</small></p>	<p>'Big Data' is data but in huge volume, but also growing exponentially with time. Like the way the universe is continually expanding there is no (real) limit to how large a 'big data' set can be. Except from storage...</p> <p>There is no standard definition of how large a data set must be for it to be defined as 'big data', but typically it is so voluminous that traditional data processing software (tableau, excel, your laptop) can't manage it. Such data is so large and complex</p>	

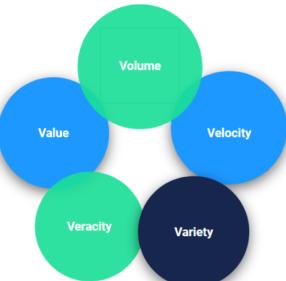
		that none of the traditional data management tools are able to store or process it efficiently.	
5	 <p><small>Big Data is not just about size or scope but also the technology that supports it. Until recently analysing Big Data sets was impossible as the tools had not been invented!</small></p>	<p>Therefore, Big Data as a topic is not only about the size and scope of the data, but the technology that supports it. Until recently, analysing big data sets was impossible- the tools literally hadn't been invented!</p> <p>Let's look at some examples of Big Data. How many tweets do you think are sent every day?</p>	
6	 <p>Generates 500 million Tweets <b>every day</b></p> <p><small>Source: Internet Live Stats</small></p>	<p>Just think about the amount of data that is! Every character, link, image is data which Twitter is processing.</p> <p>Click on the source link to get live statistics</p>	
7	 <p>Generates 1 TeraByte of trading data <b>every day</b></p> <p><small>Source: New York Stock Exchange</small></p>	<p>Consider the NY stock exchange and the sheer volume of stocks and transactions that are getting processed. All this data needs to be stored somewhere!</p>	
8	 <p>Generates 4 petabytes of user data <b>every day</b></p> <p><small>Source: Facebook</small></p>	<p>Each of these companies generate an enormous amount of data every day. These examples show not just the size of the big data but the continuous growth of data that it represents.</p> <p>Question: What other kinds of data sources can you think of that generates data at this scale? [Could say email (est. 168 million per second), sms messages (est 11 million per second) or google search results (est 698445 per second) for example].</p>	

		Current estimates are that 1820 TB of new data are created every second over the planet.	
9	<p><b>79% of enterprise executives agree that companies that do not embrace Big Data will lose their competitive position and face extinction</b></p> <p style="font-size: small; color: #888;">Source: <a href="#">Accenture</a></p>	<p>As Data Analysts it's important we are able to properly understand what Big Data means. Particularly because there are big opportunities for those companies that embrace these new tools, analytical methods and data architectures.</p> <p>Why should we care? As data analysts we have a lot more data sets to use to inform decisions in our companies. This shouldn't be surprising, think about all the devices in our homes today that gather data (Alexa, FitBit, CCTV, etc that produce 5 quintillion bytes of data daily worldwide).</p>	

Topic	The Five V's	Duration	25 Minutes
-------	--------------	----------	------------

Objectives	
<ul style="list-style-type: none"> <li>• Understand the characteristics that make up big data</li> </ul>	

Section No	Section	Notes	Timing
1	 <p>The Five V's</p> <p><small>Big Data Aim: Understand the characteristics that make up Big Data</small></p>	<p>Now we know what we mean by 'big data' we are going to look at five characteristics we need to be aware of.</p> <p>For data analysts it is important to understand the nature of a data set being considered for</p>	

		<p>inclusion as part of their analysis.</p>	
2		<p>The 5 v's are volume, velocity, variety, veracity and value. Don't worry about memorising these right now, we'll go through each one in detail. The 5 V's provide a useful toolbox of characteristics that can be deployed to assess each data set, determine any risks and create parameters for how to use the data. You should use the 5 V's as a mental checklist to employ whenever you are given a big dataset to work with.</p>	
3	<p>Volume</p>  <ul style="list-style-type: none"> <li>• Relates to how much data there is</li> <li>• Whether a dataset is considered 'big' or not is dependent on the volume</li> <li>• Data is considered Big when it is too voluminous for traditional data storage methods</li> <li>• Big Data platforms provide a way to store incredibly large amounts of data as well the ability to process it efficiently</li> </ul>	<p>The first characteristic to consider is volume, or how much data is in the dataset.</p> <p>Size of data plays a very crucial role in determining value you can get out of the data.</p> <p>Whether a particular data set can actually be considered as Big Data or not is dependent on the volume of the data.</p> <p>Typically petabytes (1000 terabytes) or exabytes (1000 petabytes) of data consisting of billions to trillions of records of millions of people- all from different sources (e.g. Web, social media, mobile data and so on).</p> <p>What do we think some issues that may arise due to the volume of our data?</p>	

4	<p><b>Velocity</b></p>  <ul style="list-style-type: none"> <li>• Refers to the speed of data generation</li> <li>• For example, how fast data flows in from sources such as application logs, networks and social media sites</li> <li>• Typically the flow will be massive and continuous</li> <li>• Increasingly companies are looking to stream analytics and data as it becomes available at the right time to make appropriate business decisions</li> <li>• Big Data platforms offer solutions to dealing with fast flowing data in terms of storage and processing</li> </ul>	<p>Refers to the speed of generation of data</p> <p>Velocity deals with the speed at which data flows in from sources like business processes, application logs, networks and social media sites, for example</p> <p>Typically for Big Data the flow of data is massive and continuous</p> <p>Increasingly companies are looking to stream analytics and data</p> <p>The data has to be available at the right time to make appropriate business decisions. The more agile a business the more the need for streaming.</p> <p>What other benefits or drawbacks can we think of to do with the velocity of data?</p>	
5	<p><b>Variety</b></p>  <ul style="list-style-type: none"> <li>• Refers to the heterogeneity of the data source</li> <li>• In other words, how different the data sources are (unstructured vs structured etc)</li> <li>• With large flows of data from several of sources it is possible that there will be a variety of data types and structures</li> <li>• Big Data platforms can handle a variety of sources, structures and data types</li> </ul>	<p>Variety refers to data types.</p> <p>What are some of the different data types we saw back in module 1?</p> <p>Heterogeneity (how different it is) of data sources is important to consider with big data. For example, we need to consider whether it is structured or unstructured or what the file types are.</p> <p>Different types of data require different strategies which you will need to consider as part of your analysis.</p>	

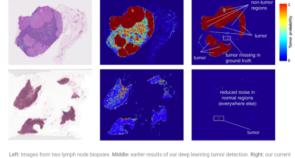
6	<p><b>Veracity</b></p>  <ul style="list-style-type: none"> <li>• Refers to the quality of the data</li> <li>• Checking the veracity of data sources is always challenging- never trust data as given</li> <li>• Enquire about the data preparation- has it been normalised already? Has it been subject to statistical manipulation (i.e. outliers removed)?</li> <li>• Where possible you should always ask to look at raw data</li> <li>• Determining veracity can be challenging to an organisations (politics and policies) as well as externally (determining authorship/ownership)</li> </ul>	<p>Veracity is the quality of being true or the habit of telling the truth</p> <p>Veracity of sources is always challenging</p> <p><b>Rule of thumb: never trust data as given</b></p> <p>Enquire and ask about the data preparation. Has it been normalised already? Been subject to statistical manipulation (i.e. outliers removed)?</p> <p>Where possible, you should always ask to look at the raw data</p> <p>Determining veracity can be challenging to an organisations (politics and policies) as well as externally (determining authorship/ownership)</p> <p>This is especially true of big data where data potentially could have drawn from a variety of disparate sources.</p>	
7	<p><b>Value</b></p>  <ul style="list-style-type: none"> <li>• Refers to the business value of the data</li> <li>• Having access to vast amounts of fast flowing data is useless unless it can be leveraged into something of value</li> <li>• How does this data add value to your business? What is the ROI?</li> </ul>	<p>It's all well and good to have access to big data, but unless we can turn it into something valuable it is useless</p> <p>Does it provide added value to your business?</p> <p>Is the organisation working on big data achieving a high return on investment?</p> <p>If working with Big Data is not adding to a businesses profits, it is useless.</p> <p>Where do you think utilising Big Data will add value to your business?</p>	

