**Google Data Analytics**

**Data Analysis with R Programming**

**Vectors:**

Store data of similar types

**Lists:**

Store data of different types

**Dataframe:**

* ﻿﻿Columns should be named
* ﻿﻿Data stored can be many different types, like numeric, factor, or character
* ﻿﻿Each column should contain the same number of data items

**Tibbles:**

Streamlined dataframes

* ﻿﻿Never change the data types of the inputs
* ﻿﻿Never change the names of your variables
* ﻿﻿Never create row names
* Make printing easier

**Tidy data (R)**

A way of standardising the organisation of data within R

Tidy data standards

* ﻿﻿Variables are organised into columns
* ﻿﻿Observations are organised into rows
* ﻿﻿Each value must have its own cell

**File Naming:**

### 

### **Do**

* Keep your filenames to a reasonable length
* Use underscores and hyphens for readability
* Start or end your filename with a letter or number
* Use a standard date format when applicable; example: YYYY-MM-DD
* Use filenames for related files that work well with default ordering; example: in chronological order, or logical order using numbers first

### **Don't**

* Use unnecessary additional characters in filenames
* Use spaces or “illegal” characters; examples: &, %, #, <, or >
* Start or end your filename with a symbol
* Use incomplete or inconsistent date formats; example: M-D-YY
* Use filenames for related files that do not work well with default ordering; examples: a random system of numbers or date formats, or using letters first

**Operators:**

**Arithmetic operators**

| **Operator** | **Description** | **Example Code** |
| --- | --- | --- |
| **+** | **Addition** | **x + y** |
| **-** | **Subtraction** | **x - y** |
| **\*** | **Multiplication** | **x \* y** |
| **/** | **Division** | **x / y** |
| **%%** | **Modulus (returns the remainder after division)** | **y %% x** |
| **%/%** | **Integer division (returns an integer value after division)** | **y%/% x** |
| **^** | **Exponent** | **y ^ x** |

**Relational operators**

| **Operator** | **Description** | **Example Code** |
| --- | --- | --- |
| **<** | **Less than** | **x < y** |
| **>** | **Greater than** | **x > y** |
| **<=** | **Less than or equal to** | **x < = 2** |
| **>=** | **Greater than or equal to** | **y >= 10** |
| **==** | **Equal to** | **y == 5** |
| **!=** | **Not equal to** | **x != 2** |

**Logical operators**

| **Operator** | **Description** |
| --- | --- |
| & | Element-wise logical AND |
| && | Logical AND |
| | | Element-wise logical OR |
| || | Logical OR |
| ! | Logical NOT |

**Note:**The main difference between **element-wise logical operators** (&, |) and **logical operators** (&&, ||) is the way they apply to operations with vectors. The operations with double signs, AND (&&) and logical OR (||), only examine the *first* element of each vector. The operations with single signs, AND (&) and OR (|), examine all the elements of each vector

**Assignment operators**

| **Operator** | **Description** | **Example Code (after the sample code below, typing x will generate the output in the next column)** |
| --- | --- | --- |
| **<-** | **Leftwards assignment** | **x <- 2** |
| **<<-** | **Leftwards assignment** | **x <<- 7** |
| **=** | **Leftwards assignment** | **x = 9** |
| **->** | **Rightwards assignment** | **11 -> x** |
| **->>** | **Rightwards assignment** | **21 ->> x** |

**Data visualisation in R:**

**Packages for visualisation:**

* ggplot2
* ﻿﻿Plotly
* ﻿﻿Lattice
* ﻿﻿RGL
* ﻿﻿Dygraphs
* ﻿﻿Leaflet
* ﻿﻿Highcharter
* ﻿﻿Patchwork
* ﻿﻿gganimate
* ﻿﻿ggridges

**ggplot2:**

**Aesthetic**

A visual property of an object in your plot

**Geom**

The geometric object used to represent your data

**Facets**

Let you display smaller groups, or subsets, of your data

**Labels and annotations**

Let you customise your plot

**Smoothing:**

Smoothing enables the detection of a data trend even when you can't easily notice a trend from the plotted data points

| **Type of smoothing** | **Description** | **Example code** |
| --- | --- | --- |
| **Loess smoothing** | **The loess smoothing process is best for smoothing**  **plots with less than 1000 points.** | **ggplot(data, aes(x=, y=))+**  **geom\_point() +**  **geom\_smooth(method="loess")** |
| **Gam smoothing** | **Gam smoothing, or generalised additive model**  **smoothing, is useful for smoothing plots with a large**  **number of points.** | **ggplot(data, aes(x=, y=)) +**  **geom\_point() +**  **geom\_smooth(method="gam",**  **formula = y ~s(x))** |

**Saving plots without ggsave() function:**

| **Example of using png()** | **Example of using pdf()** |
| --- | --- |
| **png(file = "exampleplot.png", bg = "transparent")**  **plot(1:10)**  **rect(1, 5, 3, 7, col = "white")**  **dev.off()** | **pdf(file = "/Users/username/Desktop/example.pdf",**  **width = 4,**  **height = 4)**  **plot(x = 1:10,**  **y = 1:10)**  **abline(v = 0)**  **text(x = 0, y = 1, labels = "Random text")**  **dev.off()** |

**Document and Reports in R:**

**R Markdown**

A file format for making dynamic documents with R

**Markdown**

A syntax for formatting plain text files

**R Notebook**

Lets users run your code and show the graphs and charts that visualise the code

**Some popular packages with templates for R Markdown include the following:**

* **The** [**vitae**](https://github.com/mitchelloharawild/vitae) **package contains templates for creating and maintaining a résumé or curriculum vitae (CV)**
* **The** [**rticles**](https://github.com/rstudio/rticles) **package provides templates for various journals and publishers**
* **The** [**learnr**](https://github.com/rstudio/learnr) **package makes it easy to turn any R Markdown document into an interactive tutorial**
* **The** [**bookdown**](https://github.com/rstudio/bookdown) **package facilitates writing books and long-form articles**
* **The** [**flexdashboard**](https://github.com/rstudio/flexdashboard) **package lets you publish a group of related data visualisations as a dashboard**