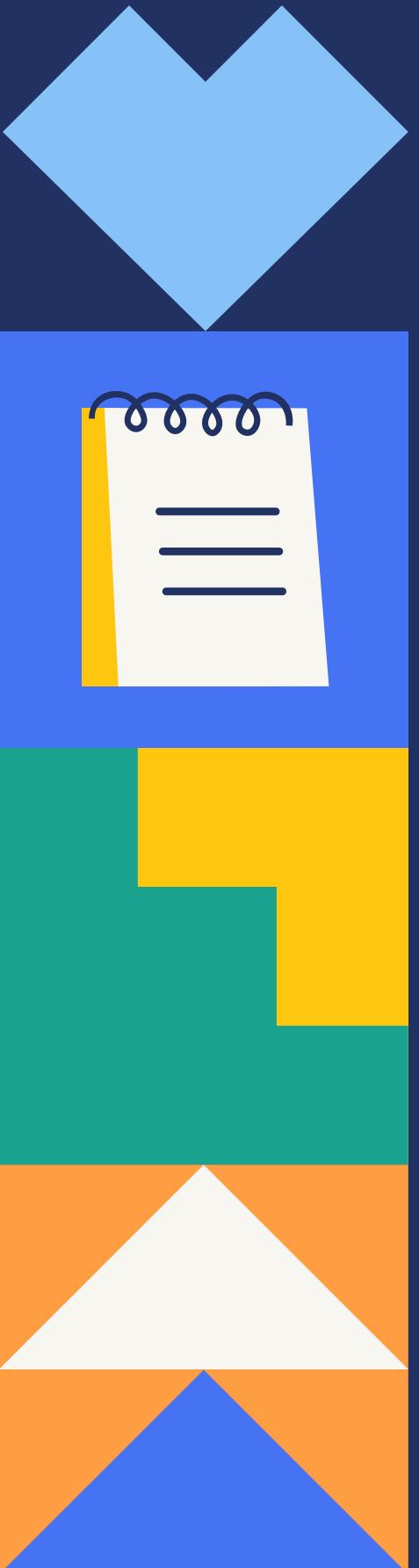


Chart Your Way to Insightful Decisions

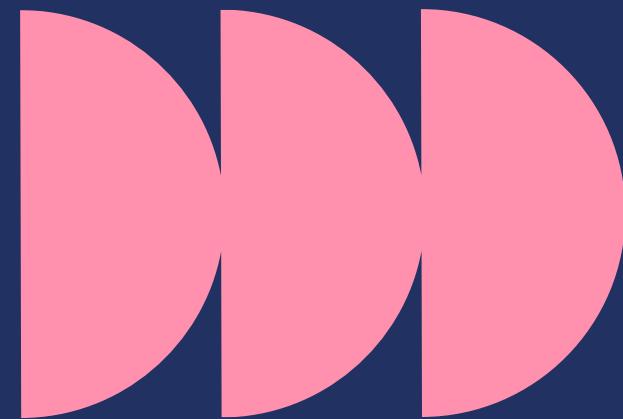
Using Data Visualisation in R for Clarity & Insight

Alexia Samiotis
PhD Candidate (Clinical Neuropsychology)
Monash Epworth Rehabilitation Research Centre





Outline



Charting

- Visualisation
- Graphics in R
- Introducing ggplot2
- Advanced graphics



Insightful decisions

- Choosing the right chart
- Good vs bad charts
- Demonstration

AlexiaSam/R-Intro

R-Intro/Epworth_Education_Graphics_in_R.Rmd at main ·
AlexiaSam/R-Intro

Contribute to AlexiaSam/R-Intro development by creating an account on GitHub.

[GitHub](#)

R Crash Course can
be found here



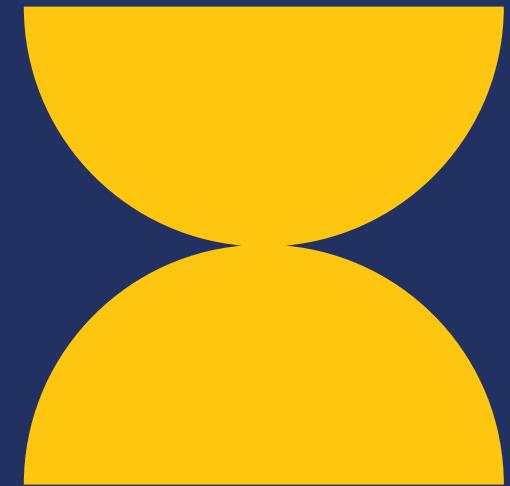
Link to code used

Charting Your Way: Visualising Psychopathology Data Using R

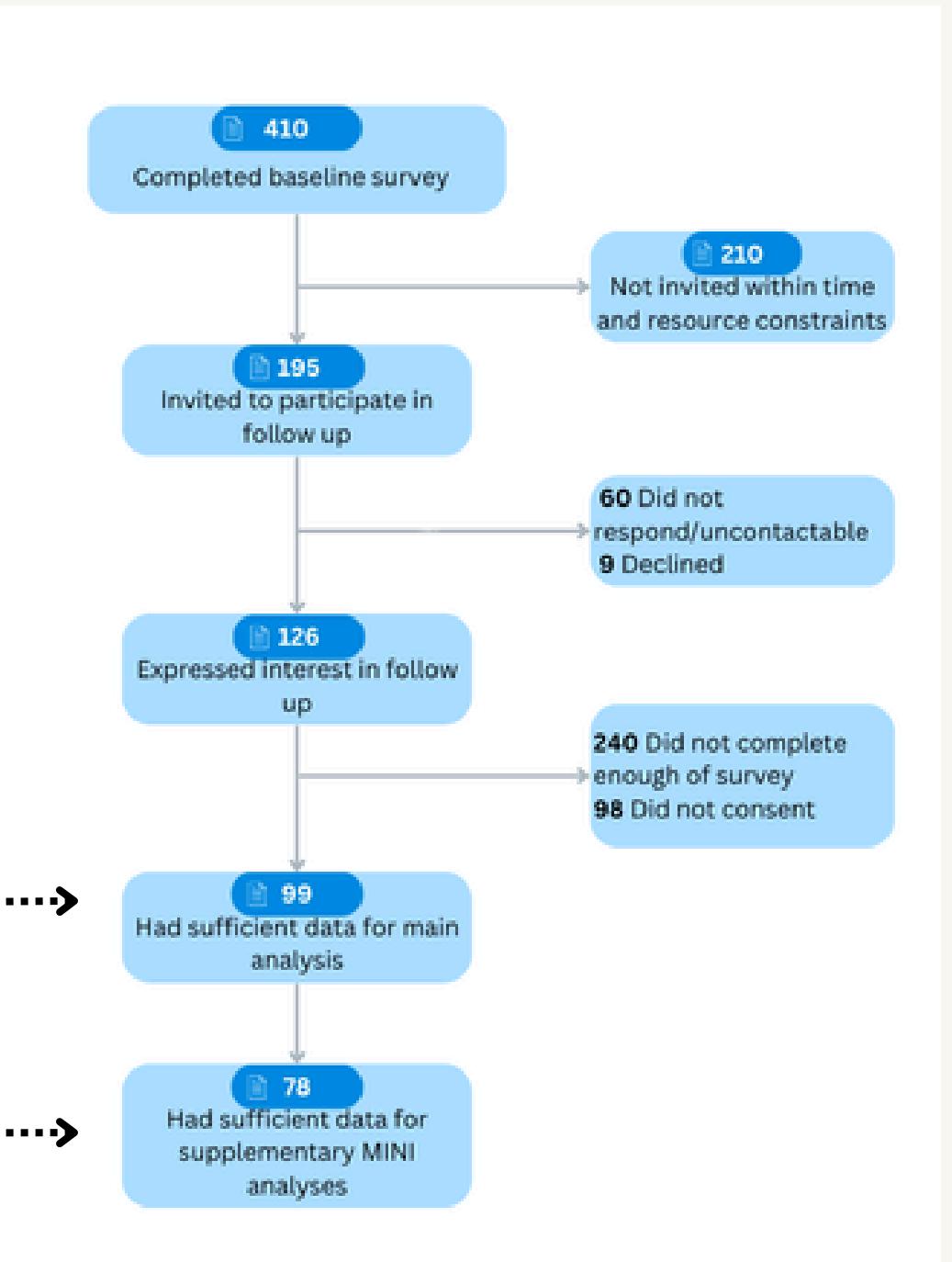
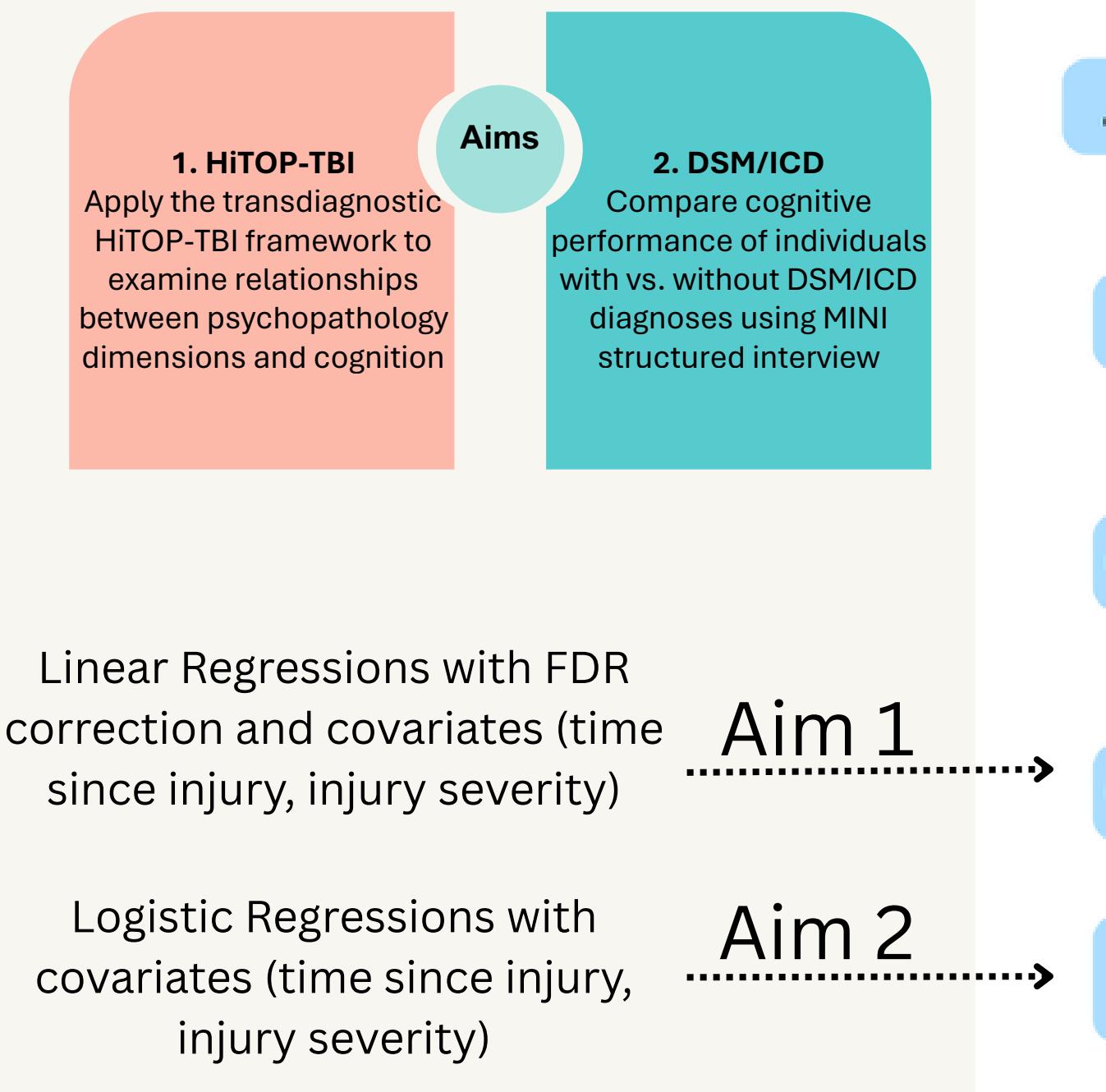