8	<p><b>Activity</b></p> <ul style="list-style-type: none"> <li>• Think about some of the data you use regularly, how does it stack up against the 5 Vs?</li> <li>• Share your thoughts with your group and be prepared to feedback to the rest of the class</li> </ul>	<p>In your groups look for some datasets you are familiar with (could be from work if closed cohort, or something they used for bootcamp, hackathon, presentation) and discuss the five v's.</p> <p>I.e. What is the value in analysis? How can we check the veracity of the data? What is the volume or velocity?</p>	10 Minutes
---	---	--	------------

Topic	Developing the Future	Duration	20 Minutes
-------	-----------------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>● Understand the potential Big Data Analysis can have on our industries</li> </ul>

Section No	Section	Notes	Timing
1	 <p><b>Developing The Future</b></p> <p><small>Top Aim: Understand the potential Big Data Analysis can have on our industries</small></p>	<p>Big Data as a subject area is still being invested in heavily and is at the forefront of some of the most recent developments in computing.</p> <p>Social media and genetic sequencing are among the fastest growing sources of big data and examples of non traditional sources of data being used for analyses.</p> <p>How do you think machine learning helps diagnoses in medicine?</p>	

2	<p><b>Medicine</b></p>  <p>Left: Images from two breast node biopsies. Middle: earlier results of our deep learning tumor detection. Right: our current results. Notice the visibly reduced noise (artificial noise that can corrupt the input images).</p> <p>Genetic sequencing and human genome mapping provide a detailed understanding of genetic makeup and lineage. The healthcare industry is looking toward these advances to help predict which illnesses a person is likely to get in their lifetime and take steps to avoid these maladies or reduce their impact through the use of personalised medicine and treatment.</p> <p>While data has grown, the cost to perform this work has fallen dramatically. The cost to sequence one human genome fell from \$100m in 2001 to \$10k in 2011. Today it costs around \$1k.</p> <p>Companies are able to leverage the vast amount of new data available to build machine learning tools to predict results based on past experience.</p> <p>For example, a 2017 report showed it was possible to train a model to detect tumours in breast cancer patients that either matched or exceeded the performance of a pathologist who had unlimited time to examine the slides.</p>		
3	<p><b>NLP</b></p>  <p>Natural Language Processing is the ability of a computer to understand human language as spoken. The most obvious examples that people can relate to these days are Google Home and Amazon Alexa. Both use NLP and other technologies to give us a virtual assistant experience.</p>	<p>10 Minutes</p>	

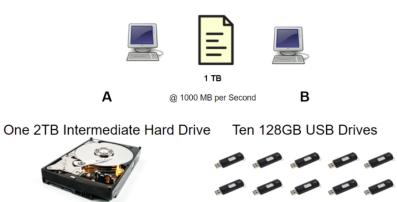
		<p>Activity- In groups discuss and sketch how you think Alexa produces personalised recommendations for you (e.g. music, restaurants, movies, etc)</p>	
4	<p><b>Alexa</b></p> <p>The diagram shows the Alexa processing flow. Step 1: A user says 'Alexa, turn on the kitchen light'. Step 2: The Alexa Service (cloud icon) receives the command and sends a 'Directive: TurnOnRequest' to a 'Smart Home Application'. Step 3: The application controls a 'Kitchen light' via a 'Device cloud' and the 'Internet'. A note at the bottom states: 'Your words are recorded. The recording is sent to Amazon's server to be analysed (due to interpreting sounds taking up a lot of computational power). Important words are identified and corresponding functions are carried out (e.g. "basketball" would open a sports app). The server then sends the information back to your device. If Alexa needs to say something it will follow a reverse process (identify the words it wants to give and convert them to the correct sounds)'.</p>	<p>Alexa is built on natural language processing, a process of converting speech into words, sounds and ideas</p> <p>Alexa will record your words. Interpreting sounds takes up a lot of computational power however, so the recording is sent to Amazon's servers to be analysed more efficiently (use of big data)</p> <p>There, the recording will be broken down into individual sounds. It then consults a database containing various word's pronunciations to find which words most closely correspond to the combinations of individual sounds.</p> <p>It then identifies important words to make sense of the tasks and carry out corresponding functions. For example, if Alexa 'hears' basketball, it would open a sports app</p> <p>Amazon's servers then send the information back to your device. Alexa may speak at this point. If Alexa needs to say something it will go through the same process, but in reverse (i.e. identify the words it wants to give, convert them into audio files and send them back).</p>	

5		<p>For the fun of it, I have included a video of Alexa in action...</p> <p>What other uses of big data are you aware of?</p>	
---	---	--	--

Topic	How Does it Work?	Duration	10 Minutes
-------	-------------------	----------	------------

Objectives	
<ul style="list-style-type: none"> <li>Understand the principles behind how Big Data technologies work</li> </ul>	

Section No	Section	Notes	Timing
1	 <p><b>How It Works</b></p> <p><small>3m Aim: Appreciate how Big Data makes data analysis more efficient</small></p>	<p>Let's now consider how Big Data technologies work.</p> <p>Take for example a SQL query or a python script which you can run on your local computer. How efficiently do you think it will process data which is in terabytes?</p> <p>This is where Big Data comes in. Instead of downloading all the data to your machine, you instead take your code to the data.</p>	

2	<p><b>Divide and Conquer</b></p>  <p>Most Big Data software use an approach called 'Divide and Conquer' where the data will be split into smaller 'chunks' and processed simultaneously across different nodes and the results combined. This process happens over several servers so if one fails the analysis can continue and the remaining can pick up the slack.</p>	<p>Big Data uses an approach called 'divide and conquer' where your data is split into smaller chunks and processed simultaneously, after which the results are combined.</p> <p>To do this efficiently it will place these chunks over several servers.</p> <p>This has a couple of benefits-by spreading out the processing across multiple servers you are requiring each one individually to do less processing which overall is faster. Also, by spreading the chunks about it means if one server fails for any reason the analysis can continue uninterrupted on the others.</p> <p>Once complete each server has completed their part and the results combined, the final output is then sent back to you.</p> <p>Let's do an example to show why it is quicker.</p>	
3	<p><b>Which is Faster?</b></p>  <p>A @ 1000 MB per Second B</p> <p>One 2TB Intermediate Hard Drive Ten 128GB USB Drives</p>	<p>Let's do some maths! [This can be done in breakout rooms]</p> <p>Imagine you want to transfer 1TB (1000GB) of data between two computers.</p> <p>Assuming the transfer speed is 1000 MB per second, how long will it take to transfer the data using:</p> <ol style="list-style-type: none"> <li>One 2TB intermediate hard-drive or</li> <li>Ten 128GB USB Flash Drives?</li> </ol> <p>Assume both computers have 10 USB ports and transferring into multiple USB drives</p>	5 minutes

		simultaneously does not affect speed.	
4	<p><b>Hard Drive</b></p>  <ul style="list-style-type: none"> <li>Moving 10000MB @ 100MB per second would take 1000 seconds (or 16.7 minutes)</li> <li>Giving time for downloading and uploading, total time would be approximately 34 minutes</li> </ul> <p><b>Ten USB sticks</b></p>  <ul style="list-style-type: none"> <li>Dividing the process between 10 USB sticks means each has to transfer 10GB (or 1000MB) each @100MB per second this will take 100 seconds (or 1.7 minutes)</li> <li>Given each drive is downloading and uploading simultaneously, this would take 3.4 minutes approximately</li> </ul>	<p>You can see then that the divide and conquer approach is much faster than doing it all in one go.</p> <p>This is the basic principle behind Big Data. Divide your analyses into different chunks and process each simultaneously. It means processing terabytes of data can happen in minutes instead of hours.</p>	

Topic	Using Big Data in Your Role	Duration	10 Minutes
-------	-----------------------------	----------	------------

Objectives	
	<ul style="list-style-type: none"> <li>Consider how Big Data technologies can be utilised in apprentices own roles</li> </ul>

Section No	Section	Notes	Timing
1	 <h2>Using Big Data In Your Role</h2> <p>30s Aim: Consider how Big Data can be utilised in your role</p>	Over to you, let's now consider how Big Data could be used to enhance your role.	
2	<p><b>Activity</b></p> <p>In breakout rooms brainstorm how you think Big Data technology could be utilised in your role or in your company. When back annotate the screen with your ideas.</p>	<p>We are going to go into breakout rooms where I would like you to discuss how Big Data could be used to enhance your roles</p> <p>Once back, ask apprentices to annotate the slide to show their</p>	7 Minutes (5 Minutes in room and 2 to annotate)