Associations between transdiagnostic psychopathology dimensions and cognitive functioning
after traumatic brain injury: An application of the HiTOP-TBI model

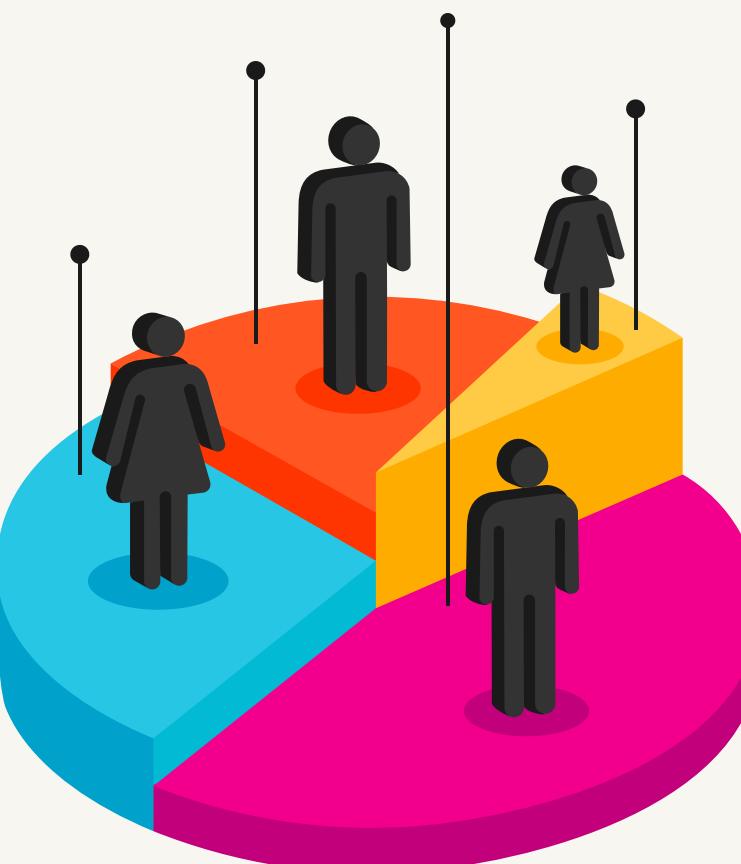
Alexia Samiotis^{*1,2}, Jai Carmichael^{1,2}, Jao-Yue Carmintati^{1,2}, Amelia J Hicks¹, Jennie
Ponsford^{1,2}, Kate Rachel Gould^{1,2}, Gershon Spitz^{1,2,3}



Charting Your Way: Quick Background to the Study and Sample

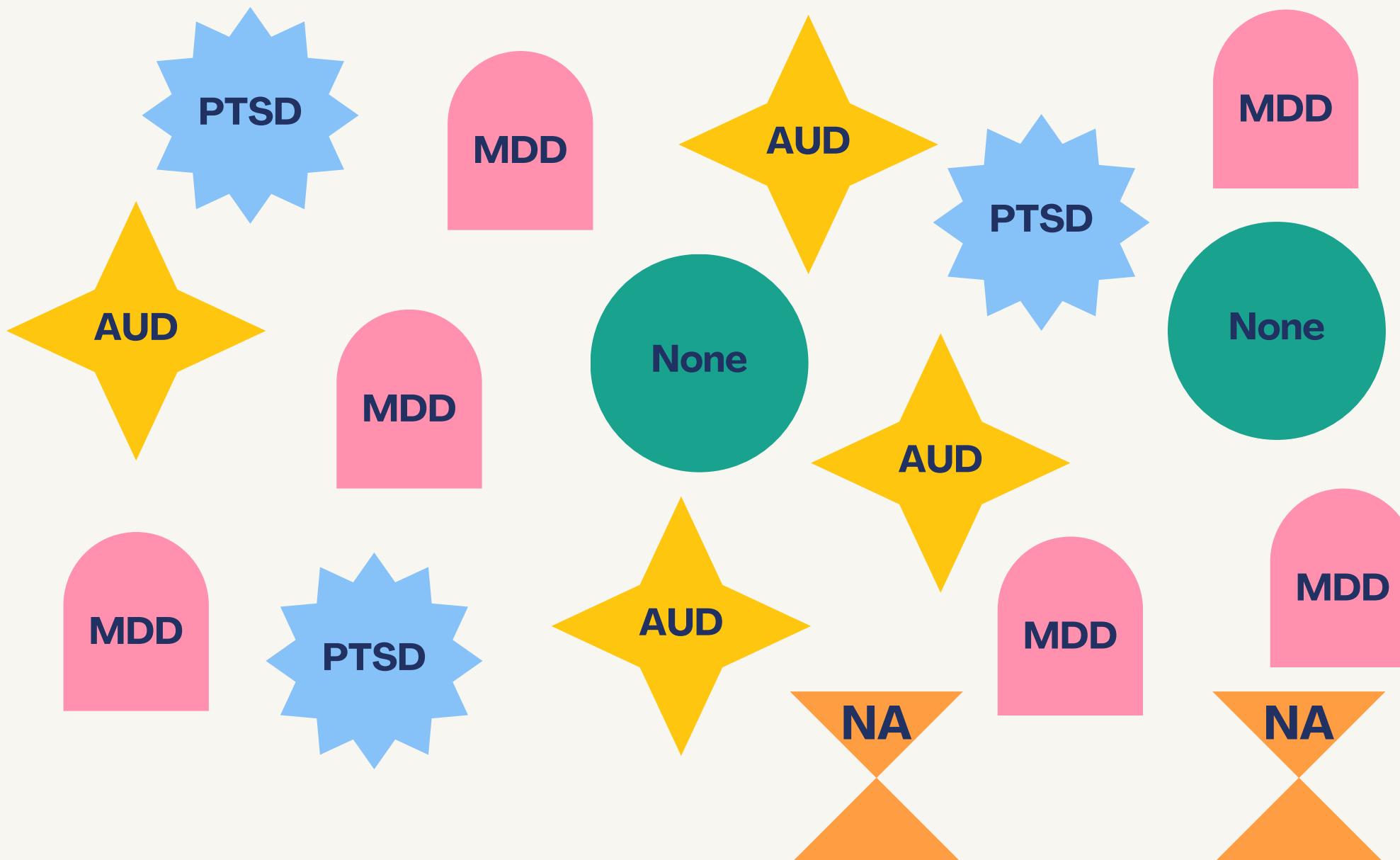


N=99
Male (73%)
White (92%)
Education = 14 years
Age = 51 years
33% Receiving mental health treatment



Charting Your Way: Why Visualisation?

Psychologists interviewed 78 individuals with TBI
and these are the diagnoses that were provided:



- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics

How can we chart the
diagnoses?

Associations between transdiagnostic psychopathology dimensions and cognitive functioning

after traumatic brain injury: An application of the HiTOP-TBI model

Alexia Samiotis^{*1,2}, Jai Carmichael^{1,2}, Jao-Yue Carmintati^{1,2}, Amelia J Hicks¹, Jennie Ponsford^{1,2}, Kate Rachel Gould^{1,2}, Gershon Spitz^{1,2,3}

Charting Your Way: Why Visualisation?

Psychologists provided 99 individuals with TBI a survey and then standardised their total score (we are neuropsychs so we love z scores).



- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics

How can we chart the survey responses against diagnoses and other variables (e.g., age)?

Associations between transdiagnostic psychopathology dimensions and cognitive functioning

after traumatic brain injury: An application of the HiTOP-TBI model

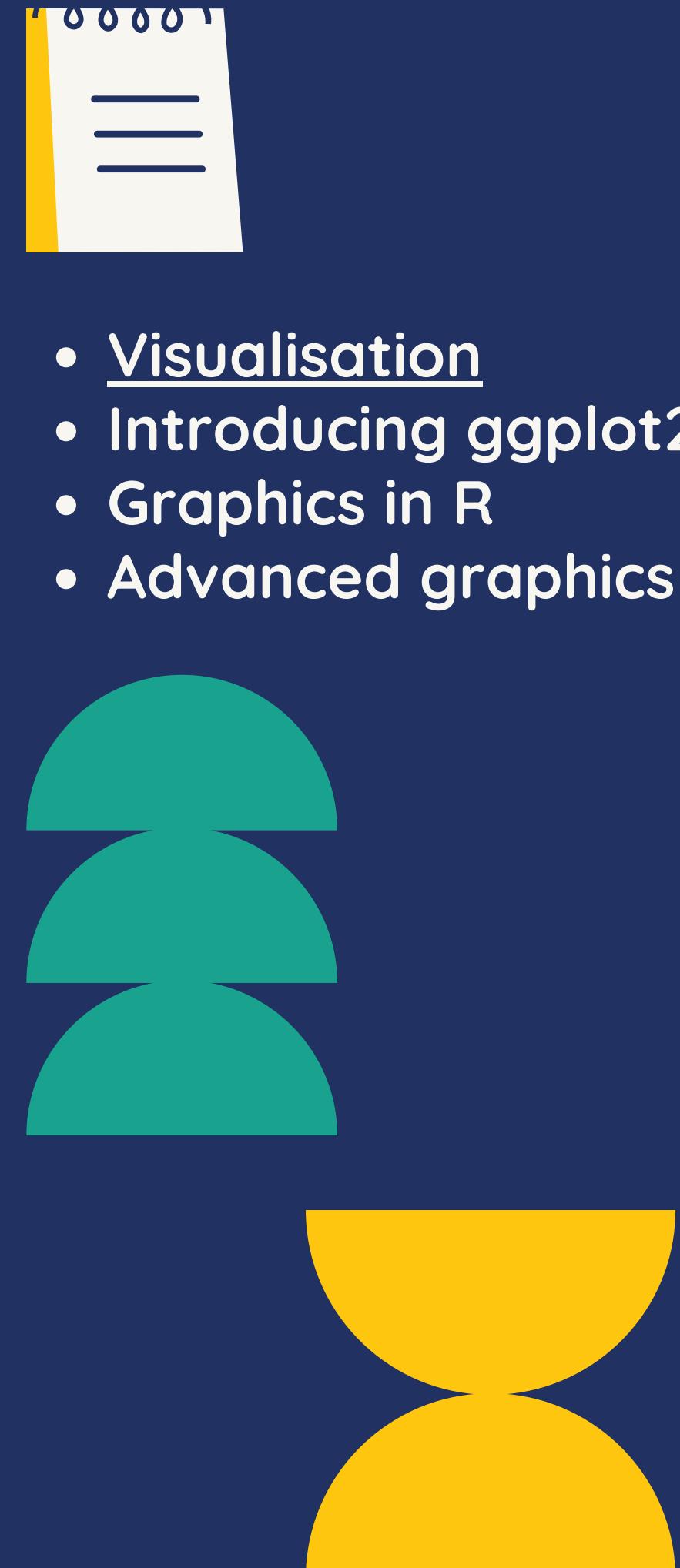
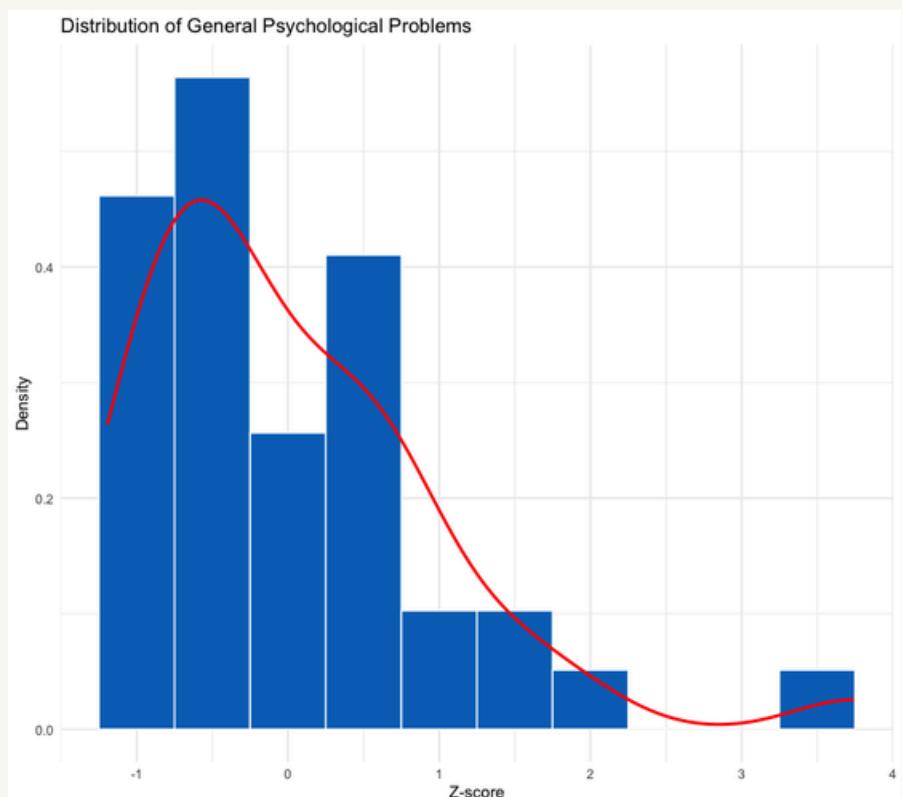
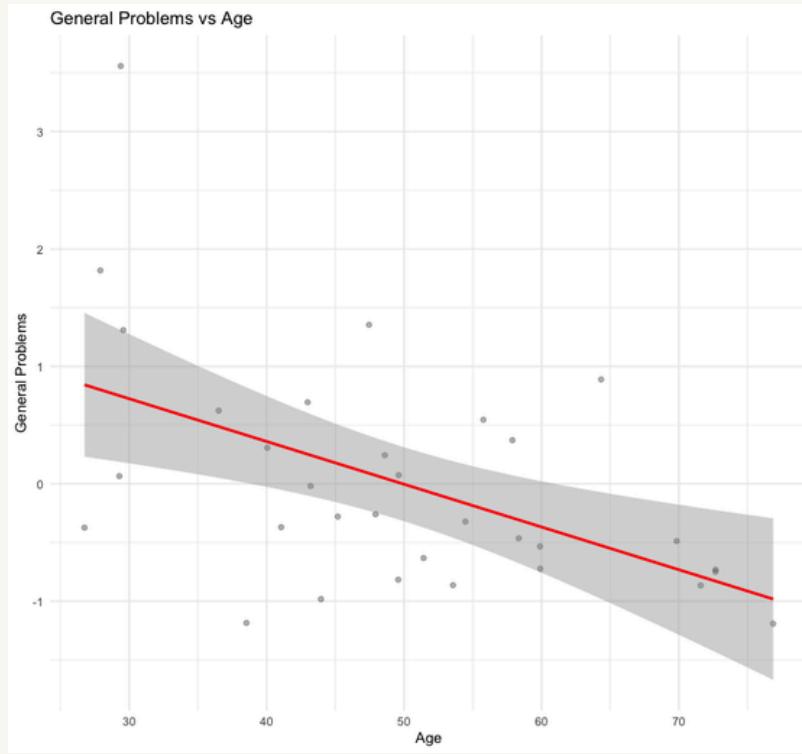
Alexia Samiotis^{*1,2}, Jai Carmichael^{1,2}, Jao-Yue Carmintati^{1,2}, Amelia J Hicks¹, Jennie Ponsford^{1,2}, Kate Rachel Gould^{1,2}, Gershon Spitz^{1,2,3}