		<p>ideas.</p> <p>When writing up a project for your portfolio, a good thing to include is your justification (or lack of) big data technologies in your analysis. We will look into this in the following section.</p>	
--	--	--	--

Topic	Advantages (and Disadvantages)	Duration	20 Minutes
-------	--------------------------------	----------	------------

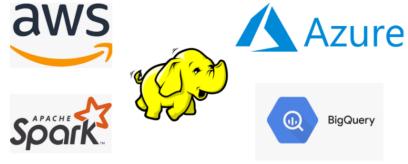
Objectives
<ul style="list-style-type: none"> <li>Justify the use of Big Data in analysis by considering the benefits and drawbacks</li> </ul>

Section No	Section	Notes	Timing
1	<p><b>Advantages and Disadvantages</b></p> <p><small>Topic: Justify the use of Big Data technologies in your analysis by considering the benefits and drawbacks</small></p>	<p>So what are the benefits of utilising big data in your analysis?</p> <p>When justifying your use or non use you will want to refer to some of the information provided in this section.</p>	
2	<p><b>Advantages</b></p> <p><small>• Ability to access and process large volumes of data quickly • Analysis run on larger datasets will be more reliable, representative and accurate as using samples from data will not be necessary • Faster analysis leads to a reduction in human utilisation, meaning more projects can be undertaken in better utilisation of analysis • Projects finished in shorter time frames can lead to increased reputation amongst customers and products being brought to market</small></p>	<p>Run through the advantages, can the apprentices think of any others?</p>	
3	<p><b>Disadvantages</b></p> <p><small>• It is expensive • Big Data technologies only provide a small amount of free processing before charging and costs can rack up quickly • Therefore you do not have time to play around or make mistakes- time is money • Big Data still requires a degree of competency from the user. It is something that can just be picked up by anybody • The nature of Big Data means you are exporting data externally, potentially bringing in cybersecurity risks • It can be challenging to integrate Big Data output into your system</small></p>	<p>Run through the disadvantages, can the apprentices think of any others?</p> <p>In most cases the advantages of using Big Data will outweigh the disadvantages, but it is important to be aware of the downsides when writing up</p>	

		your portfolio.	
4	<p><b>Activity</b></p> <p>In breakout rooms read and discuss this article.</p> <ul style="list-style-type: none"> <li>• Do you agree with what has been written?</li> <li>• How do you think Big Data will personally affect your role?</li> <li>• Can you find other articles which support or argue against the use of Big Data technologies?</li> </ul>	<p>In breakout rooms read and discuss the following article-  <a href="https://www.datamation.com/big-data/big-data-pros-and-cons.html">https://www.datamation.com/big-data/big-data-pros-and-cons.html</a></p> <p>Do you agree with what the writer has put? Can you find other articles which support or argue against the use of big data? What do you think and how can it personally affect your role?</p>	10 minutes

Topic	Big Data Products	Duration	5 Minutes
-------	-------------------	----------	-----------

Objectives
<ul style="list-style-type: none"> <li>● Be aware of the various Big Data technologies that exist</li> </ul>

Section No	Section	Notes	Timing
1	 <p><b>Big Data Products</b></p> <p><small>30s Aim: Have an awareness of the various Big Data technologies that exist</small></p>	<p>Finally, let's look at what Big Data products are available.</p> <p>We will not look into any of them individually, but it is important you know what exists if you want to try them out.</p>	
2		<p>There are many platforms or services that allow for the efficient processing of Big Data. Hadoop was one of the first built and many companies still use it today to process their data. Apache Spark provides a similar service and both can be easily integrated with python.</p>	

		<p>BigQuery is run by Google while AWS is run by Amazon, both have a wide suite of functions allowing you to run queries and scripts efficiently. AWS even has functionality to bring python scripts live. Azure is owned by Microsoft and offers a cloud computing platform with a wide degree of functionality</p> <p>Many more exist, what does your company use?</p>	
--	--	--	--

Topic	Data Platforms	Duration	20 Minutes
-------	----------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>Consider the various data platforms that exist</li> <li>Reflect on what products you use and justify their use</li> </ul>

Section No	Section	Notes	Timing
1	<p><b>Data Platforms</b></p> <p>Ideas Reflect on the products you use in your role and justify their use.</p>	<p>An aspect of data analytics you need to consider and justify is the use of data platforms.</p> <p>Companies invest in platforms to make analysis faster and easier to perform. For example, instead of building a dashboard in python you could use Tableau or PowerBI.</p> <p>In this section we will consider what platforms there are and discuss their potential uses and how you can apply them in your role.</p>	

2	<p><b>Visualisation Platforms</b></p> 	<p>Already in this course we have come across Tableau and PowerBI. These platforms allow us to build visualisations in a simple way and have a varied suite of options as well as other analysis functions.</p> <p>Ask the apprentices to give any examples of how they have used these softwares in their role.</p>	
	<p><b>Data Management</b></p> 	<p>Many of you are familiar with software like Microsoft Excel for spreadsheet and database management. It is easy to perform EDA, calculating aggregates and making visualisations as well as producing tables for showcasing your results. The GUI is fairly intuitive and macros allow for a great range of functionality including adding regression and other ML techniques)</p>	
3	<p><b>Statistics Platforms</b></p> 	<p>Has anyone used SPSS before?</p> <p>SPSS stands for statistical package for social sciences and allows you to perform a wide variety of statistical operations on a dataset, including linear and logistic regression, ANOVAs and much more.</p> <p>While it is expensive to license and has a steep learning curve, you can perform statistical analysis incredibly quickly through a few clicks of a button once you know what to do.</p>	

4	<p><b>Cloud Based Platforms</b></p> 	<p>Then you have software like Microsoft Azure. Azure is a cloud based computing platform which allows you to perform data analytics and is used to supplement companies servers (or even replace them).</p> <p>Does anyone have any experience of using Azure?</p>	
5	<p><b>Common Platforms</b></p> 	<p>There are a wide range of data platforms on the market which allow you to do anything from data mining and data retrieval to exploratory data analysis and predictive analytics.</p> <p>Ask the apprentices what softwares they use and give examples of how they have used them.</p>	
6	<p><b>Activity</b></p> <p>In Breakout Rooms:</p> <ul style="list-style-type: none"> <li>• Discuss what software platforms you use in your role and how you use them</li> <li>• What are the advantages and disadvantages of using them?</li> </ul>	<p>In breakout rooms, discuss what softwares you use in your role and what advantages (and disadvantages are of using them).</p> <p>Ask for examples once everyone has returned.</p>	8 Minutes

Topic	Platforms vs Coding it Yourself	Duration	20 Minutes
-------	---------------------------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>● Justify the use of platforms by comparing them to coding it yourself</li> </ul>

Section No	Section	Notes	Timing

1	 <p><b>Platforms vs Coding Yourself</b></p> <p><small>Aim: Justify the use of platforms by comparing them to your own coding</small></p>	<p>We have seen various platforms and softwares you can use in your role if your company has licensed them, but are they the optimal way for your analytics?</p> <p>All platforms depend on pre-written code, but could it be better to just code it yourself?</p> <p>When writing your portfolio you need to be able to justify why you used (or didn't use) certain programmes and part of this is considering the advantages and disadvantages of coding it yourself.</p> <p>The following slides will list some of the advantages and disadvantages of each you can use in your write ups.</p>			
2	<p><b>Advantages</b></p> <table border="0"> <tr> <td style="vertical-align: top; width: 50%;"> <p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Easy to use and do not require technical knowledge or skills</li> <li>• More agile as 'drag and drop' options allow for quicker building of visualisations</li> <li>• Greater agility leads to lower costs as less time is spent on development</li> <li>• Relatively simple to edit what you have built</li> <li>• Often tech support available</li> </ul> </td> <td style="vertical-align: top; width: 50%;"> <p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize colour scheme and functionality in almost any way imaginable</li> <li>• Once you learn one language, you can learn others as they are all interconnected</li> <li>• Common coding languages (R, Python, etc) are free to install and use</li> <li>• You own the source code</li> <li>• Most languages have active communities where you can ask for help</li> </ul> </td> </tr> </table>	<p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Easy to use and do not require technical knowledge or skills</li> <li>• More agile as 'drag and drop' options allow for quicker building of visualisations</li> <li>• Greater agility leads to lower costs as less time is spent on development</li> <li>• Relatively simple to edit what you have built</li> <li>• Often tech support available</li> </ul>	<p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize colour scheme and functionality in almost any way imaginable</li> <li>• Once you learn one language, you can learn others as they are all interconnected</li> <li>• Common coding languages (R, Python, etc) are free to install and use</li> <li>• You own the source code</li> <li>• Most languages have active communities where you can ask for help</li> </ul>	<p>First the advantages.</p> <p>When using software you are using external code with a graphical user interface meaning you can build what you want without having to type all the code yourself.</p> <p>Projects can be completed quicker, reducing costs and freeing up your time.</p> <p>Coding it yourself gives you complete control over the whole process and costs nothing in terms of licensing. You also own the code, meaning it is more secure and giving you intellectual property rights.</p>	
<p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Easy to use and do not require technical knowledge or skills</li> <li>• More agile as 'drag and drop' options allow for quicker building of visualisations</li> <li>• Greater agility leads to lower costs as less time is spent on development</li> <li>• Relatively simple to edit what you have built</li> <li>• Often tech support available</li> </ul>	<p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize colour scheme and functionality in almost any way imaginable</li> <li>• Once you learn one language, you can learn others as they are all interconnected</li> <li>• Common coding languages (R, Python, etc) are free to install and use</li> <li>• You own the source code</li> <li>• Most languages have active communities where you can ask for help</li> </ul>				

3	<p><b>Disadvantages</b></p> <table border="0" style="width: 100%;"> <tr> <td style="vertical-align: top; width: 50%;"> <p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Can be a steep learning curve and training will be required for each different platform you use.</li> <li>• While platforms may have a wide variety of functions, they are still limited to whatever was built in by the developer.</li> <li>• Security is usually stored around you so not having complete control over your code/data.</li> <li>• If you need to change product it can be difficult to migrate your existing/finished work.</li> <li>• Licences can be expensive</li> <li>• You do not own the source code</li> </ul> </td><td style="vertical-align: top; width: 50%;"> <p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize it to your needs and functions in almost any way imaginable.</li> <li>• Learning to code is a daunting and long process.</li> <li>• Coding is constantly evolving and it is up to you to keep up with trends.</li> <li>• Complicated codes take a long time to produce and require constant error checking.</li> <li>• Easy to make mistakes but can be difficult to find the error.</li> <li>• Making small changes to large scripts not always simple and can lead to other errors.</li> <li>• Code is harder to understand for non-technical colleagues.</li> </ul> </td></tr> </table>	<p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Can be a steep learning curve and training will be required for each different platform you use.</li> <li>• While platforms may have a wide variety of functions, they are still limited to whatever was built in by the developer.</li> <li>• Security is usually stored around you so not having complete control over your code/data.</li> <li>• If you need to change product it can be difficult to migrate your existing/finished work.</li> <li>• Licences can be expensive</li> <li>• You do not own the source code</li> </ul>	<p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize it to your needs and functions in almost any way imaginable.</li> <li>• Learning to code is a daunting and long process.</li> <li>• Coding is constantly evolving and it is up to you to keep up with trends.</li> <li>• Complicated codes take a long time to produce and require constant error checking.</li> <li>• Easy to make mistakes but can be difficult to find the error.</li> <li>• Making small changes to large scripts not always simple and can lead to other errors.</li> <li>• Code is harder to understand for non-technical colleagues.</li> </ul>	<p>While platforms can make building models quick and easy, you still need to learn how to use the programme, and if your company licenses a new software then you will need to receive more training, as most programmes have unique methods of usage.</p> <p>On top of this, while platforms often contain a wide suite of processes they are still limited compared to what you can do yourself.</p> <p>A bigger issue also relates to data security- you are not in complete control of your code and data, if the platform provider is hacked your data potentially could be compromised. Also consider what happens if you want to switch providers or they go out of business- migrating your properties across platforms can be difficult.</p> <p>Coding yourself has its challenges too. Not everyone can code and with it being an ever changing process can be a daunting process to begin learning. Even seasoned coders can be unfamiliar with new concepts. Complicated models can take a long time to build and can also be frustrating- it can be difficult to identify where errors will occur as often they won't appear until you're ready to run the entire script. If it is a large script finding errors can be challenging, and minor changes could lead to serious unintended consequences later down the line.</p> <p>Using a platform or coding it</p>	
<p><b>Platforms</b></p> <ul style="list-style-type: none"> <li>• Can be a steep learning curve and training will be required for each different platform you use.</li> <li>• While platforms may have a wide variety of functions, they are still limited to whatever was built in by the developer.</li> <li>• Security is usually stored around you so not having complete control over your code/data.</li> <li>• If you need to change product it can be difficult to migrate your existing/finished work.</li> <li>• Licences can be expensive</li> <li>• You do not own the source code</li> </ul>	<p><b>Coding</b></p> <ul style="list-style-type: none"> <li>• You have complete control over the whole process and can customize it to your needs and functions in almost any way imaginable.</li> <li>• Learning to code is a daunting and long process.</li> <li>• Coding is constantly evolving and it is up to you to keep up with trends.</li> <li>• Complicated codes take a long time to produce and require constant error checking.</li> <li>• Easy to make mistakes but can be difficult to find the error.</li> <li>• Making small changes to large scripts not always simple and can lead to other errors.</li> <li>• Code is harder to understand for non-technical colleagues.</li> </ul>				

		yourself is a decision you must make when completing a project and a decision you must justify in your portfolio. When writing up the task/action section write down what method you used to achieve your goal and give reasons for why you did and not any other method. This could be to prioritise stakeholder needs as they require output in a particular way.	
4	<b>Activity</b> <small>Mr. Jones works in a data analytics department and has been given a project to complete. He must design a dashboard that displays daily KPIs for his stakeholders.  In teams you will be each be assigned a different product (Tableau, Python, etc) and will come up with arguments to convince Mr. Jones to use your product.  Each team will then be given 1 minute to pitch their product.</small>	7 minutes in breakout rooms	10 Minutes

Topic	Setting Up R	Duration	5 Minutes
-------	--------------	----------	-----------

Objectives
<ul style="list-style-type: none"> <li>• Ensure apprentices have correct software for the following R section</li> </ul>

Section No	Section	Notes	Timing
1	 <p><b>Setting Up R</b></p> <p><small>To do:</small> Make sure the correct software installed to use R</p>	Before we finish this session let's take some time to set up R which we will be looking at tomorrow.	
2	<p><small>1. Open Anaconda 2. Click Environments 3. Click Create 4. Name it R and check the R box 5. Make sure the R environment is selected 6. Once ready, click on the triangle next to your new environment and select "Open with Jupyter Notebook" 7. Whenever you want to use R make sure you use this new environment. If you want to use Python make sure you use the environment called base/root 8. To create a R notebook, select new in the Jupyter navigator and select R</small></p>	<ol style="list-style-type: none"> <li>1. Open Anaconda</li> <li>2. Click Environments</li> <li>3. Click Create</li> </ol>	

		<p>4. Name it R and check the R box</p> <p>5. Wait a few minutes for the environment to be created</p> <p>6. Once ready, click on the triangle next to R and select 'Open with Jupyter Notebook'</p>	
--	--	--	--

Topic	Session 1 Recap	Duration	5 Minutes
-------	-----------------	----------	-----------

Objectives	
<ul style="list-style-type: none"> <li>Review the days learning and introduce the assignment</li> </ul>	