Charting Your Way: Why Visualisation?

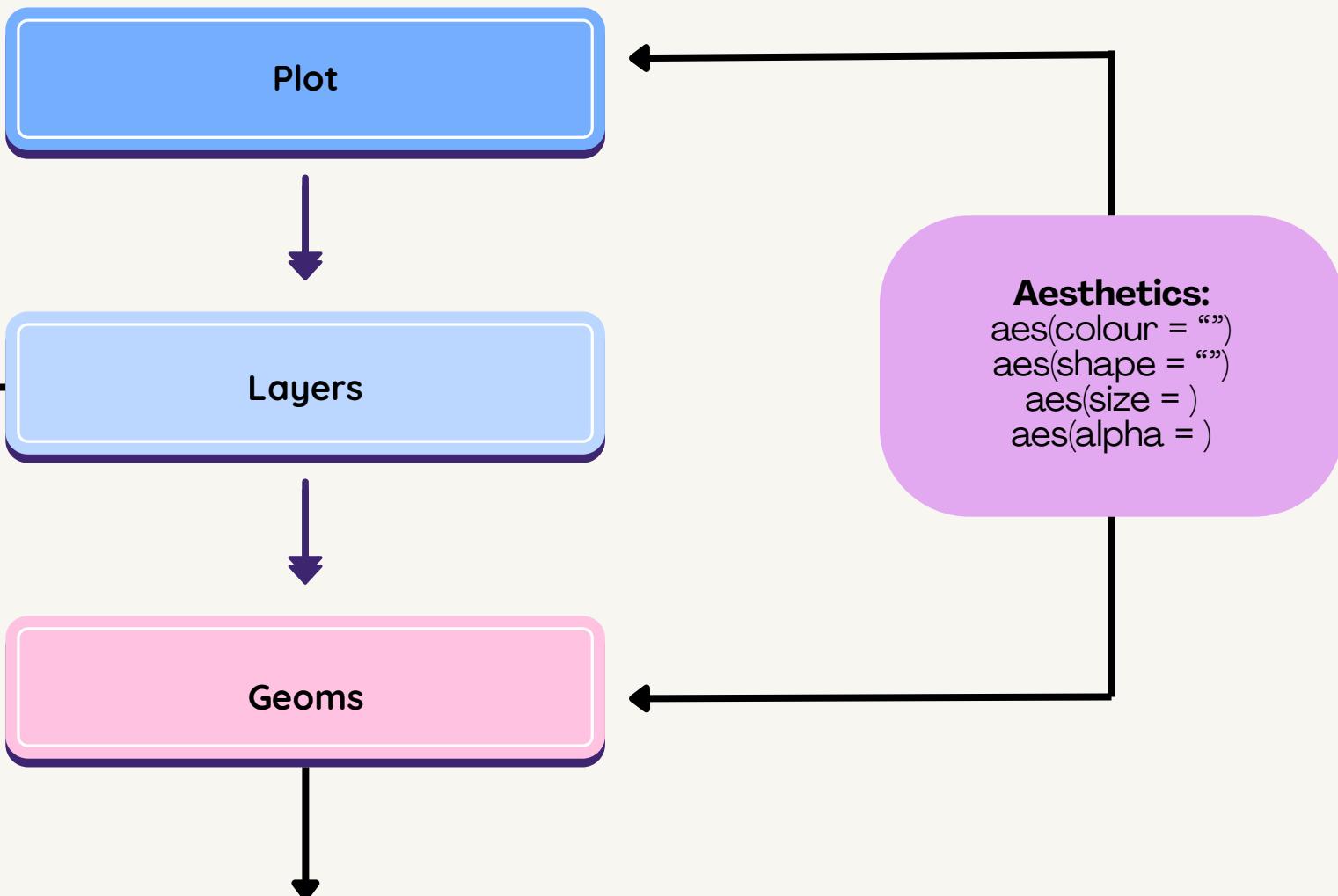
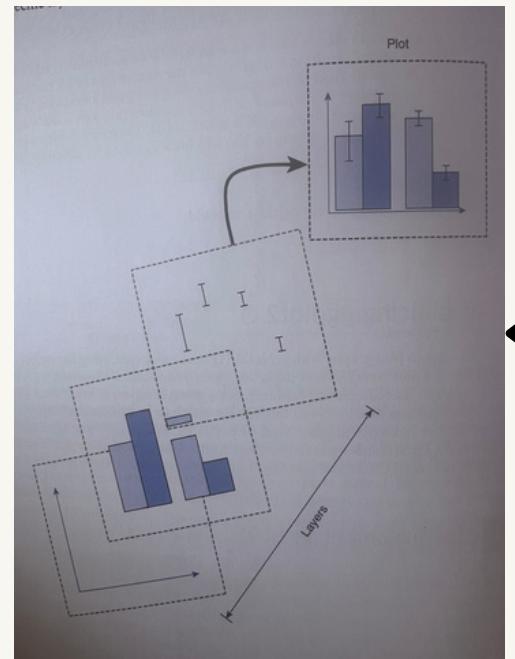
- Charts can help spot trends, outliers, and relationships quickly
- Reduce load on our cognitive processes, e.g., visual processing and memory

Which of these makes it easier to process the data?

ID	General Problems
1	-0.23
2	1.12
3	-0.45
4	0.87
5	-1.05
6	0.23
7	1.45
8	-0.78
9	0.56
10	0.02



Charting in R: Introducing ggplot2



geom_	graphic
geom_bar	creates a layer with bars
geom_point	plots the actual data points (i.e., scatterplot)
geom_line	plots a straight line that connects data points
geom_smooth	a smooth line (e.g., curvy) that summarises data overall rather than connecting individual points
geom_histogram	creates a histogram layer
geom_boxplot	creates a box-whisker diagram