Section No	Section	Notes	Timing
1		Review the day	
2	<p><b>Learning Objectives</b></p> <ul style="list-style-type: none"> <li>Understand different contexts of Big Data</li> <li>Justify the use or lack of Big Data technologies in your analysis</li> <li>Critically evaluate the differences in using a data platform and coding it yourself</li> </ul> <p><b>Assignment</b></p> <p>Justify the use (or not) of Big Data technologies in your analysis</p> <p>Required:</p> <p>Written Justification (max 1500 words)</p> <p>Things to consider:</p> <ul style="list-style-type: none"> <li>How does your company make use of Big Data technologies?</li> <li>Do you have opportunities to use them in your role?</li> <li>Do you think you could use these technologies effectively in your role?</li> <li>Do you think your company uses them effectively?</li> <li>What do you think the potential business impact of your companies attitude towards Big Data will be?</li> </ul>		

Topic	Session 2 Intro	Duration	10 Minutes
-------	-----------------	----------	------------

## Objectives

- To provide an overview of the class agenda and the expected learning objectives

Section No	Section	Notes	Timing
55	<b>Session 2: R</b> <p>In this session we will be exploring R, a computing language which has been optimised for statistical analysis and machine learning. While it does not have the range of functionality Python has it more than makes up for it with the sorts of packages it has for data manipulation, analysis and visualisation. This notebook will cover the basics of R, exploratory data analysis and visualisations.</p> <p><b>Session Outline</b></p> <ul style="list-style-type: none"> <li>• Features of R</li> <li>• Creating Data in R</li> <li>• Exploratory Data Analysis in R</li> <li>• Manipulating Data</li> <li>• Linear Models in R</li> <li>• Scripts</li> <li>• etc...</li> </ul> <p><b>Learning Objectives</b></p> <ul style="list-style-type: none"> <li>• Understand the Basics of R</li> <li>• Compare and Contrast the functions of R with Python</li> <li>• Utilise R to perform Data Analytics</li> </ul>	Coach welcome to session, run an ice breaker from <a href="#">here</a> . Run through session plan and learning objectives	

Topic	Features of R	Duration	30 Minutes
-------	---------------	----------	------------

## Objectives

- Understand the Basic Features of R

Section No	Section	Notes	Timing
1	 <p><b>Features of R</b></p> <p><small>After Understanding the Basic Features of R:</small> R is a statistical language developed in 1995 by Ross Ihaka and Robert Gentleman at the University of Auckland. It was named after the first letter of both author's names, and as a play on S, the computing language R was built upon. R is mostly used for statistical computation and analysis but has also gained attention for developing machine learning products. Like Python, R's statistical computing capabilities can be extended with packages (called packages). R is also an open-source language, so it is free to use and learn. R is similar to Python in terms of syntax, making it easier for programmers to learn. However, the wider range of statistical tools and the ease of building visualizations makes R a favorite of statisticians across the globe.</p>	<p>Explain that R is a language built for statistical analysis built in 1995.</p> <p>It carries much of the same functionality as python but has different syntax.</p> <p>In this session you will be showing the R equivalents to different concepts learned previously.</p> <p>There will be chances to practice as well.</p>	

2	<p><b>Packages</b></p> <p>R packages are collections of functions that allow for reusability, modularity and building cleaner code through extending R capabilities. Packages are stored in repositories called Comprehensive R Archive Networks or CRANs. As with Python, R is open source meaning anyone can write packages and you can install any package available in a CRAN.</p> <p>A script in R is a text file with extension .R where you can write your code and save anywhere.</p>	<p>In R repositories are called CRAN.</p>																			
3	<p><b>Data Types in R</b></p> <p>The table below shows the different data types in R and what their python equivalents are.</p> <table border="1"> <thead> <tr> <th>Data Type (R)</th> <th>Python Equivalent</th> <th>Example</th> </tr> </thead> <tbody> <tr> <td>Character</td> <td>String</td> <td>"Multiverse" 972360C</td> </tr> <tr> <td>Integer</td> <td>Integer</td> <td>3, 7, 102</td> </tr> <tr> <td>Numeric</td> <td>Float</td> <td>10.5, 3.14, 2.3</td> </tr> <tr> <td>Logical</td> <td>Boolean</td> <td>TRUE, FALSE</td> </tr> <tr> <td>NA</td> <td>Nan</td> <td></td> </tr> </tbody> </table> <p>Note that for logical data types they are all capitalised as opposed to python where only the first letter is uppercase.</p> <p>If you are ever stuck in R, remember there is the ?help() function to bring up the documentation. Otherwise stackoverflow has plenty of solutions to common problems.</p>	Data Type (R)	Python Equivalent	Example	Character	String	"Multiverse" 972360C	Integer	Integer	3, 7, 102	Numeric	Float	10.5, 3.14, 2.3	Logical	Boolean	TRUE, FALSE	NA	Nan		<p>Introduce the different data types and how they relate to those from python.</p>	
Data Type (R)	Python Equivalent	Example																			
Character	String	"Multiverse" 972360C																			
Integer	Integer	3, 7, 102																			
Numeric	Float	10.5, 3.14, 2.3																			
Logical	Boolean	TRUE, FALSE																			
NA	Nan																				
4	<p><b>Declaring Variables in R</b></p> <p>For a single variable</p> <pre>In [1]: x = "my variable" Traditionally we use &lt;-&gt; to assign a variable, but = works just as well.  In [2]: x = my_variable Like python, R has a print function to see what is stored in a variable.  In [3]: print(x) Or we could just call the variable name itself to see what is stored.  In [4]: x We can also call a collection of integers  In [5]: y = 1:5 You can check the datatype of a variable by using the command: typeof()  In [6]: typeof(x) In [7]: # we can add comments in R the same way we do in Python  In [8]: # Not there are other methods of doing it too"</pre>	<p>Demonstrate how to use basic R functions like calling a variable.</p> <p>Note that you can use = or &lt;- when assigning a variable.</p>																			
5	<p><b>Mathematical Operators</b></p> <p>Like python, R has a range of built operators</p> <pre>In [1]: 1 + 1 In [2]: 10 * 5 In [3]: 5 * 4 In [4]: 2 ** 5 In [5]: 1000 % 5 # modulo In [6]: 100 / 9 In [7]: sqrt(23) In [8]: mean(c(5,4))</pre>	<p>Go through the different mathematical operators. Note how square root and mean are already built in, whereas in python you had to import a library.</p>																			
6	<p><b>Data Structures in R</b></p> <p>Like in python there exist data structures we use to store our data for analysis.</p> <p>1) <b>List:</b></p> <p>A collection of elements that can hold a combination of data types. Note, a list is also similar to a dictionary in python</p> <pre>In [1]: l = list("a", "b", "c") [1] "a" "b" "c"  If I want to call an element from a list I need to use the \$-notation and reference the tag.  In [2]: l\$a Lists do not require tags to be called however.  In [3]: d = list(x="some text", y=100) d You can reference elements from a list by using their key. Lists in R are more similar to dictionaries in python.  In [4]: d[x] In [5]: d["x"]  If I want to replace this value for further use I need to use [[[]]  In [6]: d[[x]] = 100 d In [7]: d["x"]  To add items to a list you need to call the tag of where you want it to be placed and assign it the value you want.  In [8]: d[[x]] = "Multiverse" [1] "Multiverse"  Editing a value in a list follows the same format.</pre>	<p>In the following section you will be explaining much of the different data structures in R.</p> <p>Start with lists, note how these are more similar to dictionaries in python. They can hold a combination of data types and have tags.</p> <p>Demonstrate how to reference lists and manipulate them.</p>																			
7	<p><b>2) Vectors</b></p> <p>A collection of elements of the same data type, similar to numpy arrays</p> <p>We use .x() to define a vector</p> <pre>In [1]: numbers &lt;- c(1,2,3); numbers In [2]: str(numbers) To edit, append or view values from a vector you just use the same notation as a list. Note that vectors are not zero indexed.  In [3]: numbers[1] In [4]: numbers[1:3] # access several elements In [5]: numbers[3]; n; numbers In [6]: numbers[3]; numbers  We can make it more like a dictionary and apply tags to the values.  In [7]: more_numbers &lt;- c(first=1, second=2, third=3); more_numbers In [8]: more_numbers["first"] In [9]: more_numbers["first"]  What is the difference between a list and a vector?  In [10]: # A:</pre> <p>Click here for solution</p>	<p>Show vectors and highlight they are different to lists as they must contain the same data type.</p> <p>Demo basic functions.</p>																			

8	<pre> <b>3) Matrices:</b> A table like structure with rows and columns (an extension of a vector)  In [1]: X = matrix(c(1,2,3,4),ncol=2); print(X)           [,1] [,2]      [1,]    1    2      [2,]    3    4  In [2]: print(X)           [,1] [,2]      [1,]    1    2      [2,]    3    4  To add a column to a matrix we use c(...,n)  In [3]: X2 = cbind(X,c(5,6)); print(X2)           [,1] [,2] [,3]      [1,]    1    2    5      [2,]    3    4    6  To add a row to a matrix we use rbind()  In [4]: X3 = rbind(X,c(7,8,9)); print(X3)           [,1] [,2]      [1,]    1    2      [2,]    3    4      [3,]    7    8      [4,]    9   10 </pre>	Introduce matrices as an extension of a vector.	
9	<pre> <b>4) DataFrames:</b> Another data structure to perform statistical operations. Other than vectors, dataframes will likely be the most used data structure in data manipulation and analysis  In [1]: df = data.frame(name=c("John","Peter","Paul","Mick"),age=c(20,22,25,28)); print(df)       name age 1 John   20 2 Peter  22 3 Paul   25 4 Mick   28  In [2]: str(df)  In [3]: dfform(df)  Adding rows and columns is the same as matrix  In [4]: df = rbind(df,data.frame(name="George",age=26))       name age 1 John   20 2 Peter  22 3 Paul   25 4 Mick   28 5 George 26  In [5]: df = cbind(df,data.frame(three=c(TRUE,FALSE,TRUE,TRUE, FALSE))); df       name age three 1 John   20  TRUE 2 Peter  22 FALSE 3 Paul   25  TRUE 4 Mick   28  TRUE 5 George 26 FALSE  Or we can add columns by using the \$ notation  In [6]: df\$name = c("X","Y","Python","Python","X") ; df       name age three 1 John   20  TRUE 2 Peter  22 FALSE 3 Paul   25  TRUE 4 Mick   28  TRUE 5 George 26 FALSE  To delete a column, the simplest method is to create a subset() and select which rows you want to drop using -c()  In [7]: df = subset(df,excete=c(2,5))  To delete a row, you reference the data frame by calling the complement of the rows you do not want by using -c()  In [8]: df[-c(1,5)]       name age three 2 Peter  22 FALSE 3 Paul   25  TRUE 4 Mick   28  TRUE  Remember to reassign the data frame to your variable name or your changes won't actually hold </pre>	Introduce dataframes and link them to pandas. State how you will be using them a lot with data manipulation.  Demonstrate how you can columns or rows and how to reference a column	
10	<p><b>Exercise</b> Create the following: 1. A variable to store your name 2. A variable to store a vector from 1 to 10 3. A variable to store a vector showing your name, age, city and if your CTJ is above 100% (True or False) 4. A variable to store a data frame with three columns: name, age and CTJ for 5 people</p> <pre>In [1]: # A:</pre>	Give them 5 minutes to complete this exercise	5 Minutes

Topic	Control Flow in R	Duration	30 Minutes
-------	-------------------	----------	------------

Objectives	<ul style="list-style-type: none"> <li>Understand how to set up loops and functions in R</li> </ul>
------------	---

Section No	Section	Notes	Timing																
1	 <p><b>Control Flow in R</b></p> <p><small>Top Aims: Understand how to set up loops and functions in R Like with Python, R allows you to write loops which you can use to parse data or generate new data structures for your analysis.</small></p>	Depending on the skill level of the group, this can be skipped.  In python they learned how to write loops to parse data and automate processes. We will now show them how to do this in R																	
2	<p><b>Logic Statements</b></p> <p>In python these were denoted by if followed by some logical statement e.g. if ==2: R allows you to do these as well, although the logic statements are the same, the syntax for calling the statement is different</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding-right: 20px;">Operator</th> <th style="text-align: left; padding-right: 20px;">Meaning</th> </tr> </thead> <tbody> <tr> <td style="padding-right: 20px;">==</td> <td>Equal</td> </tr> <tr> <td style="padding-right: 20px;">!=</td> <td>Not Equal</td> </tr> <tr> <td style="padding-right: 20px;">&gt;</td> <td>Greater Than</td> </tr> <tr> <td style="padding-right: 20px;">&gt;=</td> <td>Greater Than or Equals To</td> </tr> <tr> <td style="padding-right: 20px;">&lt;</td> <td>Less Than</td> </tr> <tr> <td style="padding-right: 20px;">&lt;=</td> <td>Less Than or Equals To</td> </tr> <tr> <td style="padding-right: 20px;">is</td> <td>Is In</td> </tr> </tbody> </table>	Operator	Meaning	==	Equal	!=	Not Equal	>	Greater Than	>=	Greater Than or Equals To	<	Less Than	<=	Less Than or Equals To	is	Is In	Show the different logic statements, note how similar they are to python.  Demonstrate if statements and build in else then else if. Note	
Operator	Meaning																		
==	Equal																		
!=	Not Equal																		
>	Greater Than																		
>=	Greater Than or Equals To																		
<	Less Than																		
<=	Less Than or Equals To																		
is	Is In																		

		the difference from python (not elif) and the way the statement is called ({} instead of :)	
3	<p><b>For Loops</b></p> <p>Now we know how to build a logical statement we can add those into for loops. In this example we will loop through a vector of names and use the <code>paste()</code> command to create a string list.</p> <pre>In [1]: for (name in c("Jack", "Andrew", "Joseph", "Laura", "Grace", "Natalie", "Hayden")){   print(paste("Name: ", name)) }</pre> <p>It's simple to add in a logical statement to our loop, for example I've wanted to check if a name had 4 letters in it. Note we use <code>ischar()</code> in R to check string length.</p> <pre>In [2]: for (name in c("Jack", "Jack", "Andrew", "Joseph", "Laura", "Grace", "Natalie", "Hayden")){   if (ischar(name) == 4){     print(paste("Name: ", name, "has 4 letters"))   } else {     print(paste("Name: ", name, "does not have 4 letters"))   } }</pre>	Show how to build a for loop and how to add in a logic statement	
4	<p><b>While Loops 1</b></p> <p>The syntax is similar to before, always remember to add in your self-close!</p> <pre>In [3]: n = 20 while (n &gt; 0):   print(paste("Time: ", n, " is ", "n"))   n = n - 1</pre>	Do the same with while loops.	
5	<p><b>Functions</b></p> <p>Functions would be useful for storing our code so we can use them again later. The process of building a function is simple and consists of defining the function and then writing what it does. For example, we can write a function that returns the square root of the difference of two squared numbers.</p> <pre>In [4]: sqrt_squared_diff &lt;- function(x,y){   if (x &gt; y){     return ((sqrt(x)*2-y)*2)   } else {     return ((sqrt(y)*2-x)*2)   } }</pre> <pre>In [5]: sqrt_squared_diff(5,4)</pre>	Show how to build a function, note the use of the return statement.	
6	<p><b>Exercise</b></p> <p>1 Write a loop which squares the numbers from 1 to 20 and prints the result    2 Write a loop which takes 5 names and prints TRUE/FALSE if the number of characters is greater than 7    3 Write a loop that counts down from 20. You have (number) seconds left. When the number reaches 0 print a 'Game Over' statement.    4 Write a loop that prints out the first 1000 prime numbers. Hint: This is a multiple of 5 and 7 because 5*7=35    5 Create a game of rock paper scissors. As this for the numbers from 1 to 100. Scissors you will need to look up it's random and user input functions. Note that you should check the player's input is valid.</p> <pre>In [6]: # A: Click here for solutions</pre>	Solutions embedded to notebook	10 Minutes

Topic	Exploratory Data Analysis in R	Duration	35 Minutes
-------	--------------------------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>Consider how to perform EDA in R</li> </ul>

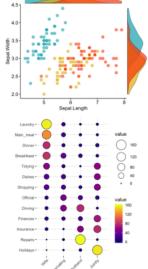
Section No	Section	Notes	Timing
1	 <p><b>Exploratory Data Analysis in R</b></p> <p><small>10m Aim: Consider how to perform EDA in R</small></p> <p>10% of your spent on a data project will be spent cleaning data, the rest will be spent analysing – pretty much every data scientist and analyst ever. We have over time and time again throughout this course that EDA is an important and unique part of the data analysis life cycle. The principles in R are the same as with Python and Excel but the processes are different. We will now look at some common EDA tools you will use with R.</p>	<p>Link back to previous lessons on EDA (python, SQL, tableau/PowerBI, Excel) and reiterate its importance.</p> <p>R has many functions readily available to help with this which are easy to use.</p>	