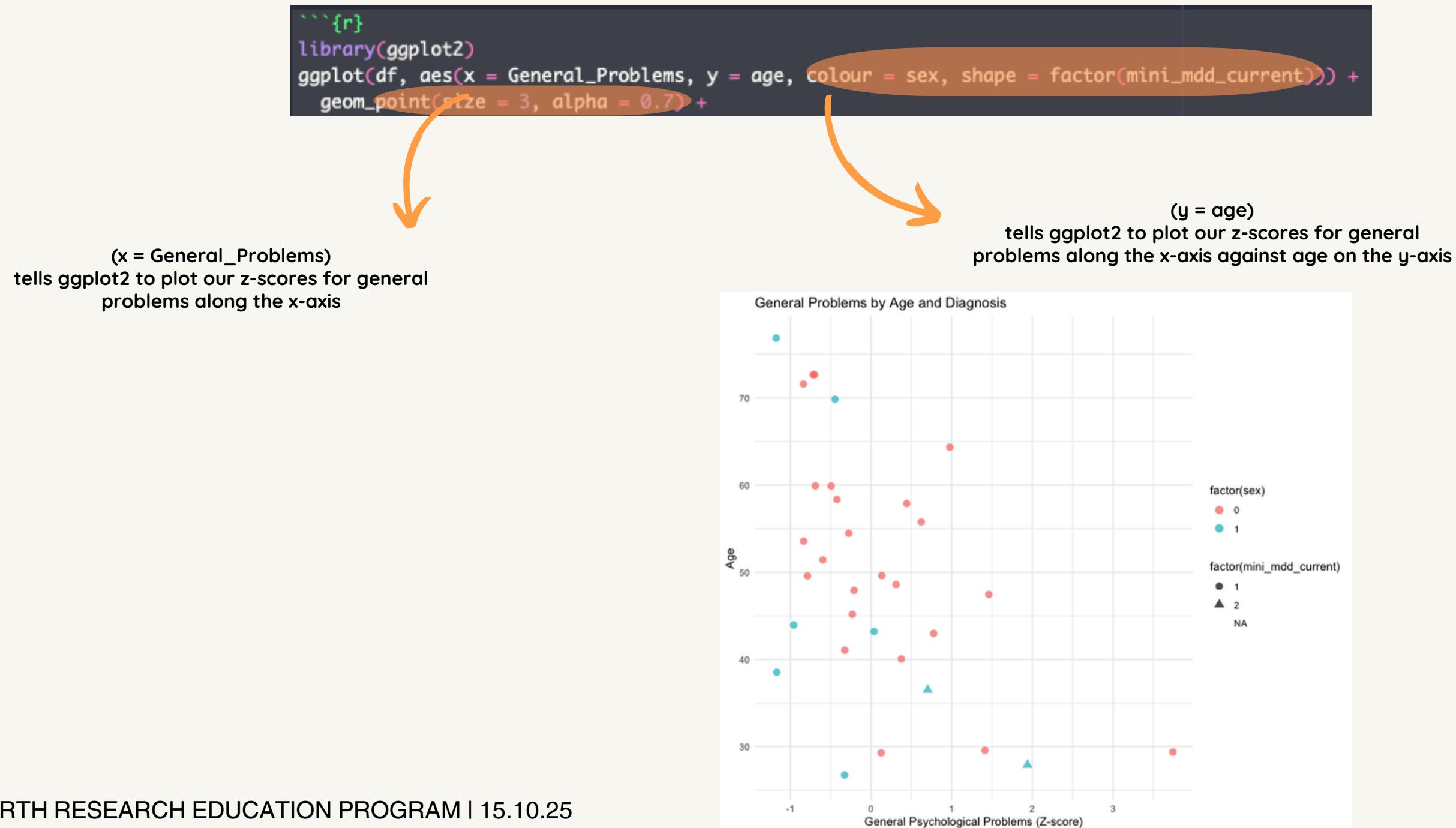


- Visualisation
- Introducing_ggplot2
- Graphics in R
- Advanced graphics



Aesthetics and Geoms with General Problems and MDD diagnoses

x and y axes

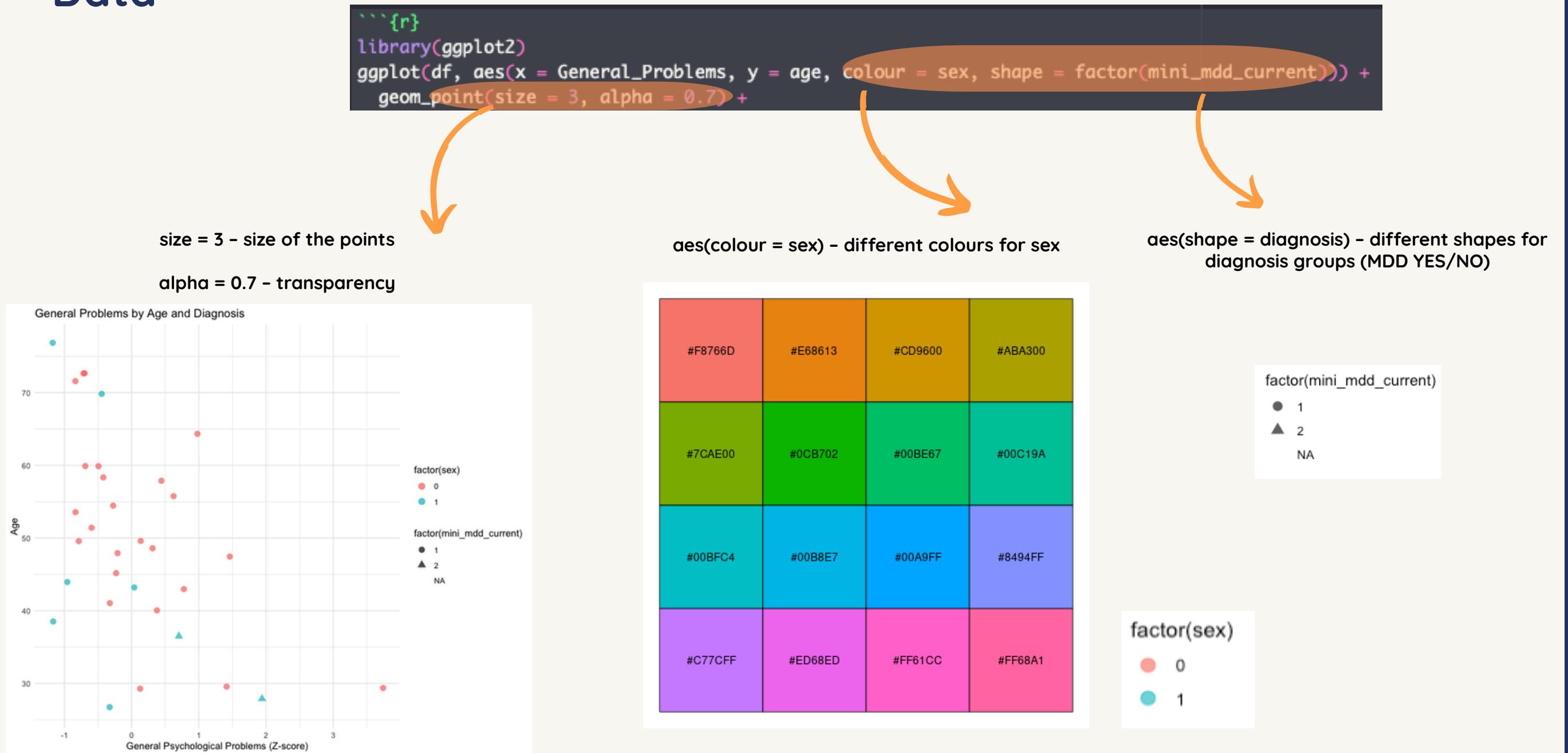


- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



Charting: Ingredients for Your Graphics

Layer 1. Aesthetics: Mapping Visual Properties to Data



Aesthetics:
aes(colour = "")
aes(shape = "")
(size =)
(alpha =)



- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



Charting: Ingredients for Your Graphics

Layer 1. Aesthetics: Adjusting these values

```
```{r}
library(ggplot2)
ggplot(df, aes(x = General_Problems, y = age, colour = sex, shape = factor(mini_mdd_current)) +
 geom_point(size = 3, alpha = 0.7) +
```

increased size from 3 to 6

increased transparency (alpha) - can you notice the difference?

aes(colour = sex) - different colours for sex

aes(shape = diagnosis) - different shapes for diagnosis groups (MDD YES/NO)

#F8766D	#E68613	#CD9600	#ABA300
#7CAE00	#0CB702	#00BE67	#00C19A
#00BFC4	#00B8E7	#00A9FF	#8494FF
#C77cff	#ED68ED	#FF61CC	#FF68A1

```
Define color palette by cognitive domain group
domain_colors <- c(
 "BTACT_Composite" = "#9e9ac8", # Purple - Global Composite
 "Episodic_Memory" = "#f768a1", # Strong pink - Verbal Memory
 "Verbal_Encoding" = "#fbba99", # Light pink - Verbal Encoding
 "Delayed_Memory" = "#e78ac3", # Mid pink - Delayed Memory
 "Executive_Function" = "#66c2a5", # Teal - EF Summary
 "Digits_Backward" = "#8dd3c7", # Aqua - EF
 "Backward_Counting" = "#a6cee3", # Light Blue - EF
 "Category_Fluency" = "#1f78b4", # Blue - EF
 "Number_Reasoning" = "#4daaf4" # Green - EF
```

You can also manually specify colours using these codes + more



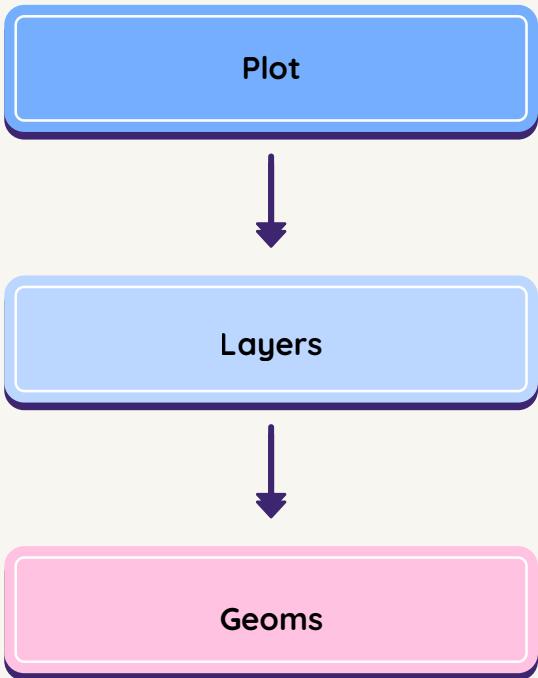
- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Ingredients for Your Graphics

## Layer 2. Geometric Objects: choosing which geom\_() to use

geom_	graphic
geom_bar	creates a layer with bars
geom_point	plots the actual data points (i.e., scatterplot)
geom_line	plots a straight line that connects data points
geom_smooth	a smooth line (e.g., curvy) that summarises data overall rather than connecting individual points
geom_histogram	creates a histogram layer
geom_boxplot	creates a box-whisker diagram
geom_density	applies a density plot layer



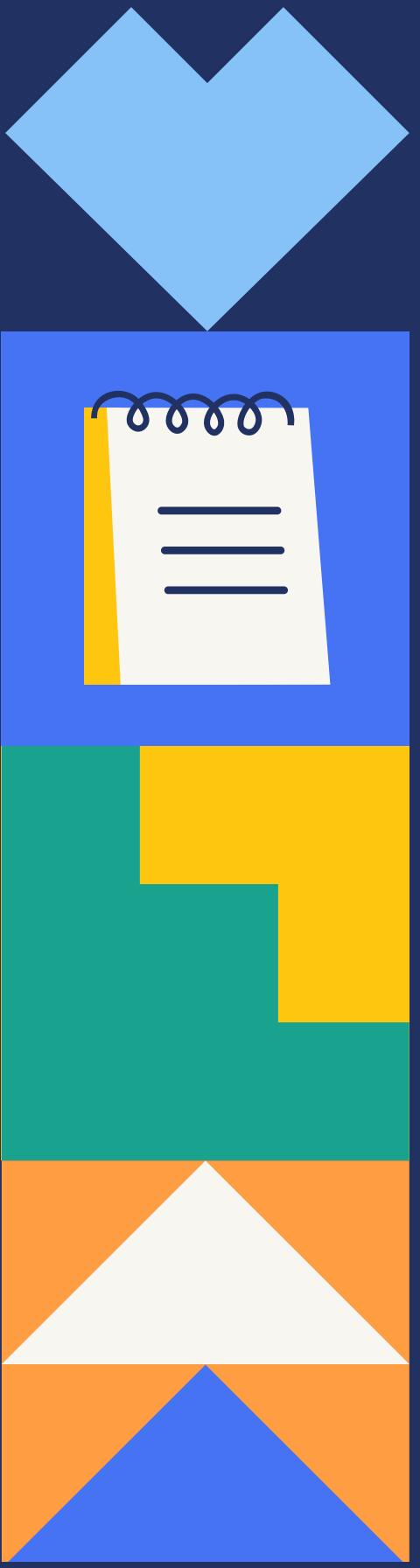
How do I know which one to use?

There are also many more of these!



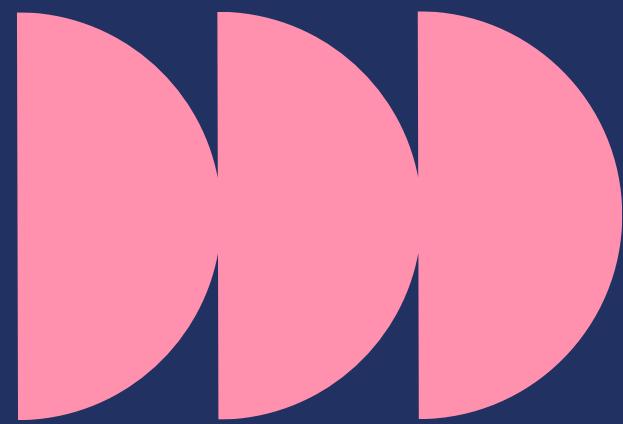
- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics





Link to code used

# Outline



## Charting

- Visualisation
- Graphics in R
- Introducing ggplot2
- Advanced graphics

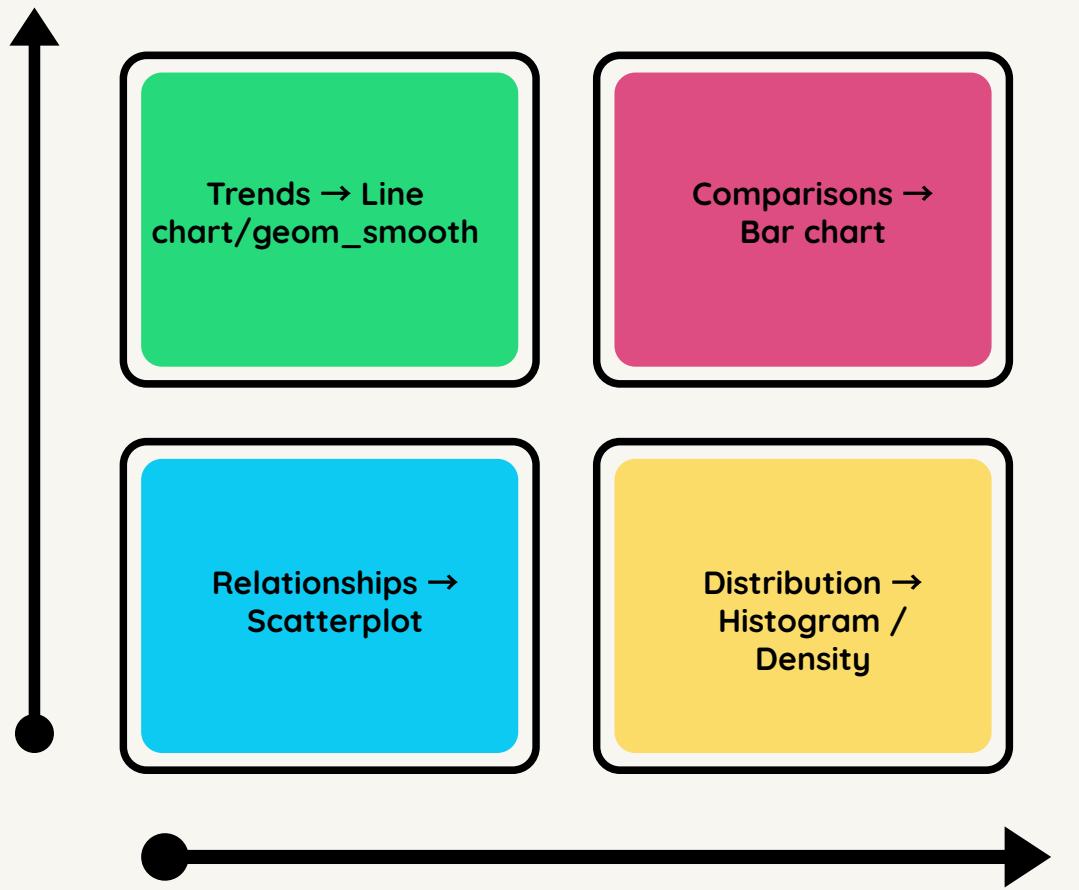


## Insightful decisions

- Choosing the right chart
- Good vs bad charts
- Demonstration



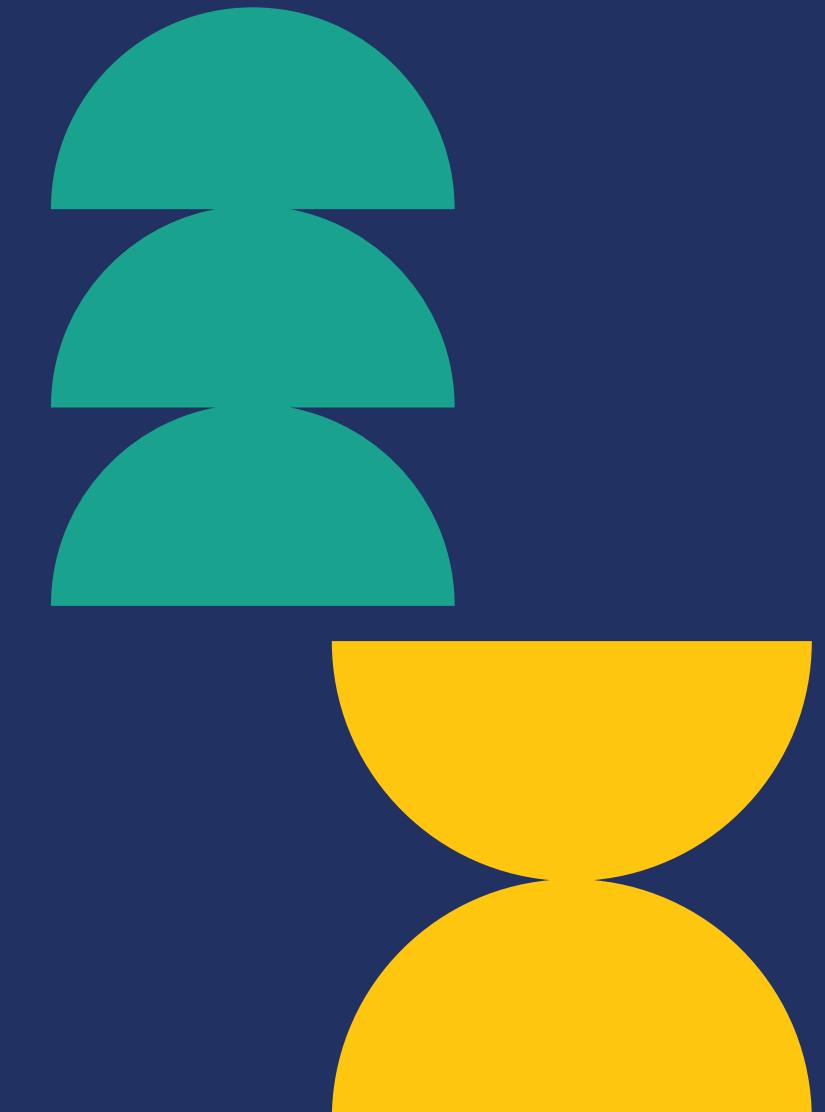
# Insightful Decisions: Choosing the Right Chart



geom_	graphic
geom_bar	creates a layer with bars
geom_point	plots the actual data points (i.e., scatterplot)
geom_line	plots a straight line that connects data points
geom_smooth	a smooth line (e.g., curvy) that summarises data overall rather than connecting individual points
geom_histogram	creates a histogram layer
geom_boxplot	creates a box-whisker diagram



- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics



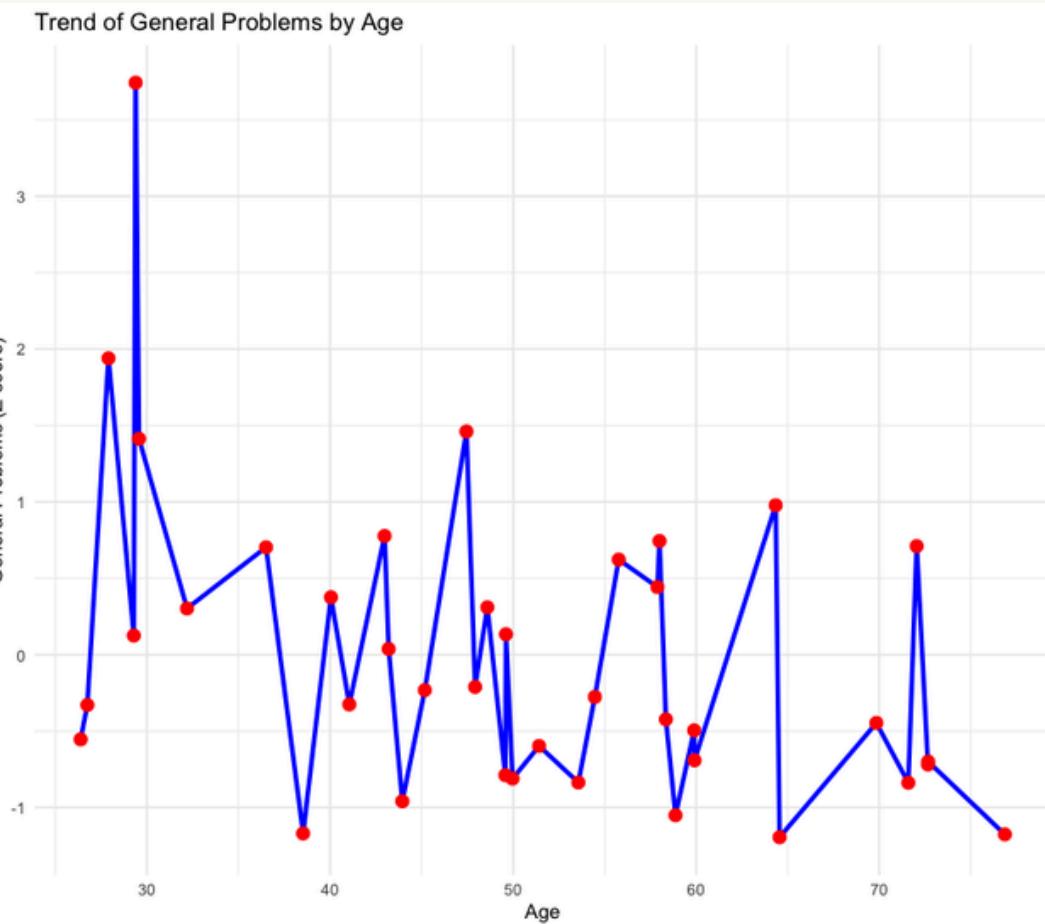
# Insightful Decisions: Examples of different charts

Trends → Line chart

Is this a good chart for presenting our psychological data?  
geom\_line



geom\_line plots a straight line that connects data points



```
```{r}
# Simple line chart example using geom_line
ggplot(df, aes(x = age, y = General_Problems)) +
  geom_line(colour = "blue", size = 1.2) + # Add a blue line
  geom_point(size = 3, colour = "red") + # Add red points on the line
  labs(title = "Trend of General Problems by Age",
       x = "Age",
       y = "General Problems (Z-score)") +
  theme_minimal()
...```

```



- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics



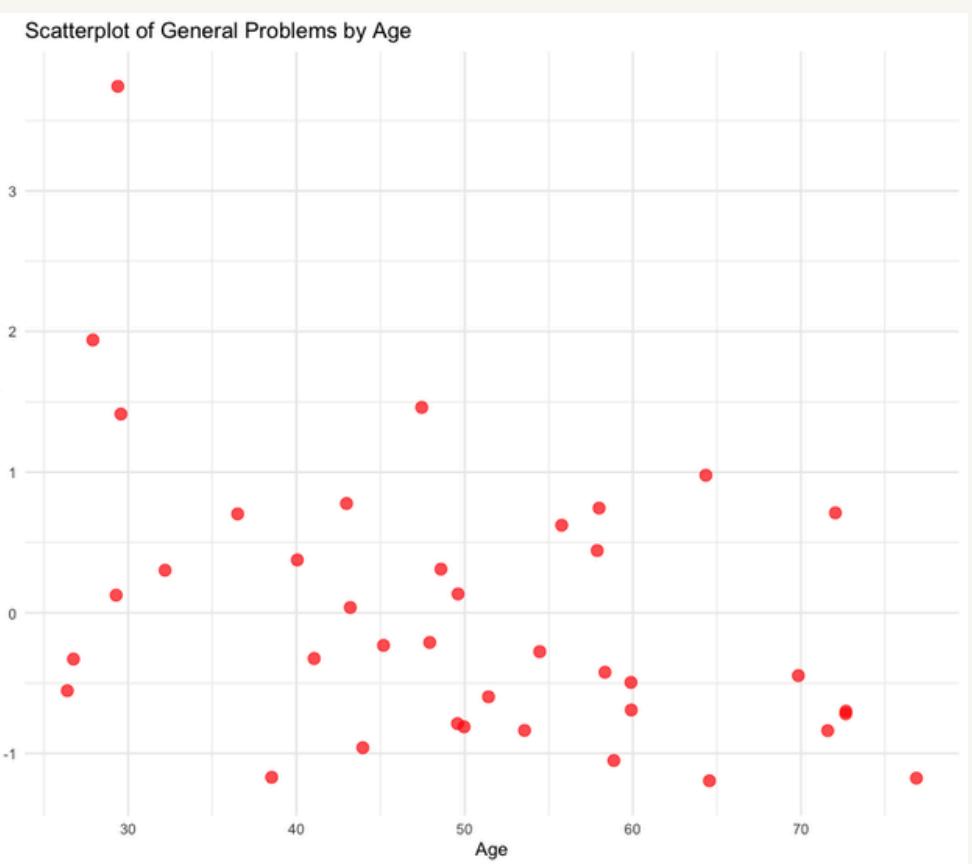
Insightful Decisions: Let's try again...