2	<p>Much of the functionality of R is enhanced by functions and packages. Just as in Python, some functions are pre-installed or we can use the closest available packages. To build them in Note, you only have to install a package once and then any time you want to use it you just call the name of the package.</p> <p>For this lesson we will use tidyverse which includes everything you see in this image:</p>  <pre>In [1]: # After the first time install.packages("tidyverse")  In [2]: # Every time after library(tidyverse)</pre>	<p>Packages are the same as libraries in python (many packages in R can be used in python as well). Show how to install (only need to do once) and then how to call it up.</p>													
3	<p>We are now going to look through some common EDA commands you will use in your analysis</p> <pre>In [1]: # Loading a data set data(bottles_products)  # Note different import functions exist for different data files  In [2]: # Check the head head(data)  In [3]: # Check summary statistics summary(data)  In [4]: # For one statistic mean(data\$bottle_size)  In [5]: # For one statistic sd(data\$bottle_size)  In [6]: # Correlations cor(data\$bottle_size)</pre>	<p>Show how to read in data and how to perform the basic checks we saw in python.</p>													
4	<p><b>Common Functions</b></p> <p>The following table shows the most common functions of tidyverse that you will use.</p> <table border="1" data-bbox="373 920 771 1154"> <thead> <tr> <th>Function</th> <th>What It Does</th> </tr> </thead> <tbody> <tr> <td>filter()</td> <td>Include or exclude certain rows</td> </tr> <tr> <td>arrange()</td> <td>Change the order of observations</td> </tr> <tr> <td>select()</td> <td>Create new columns</td> </tr> <tr> <td>mutate()</td> <td>Generate new variables</td> </tr> <tr> <td>summarise()</td> <td>Perform aggregate analysis</td> </tr> </tbody> </table> <pre>In [1]: # Filter filter(data, proof==40)  In [2]: # If more than one filter? filter(data, proof==40   packid==1)  In [3]: # Arranging arrange(data, vendor)  In [4]: # Change the order arrange(data, desc(vendor))  In [5]: # Filter then arrange arrange(filter(data, bottle_size&gt;10), vendor)</pre>	Function	What It Does	filter()	Include or exclude certain rows	arrange()	Change the order of observations	select()	Create new columns	mutate()	Generate new variables	summarise()	Perform aggregate analysis	<p>Introduce the common functions and state that you will be demonstrating them shortly.</p> <p>Note the odd syntax of trying to apply more than one function.</p>	
Function	What It Does														
filter()	Include or exclude certain rows														
arrange()	Change the order of observations														
select()	Create new columns														
mutate()	Generate new variables														
summarise()	Perform aggregate analysis														
5	<p><b>Piping</b></p> <p>Using subroutines in R can be tedious if you want to apply more than one to your data as every new function needs to be wrapped around your statement (nesting it could get quite long). Instead we can use a technique called piping which allows you to concatenate functions into a single statement using %&gt;%</p> <p>Using the example from above</p> <pre>In [1]: data %&gt;% filter(bottle_size&gt;10) %&gt;% arrange(vendor)  In [2]: # This can be inserted in a variable d = data %&gt;% filter(bottle_size&gt;10) %&gt;% arrange(vendor)  Piping is a more intuitive way of writing your code as it starts with your data and then applies the functions.</pre> <p>The process can be summarized by:</p> <pre>'data %&gt;% filter(col1 == "Yes") %&gt;% arrange(col2)'  Data Source           Function 1           Function 2</pre>	<p>As wrapping functions can be confusing, piping is a better way of adding in several layers to the same command.</p> <p>Demonstrate how much simpler piping is compared to wrapping.</p>													
6	<p><b>Slicing, Column Selection and Renaming</b></p> <p>Note that R is 1-indexed (unlike Python which is 0-indexed). Keep this in mind when slicing data</p> <pre>In [1]: data %&gt;% select(-1)  Selecting a column is as simple as just saying what column you want  In [2]: data %&gt;% select(category_name) %&gt;% head()  When renaming a column notice that the new name comes first  In [3]: data %&gt;% rename(vendor_no=vendor)</pre>	<p>Show how to slice a datafram, select and rename columns.</p>													
7	<p><b>Mutate, Group By and Summarise</b></p> <p>Mutate allows you to update or create new columns</p> <pre>In [1]: data %&gt;% mutate(total_cost=gross_cost)  In [2]: data %&gt;% mutate(vendor_no=vendor)  Summarise allows you to sum many observations into a single datapoint. This is not to be confused with summarise() which we saw earlier</pre> <pre>In [3]: data %&gt;% summarise(mean_bottle_size=mean(bottle_size))  Summarise is particularly powerful when combined with group_by which allows you to summarise within a group</pre> <pre>In [4]: data %&gt;% group_by(category_name) %&gt;% summarise(mean_bottle_size=mean(bottle_size)) %&gt;% head()</pre>	<p>Demonstrate how to add new columns or amend an existing one, how to group by and aggregate data frames.</p>													
8	<p><b>Exercise</b></p> <ol style="list-style-type: none"> <li>Import gdp minder data using <code>read_csv</code></li> </ol> <pre>In [1]: gdp_data &lt;- read_csv("./data/gdp_minder_data.csv")</pre> <p>Check the head, tail and summary statistics of your data</p> <pre>In [2]: # A: Click here for solution</pre> <ol style="list-style-type: none"> <li>Calculate the summary statistics for the <code>pop</code> column</li> </ol> <pre>In [3]: # A: Click here for solution</pre>	<p>Solutions embedded</p>	<p>10 Minutes</p>												

Topic	Visualisation in R	Duration	35 Minutes
-------	--------------------	----------	------------

## Objectives

- Build Visualisations in R

Section No	Section	Notes	Timing
1	 <p>The title slide for the 'Build Visualisations in R' section.</p> <p>Aim: Build Visualisations in R</p> <p>We know from previous module that visualisation is an important part of our exploratory data analysis. You have seen how to do it in Excel, Python and with a platform. For your portfolio it is important you show a range of skills and techniques so it is a good idea to consider how you can demonstrate some analysis using R, and visualisation is a good place to start.</p>	<p>Visualisation is an important part of EDA and presenting your findings.</p> <p>It is good for your portfolio to have a variety of methods, so consider building some in R.</p>	
2	<p><b>Visualisation with ggplot</b></p> <p>The main visualisation package we will use is ggplot which offers a wide variety of visualisations for you to choose from.</p>  <pre> In [1]: # To build a scatter plot we need to state the data source, what our x and y values are and what type of plot we want. ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width)) + geom_point()  In [2]: # If we wanted to colour by species we could do this ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width,color=Species)) + geom_point()  In [3]: # To build a bar graph we change to geom_bar. In this example we are calculating the mean life_expectancy per continent ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary",fun=mean)  In [4]: # As before ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary")  We can also change things about our plots to make them more attractive, such as colours, lines and captions.  In [5]: ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary",color="red","green","yellow","orange") +   labs(x="Continent",y="Mean Life Expectancy",title="Mean Life Expectancy per Continent")  If we want to fit more than one visualisation on the page, we can use faceting to create subplots.  In [6]: ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width)) + geom_point() + facet_wrap(~continent) </pre> <p>Of course there are far more things you can do to build visualisations in R; more than we can show you in this session. Like Python, R has a well documented API to help you understand and build the visualisations you want.</p>	<p>The library we will be using is ggplot2 (comes with tidyverse). It is simple to use and has a wide array of functionality.</p> <p>While we will be keeping it simple, there are many ways you can build outstanding visualisations.</p> <p>Note: plotly dash can also be run in R</p>	
3	<pre> The syntax for building a visualisation in ggplot(data,aesthetics) + plot_type. Let's build a simple scatter graph from our iris_data to demonstrate.  In [1]: # To build a scatter plot we need to state the data source, what our x and y values are and what type of plot we want. ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width)) + geom_point()  In [2]: # If we wanted to colour by species we could do this ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width,color=Species)) + geom_point()  In [3]: # To build a bar graph we change to geom_bar. In this example we are calculating the mean life_expectancy per continent ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary",fun=mean)  In [4]: # As before ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary")  We can also change things about our plots to make them more attractive, such as colours, lines and captions.  In [5]: ggplot(data=iris_data,aes(x=continent,y=life_expectancy)) + geom_bar(stat="summary",color="red","green","yellow","orange") +   labs(x="Continent",y="Mean Life Expectancy",title="Mean Life Expectancy per Continent")  If we want to fit more than one visualisation on the page, we can use faceting to create subplots.  In [6]: ggplot(data=iris_data,aes(x=Sepal.Length,y=Sepal.Width)) + geom_point() + facet_wrap(~continent) </pre> <p>Of course there are far more things you can do to build visualisations in R; more than we can show you in this session. Like Python, R has a well documented API to help you understand and build the visualisations you want.</p>	<p>Show how to build a scatter plot, bar chart and box plot. Show how to add colour and other features to the plots.</p> <p>Show facet_wrap as a way of putting multiple plots on the page.</p> <p>Note the structure of geom_graph and state that this is the general structure of</p>	