Relationships → Scatterplot

Let's just start by plotting the individual points using geom_point



geom_point plots the actual data points (i.e., scatterplot)



```
```{r}
Scatterplot example
ggplot(df, aes(x = age, y = General_Problems)) +
 geom_point(size = 3, colour = "red", alpha = 0.7) + # Red points with some transparency
 labs(title = "Scatterplot of General Problems by Age",
 x = "Age",
 y = "General Problems (Z-score)") +
 theme_minimal()
```
```



- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics



Insightful Decisions: Combining multiple geoms



Trends → Line chart/geom_smooth

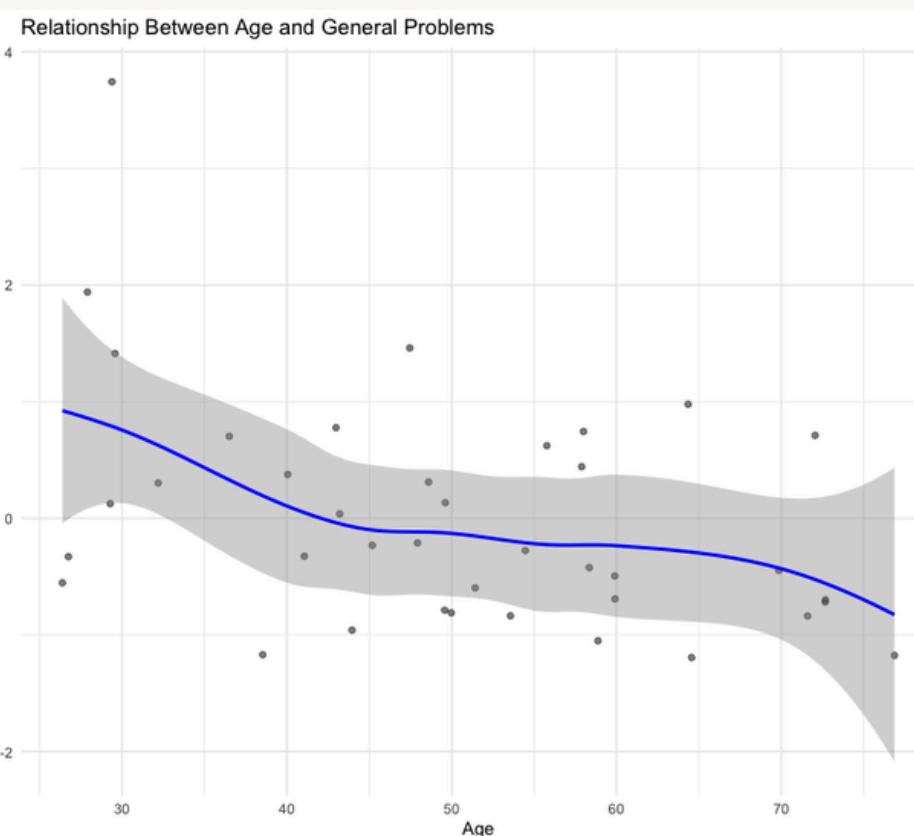
Relationships → Scatterplot

Let's combine the individual points (geom_point) with a smooth line that summarises the trend between the points (geom_smooth).

You can specify which method to use for smoothing the line. Here, I have chosen "loess".

More information found on R documentation

| | |
|-------------|---|
| geom_point | plots the actual data points (i.e., scatterplot) |
| geom_smooth | a smooth line (e.g., curvy) that summarises data overall rather than connecting individual points |



```
```{r}
ggplot(df, aes(x = age, y = General_Problems)) +
 geom_point(alpha = 0.5) +
 geom_smooth(method = "loess", colour = "blue") +
 labs(title = "Relationship Between Age and General Problems",
 x = "Age",
 y = "General Problems (Z-score)") +
 theme_minimal()
```
```

- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics



Insightful Decisions: Choosing the Right Chart



- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics

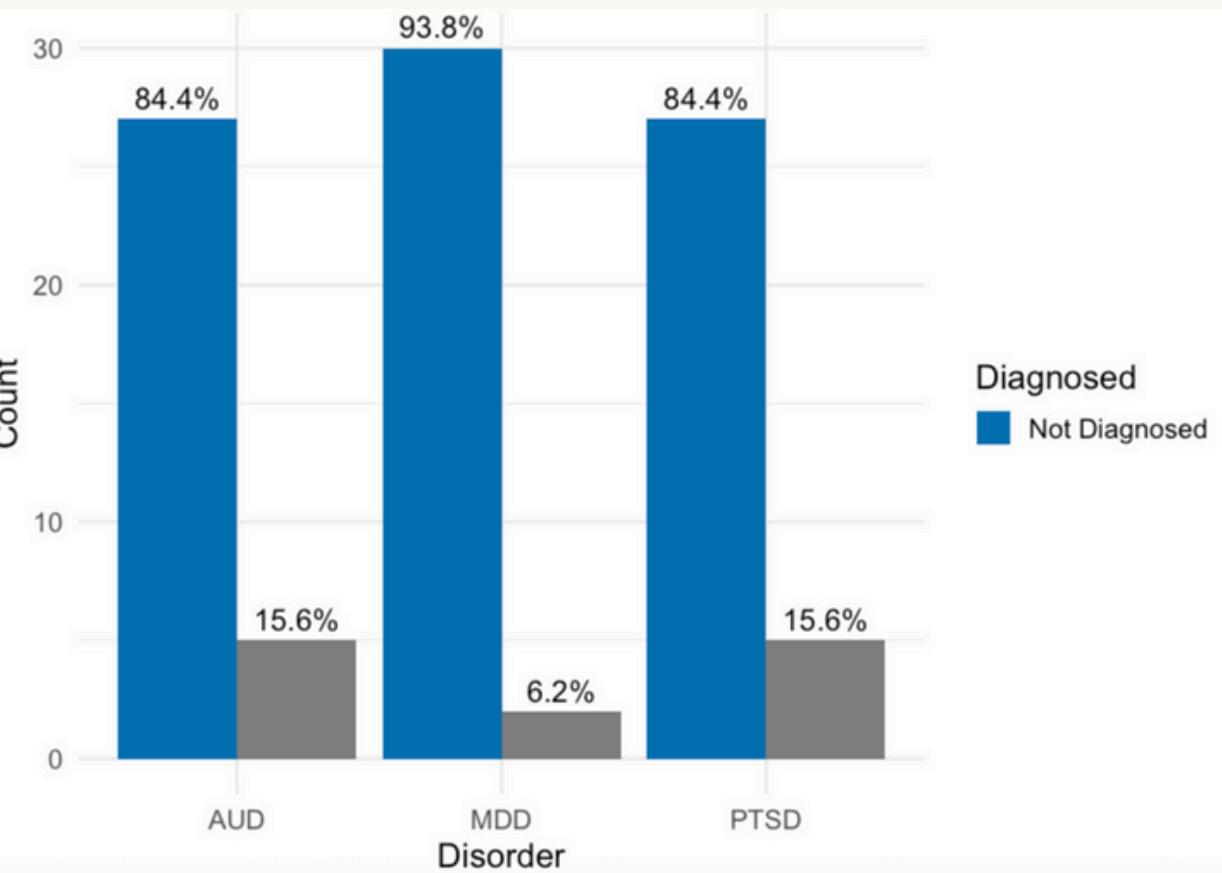
Comparisons → Bar chart

What if we wanted to compare the frequency of diagnoses across different categories?

Use geom_bar



geom_bar creates a layer with bars



```
```{r}
ggplot(plot_data, aes(x = Diagnosis, y = Count, fill = factor(Status))) +
 geom_bar(stat = "identity", position = "dodge") +
 geom_text(aes(label = Label),
 position = position_dodge(width = 0.9),
 vjust = -0.5, size = 5) +
 scale_fill_manual(values = c("0" = "#999999", "1" = "#0072B2"),
 labels = c("Not Diagnosed", "Diagnosed")) +
 labs(
 title = "Current Diagnoses (Cleaned Data)",
 subtitle = paste("Missing data removed: ", missing_rows, " of ", total_rows, " cases (",
 round(100 * missing_rows / total_rows, 1), "%)", sep = ""),
 x = "Disorder",
 y = "Count",
 fill = "Diagnosed"
) +
 theme_minimal(base_size = 16)
...```

```

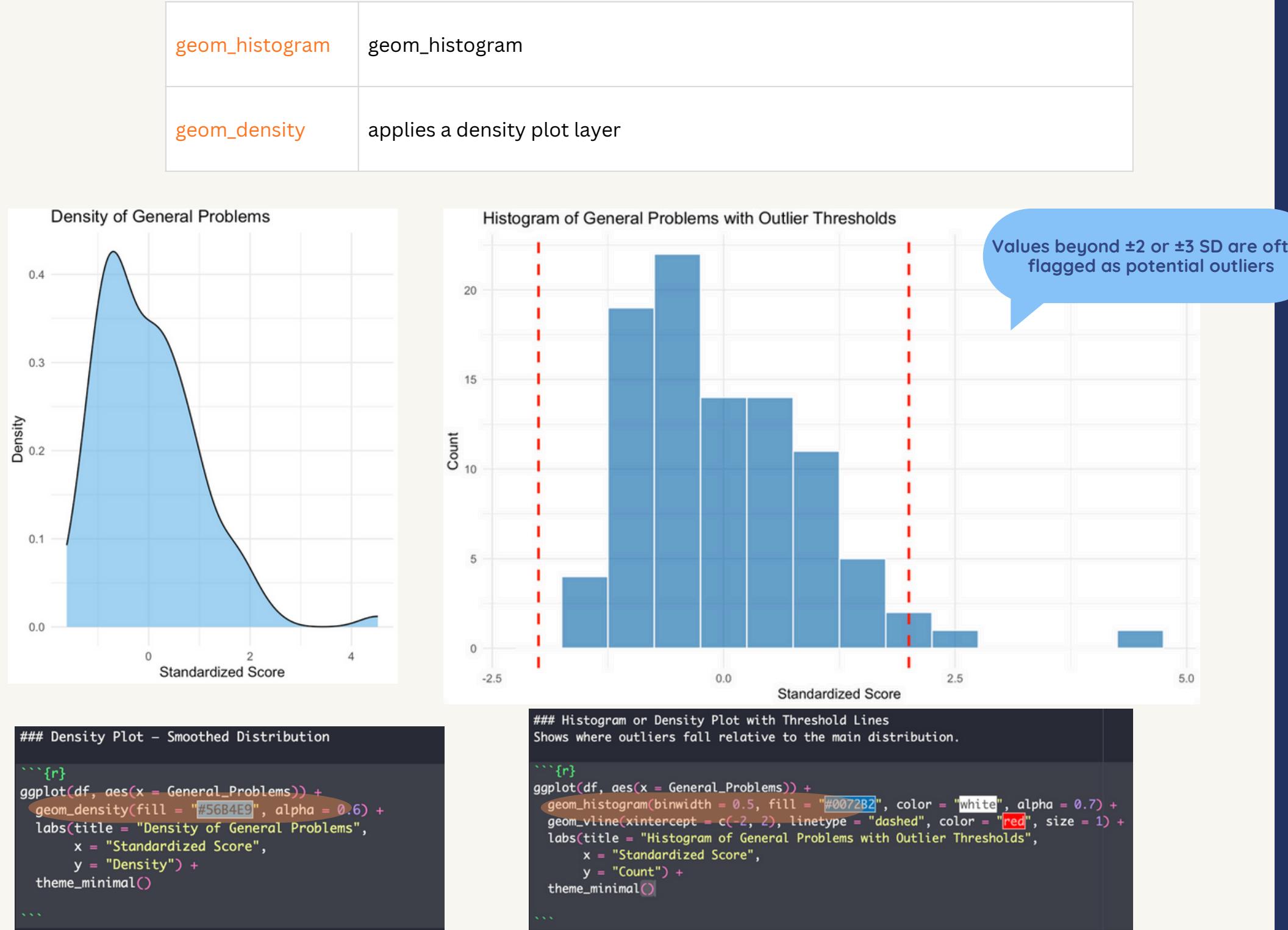


# Insightful Decisions: Choosing the Right Chart



Distribution →  
Histogram /  
Density

Displaying how the data  
is spread to help check  
for normality, outliers  
etc..



- Choosing the right chart
- Good vs bad charts
- Demonstrations
- Advanced graphics

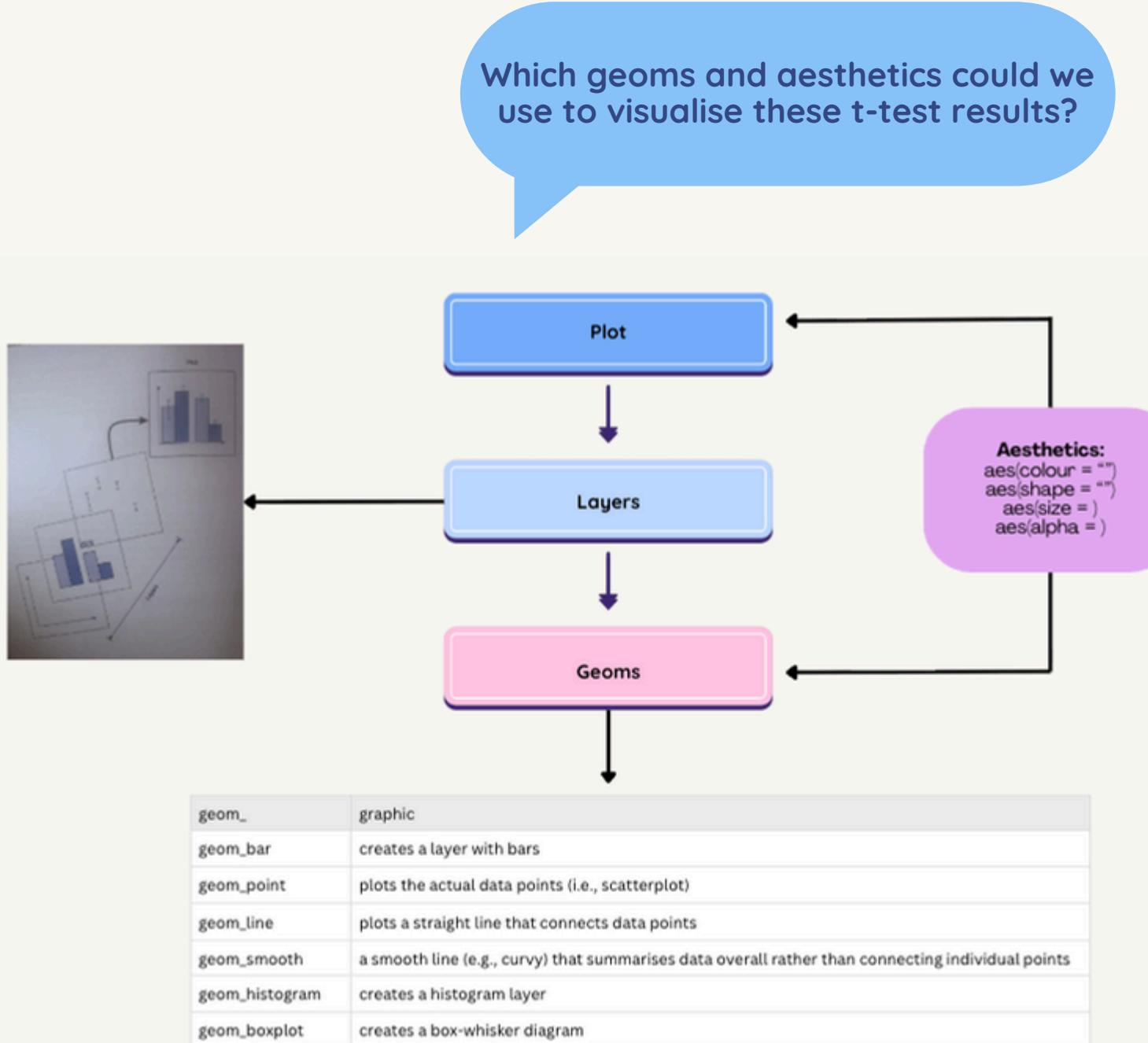


# Charting: Advanced Graphics

How does our TBI group compare to community norms on the psychopathology scales?  
Ran a series of t-tests and here are the results - not very nice to look at!

t_tests_results				
scale	t_stat	p_value	effect_size	formatted_p_value
Dysphoria	0.96	0.34	0.12	0.34
Lassitude	-1.37	0.17	-0.11	0.17
Insomnia	2.36	0.02	0.45	**0.02**
Suicidality	0.76	0.45	0.18	0.45
Appetite Loss	0.25	0.8	0.15	0.8
Appetite Gain	-1.22	0.22	-0.19	0.22
Low Well-being	2.61	0.01	0.25	**0.01**
Ill Temper	1.9	0.06	0.1	0.06
Mania	2.14	0.03	0.18	**0.03**
Euphoria	-3.63	0	-0.39	
Panic	0.83	0.41	0.07	0.41
Social Anxiety	-1.95	0.05	-0.07	0.05
Claustrophobia	0.15	0.88	0.08	0.88
Traumatic Intrusions	-0.76	0.45	0.1	0.45
Traumatic Avoidance	2.07	0.04	0.28	**0.04**
Checking	2.93	0	0.54	**0**
Ordering	2.24	0.03	0.34	**0.03**
Cleaning	2.97	0	0.15	**0**
Somatic Symptoms	18.01	0	2.65	**0**
Affective Liability	1.05	0.3	0.2	0.3
Anger	3.65	0	0.43	**0**
Anhedonia	4.21	0	0.41	**0**
Anxiousness	-1	0.32	-0.1	0.32
Callousness	-0.69	0.49	-0.12	0.49
Cognitive Problems	2.48	0.01	0.44	**0.01**
Depressiveness	1.09	0.28	0.27	0.28
Domineering	-4.17	0	-0.68	**0**
Emotional Detachment	-1.89	0.06	-0.01	0.06
Exhibitionism	-3.73	0	-0.5	**0**
Fantasy Proneness	-3.98	0	-0.48	**0**
Grandiosity	-5.18	0	-0.52	**0**
Health Anxiety	-0.68	0.5	-0.06	0.5
Hostile Aggression	-1.61	0.11	-0.11	0.11

Which geoms and aesthetics could we use to visualise these t-test results?



- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Advanced Graphics

How does our TBI group compare to community norms on the psychopathology scales?  
Ran a series of t-tests and here are the results - not very nice to look at!

t_tests_results				
scale	t_stat	p_value	effect_size	formatted_p_value
Dysphoria	0.96	0.34	0.12	0.34
Lassitude	-1.37	0.17	-0.11	0.17
Insomnia	2.36	0.02	0.45	**0.02**
Suicidality	0.76	0.45	0.18	0.45
Appetite Loss	0.25	0.8	0.15	0.8
Appetite Gain	-1.22	0.22	-0.19	0.22
Low Well-being	2.61	0.01	0.25	**0.01**
Ill Temper	1.9	0.06	0.1	0.06
Mania	2.14	0.03	0.18	**0.03**
Euphoria	-3.63	0	-0.39	
Panic	0.83	0.41	0.07	0.41
Social Anxiety	-1.95	0.05	-0.07	0.05
Claustrophobia	0.15	0.88	0.08	0.88
Traumatic Intrusions	-0.76	0.45	0.1	0.45
Traumatic Avoidance	2.07	0.04	0.28	**0.04**
Checking	2.93	0	0.54	**0**
Ordering	2.24	0.03	0.34	**0.03**
Cleaning	2.97	0	0.15	**0**
Somatic Symptoms	18.01	0	2.65	**0**
Affective Liability	1.05	0.3	0.2	0.3
Anger	3.65	0	0.43	**0**
Anhedonia	4.21	0	0.41	**0**
Anxiousness	-1	0.32	-0.1	0.32
Callousness	-0.69	0.49	-0.12	0.49
Cognitive Problems	2.48	0.01	0.44	**0.01**
Depressiveness	1.09	0.28	0.27	0.28
Domineering	-4.17	0	-0.68	**0**
Emotional Detachment	-1.89	0.06	-0.01	0.06
Exhibitionism	-3.73	0	-0.5	**0**
Fantasy Proneness	-3.98	0	-0.48	**0**
Grandiosity	-5.18	0	-0.52	**0**
Health Anxiety	-0.68	0.5	-0.06	0.5
Hostile Aggression	-1.61	0.11	-0.11	0.11

```
Plot
norms_comp <- ggplot(t_tests_results, aes(x = effect_size, y = scale, fill = effect_size_color)) +
 geom_bar(stat = "identity", show.legend = FALSE) +
 scale_fill_manual(values = c("red" = "red", "blue" = "blue", "grey" = "grey")) +
 geom_vline(xintercept = 0, linetype = "dashed", color = "black") +
 labs(
 x = "Cohen's d",
 y = NULL
) +
 geom_text(aes(label = formatted_p_value, x = effect_size + 0.1),
 hjust = 0, size = 3.5, color = "black") +
 theme_minimal() +
 theme(
 axis.text.y = element_text(size = 10),
 axis.title.x = element_text(size = 12),
 plot.title = element_text(size = 14, face = "bold"),
 plot.subtitle = element_text(size = 10)
)
ggsave("norms_comp_plot.pdf", plot = norms_comp, width = 8, height = 6)