	ggplots.	
4	<p><b>Exercise</b></p> <ol style="list-style-type: none"> <li>1. Plot the gdpPercap against lifeExp as a scatter plot using 2007 data</li> <li>2. Plot the gdpPercap against lifeExp as a scatter plot, colour by continent using 2007 data</li> <li>3. Plot the gdpPercap such as lifeExp as a scatter plot, colour by continent and size by pop using 2007 data (Hint: you can add the argument size to the scatterplot function)</li> <li>4. Make a line plot of the pop column for the United Kingdom (Hint: you will need geom_line() not geom_point())</li> </ol> <pre>In [1]: # A:</pre> <p><a href="#">Click here for solutions</a></p> <p>Stretch</p>	Solutions embedded, encourage those who are more confident to attempt the stretch questions. 15 Minutes

Topic	Linear Modelling in R	Duration	15 Minutes
-------	-----------------------	----------	------------

Objectives
<ul style="list-style-type: none"> <li>• Learn about different modelling techniques in R</li> </ul>

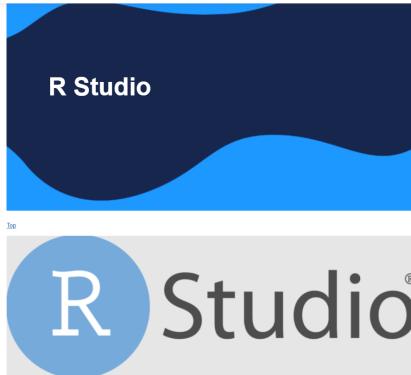
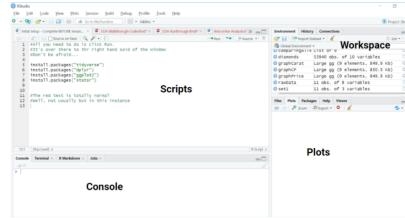
Section No	Section	Notes	Timing
1	 <p>This section can be skipped if short on time or the group not confident enough.</p> <p>Amongst R's many applications is statistical analysis and machine learning. In fact, R is possibly better suited for it due to the wide range of built in statistical tests and models already built in.</p> <p>In python we saw different types of hypothesis tests, R has more and the syntax is easier!</p> <p>It can also easily do ANOVA, chi-square test, f-test and more.</p> <p>We will look at a few now.</p>		

2	<pre><code>Linear Regression Think back to statsmodels we saw in Python for building a model. The syntax is very similar (If not simpler). To start all you need is 3: (dependent ~ independent, data=df)  In [1]: myModel = lmfit.LinearRegression().fit(gdpPerCap,data=gdpData) In [2]: myMultiModel = lmfit.LinearRegression().fit(gdpPerCap, data=gdpData) # Adding in more than one predictor To evaluate your model, checking coefficients and scores all you need is .summary()  In [3]: summary(myModel)  Value you should expect: R-squared and the coefficients of your predictors (labeled as Estimates). Like with statsmodels, the Df(2) will tell you how statistically significant the model is (lower is good) In [4]: summary(myMultiModel)</code></pre>	<p>Demonstrate how to build a linear regression.</p> <p>Show how to call a model and note the simplicity. Train test split is possible, but we won't be showing it today.</p> <p>Walk through the summary and draw particular attention to p-values and r squared.</p>	
3	<pre><code>Logistic Regression In Statsmodels logistic regression is known as a logit model and can be accessed using a generalized linear model (GLM). For this example we will create a new column for our 2007 data called 'Is_Europe' where if a country is in Europe the value will be 1 and 0 otherwise. We can use the .get_dummies function combined with np.where to create this feature  In [1]: gdp_europe_2007 = gdp_2007[~gdp_2007['is_europe'].where(~gdp_2007['continent']=='Europe',1,0)] In [2]: logitModel = glm('Is_Europe ~ gdpPerCap', data=gdp_europe_2007) In [3]: summary(logitModel)</code></pre>	<p>Do the same for logistic regression. Walk through the data engineering steps (not seen yet) and then how to build and interpret the model.</p> <p>Note the similarity to the python statsmodels package.</p> <p>Also note the different name from python.</p>	
4	<pre><code>t-test Earlier in the course we discussed hypothesis testing to see if something was statistically significant (e.g. more students failing a test than expected). We can use a t-test to check if there is a statistically significant difference between our observed mean and the one we would expect. For example, in 1997 the average life expectancy in Asia was 61 but by 2007 it was nearly 70. Is this a significant increase? We can check with a t-test  In [1]: gdpData = pd.read_csv('lifeExp.csv') #Rd group_by(continent) .Rd summarise(means_l1=fehlyr_mean(lifeExp)) In [2]: gdpData.Rd %&gt;% filter(year==1997) Rd group_by(continent) .Rd summarise(means_l1=fehlyr_mean(lifeExp)) In [3]: gdpData.Rd %&gt;% filter(year==2007) Rd group_by(continent) .Rd summarise(means_l2=fehlyr_mean(lifeExp)) H0: Life Expectancy in 2007 is the same as in 1997 (means=0) H1: Life Expectancy in 2007 is not the same as in 1997 (means!=0) This is a two tailed test which we will conduct at the 5% level  However, this is not a two tailed test. We can amend our code for it to be one tailed.  The values to pick are the p-value (less than 0.05 so we have sufficient evidence to reject the null hypothesis in favour of the alternative) and the 95% confidence interval. We can use the t.test function in R to do this. In R, the p-value for the hypothesis test for Asia in 2007 is between 67.9 and 73.5. As the 1997 average of 61 does not fall in this range we have further evidence we should reject the null hypothesis.  In [4]: t.test(gdpData\$lifeExp[which(gdpData\$continent=="Asia")], mu=61, alternative="greater")  There are many other tests available in R to consider for hypothesis testing. This is an area where R has a clear advantage of Python as these tests are already built in.</code></pre>	<p>Finally show a t-test. Stress the importance that normally you would do checks such as looking at distributions, summary statistics, etc but for today we are only going to look at how to run and interpret one.</p>	

Topic	RStudio	Duration	5 Minutes
-------	---------	----------	-----------

Objectives
<ul style="list-style-type: none"> <li>• Be aware that RStudio exists</li> </ul>

Section No	Section	Notes	Timing
------------	---------	-------	--------

1		We used Jupyter because apprentices are already familiar with it  In most industry cases they would use RStudio.	
2		If you have time, download and bring up RStudio to show how it works and although the GUI is different, the syntax is the same.	

Topic	Recap	Duration	5 Minutes
-------	-------	----------	-----------

Objectives
<ul style="list-style-type: none"> <li>• Recap the day</li> </ul>

Section No	Section	Notes	Timing
1	  <b>Learning Objectives</b> <ul style="list-style-type: none"><li>• Understand the basic of R</li><li>• Compare and Contrast the functions of R with Python</li><li>• Utilise R to perform Data Analytics</li></ul>	Recap the day, remind apprentices about OTJ, assignments and SALs.  Assignment is in other notebook titled 'assignment.'  Solutions are in repo.	