norms_comp
```

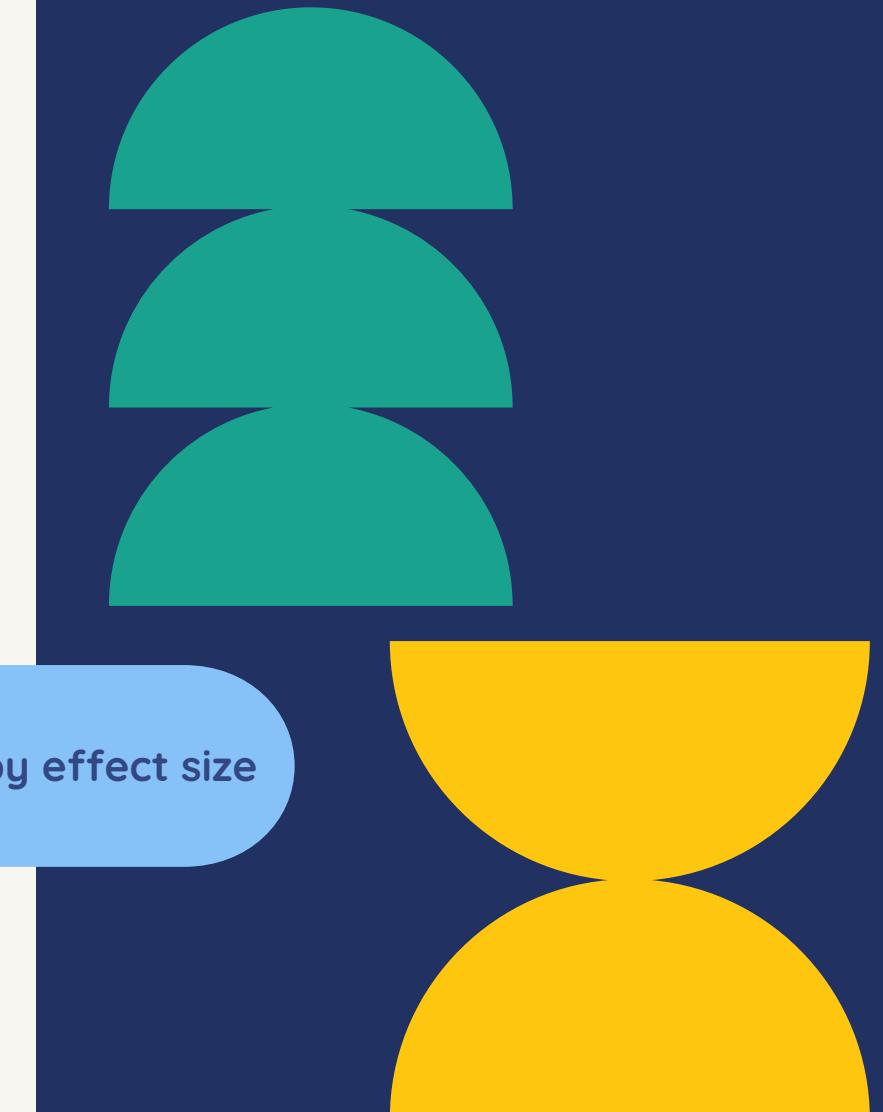
How does this look?

geom\_bar, geom\_text, different colours for significant/non-significant t-test results, and different colours for positive/negative t-stat

Order the results by effect size



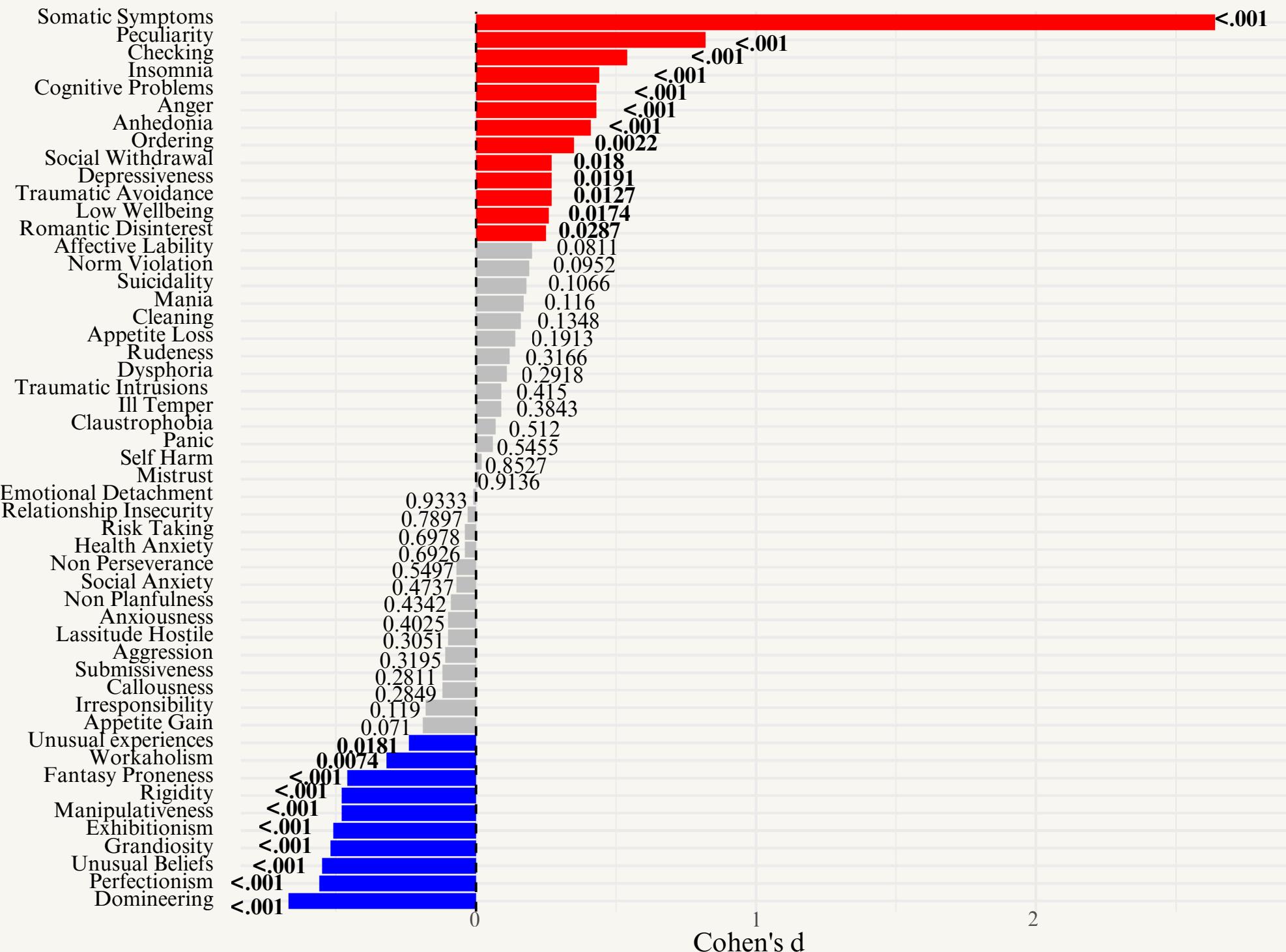
- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Advanced Graphics

How does our TBI group compare to community norms on the psychopathology scales?

Red = higher in the TBI group



Blue = lower in the TBI group

What can you take away from this?

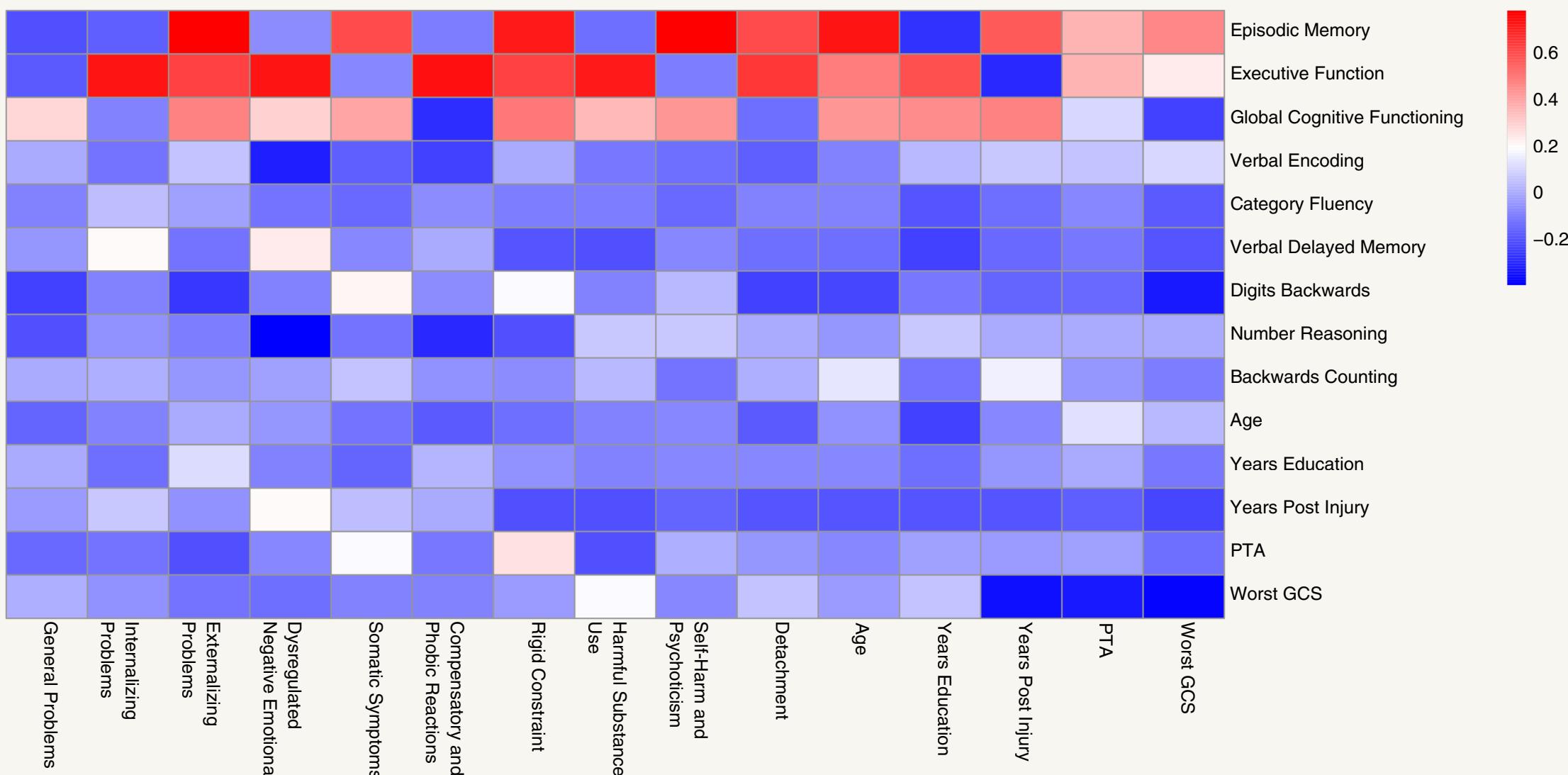


- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Advanced Graphics

Simultaneous charting of all variables of interest: Introducing Heatmaps - package - pheatmap



```
Load the pheatmap package
library(pheatmap)

Plot the heatmap using pheatmap
pheatmap(cor_matrix_rounded,
 scale = "none", # Do not scale the data
 cluster_rows = FALSE, # Cluster the rows (BTact variables)
 cluster_cols = FALSE, # Cluster the columns (HiTOP variables)
 color = colorRampPalette(c("blue", "white", "red"))(100), # Custom color palette
 main = "Heatmap of Correlations between BTACT Cognitive Domains and HiTOP Dimensions and Covariates")
...
```

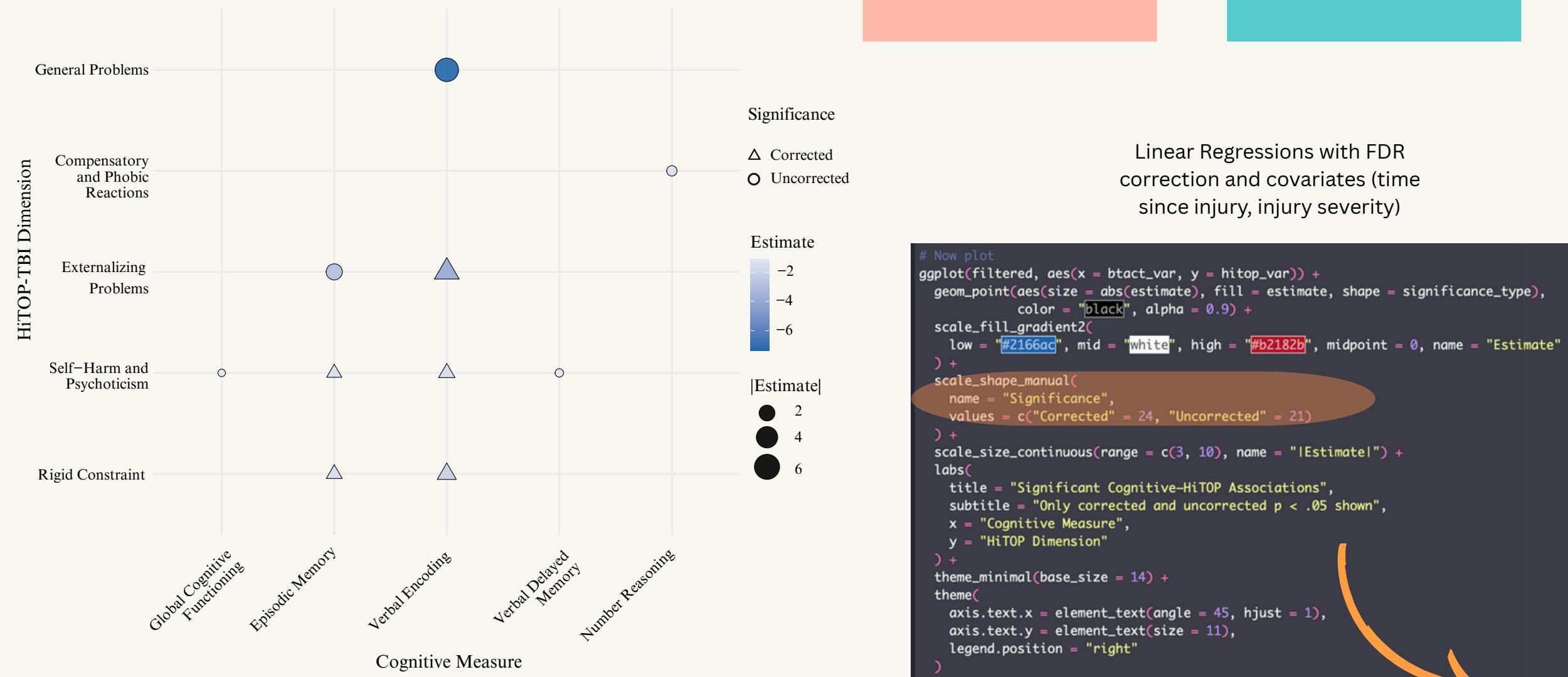


- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Advanced Graphics

Bubble Plots: Summarising Many Results (e.g., of several regressions), here I want to display only significant results.

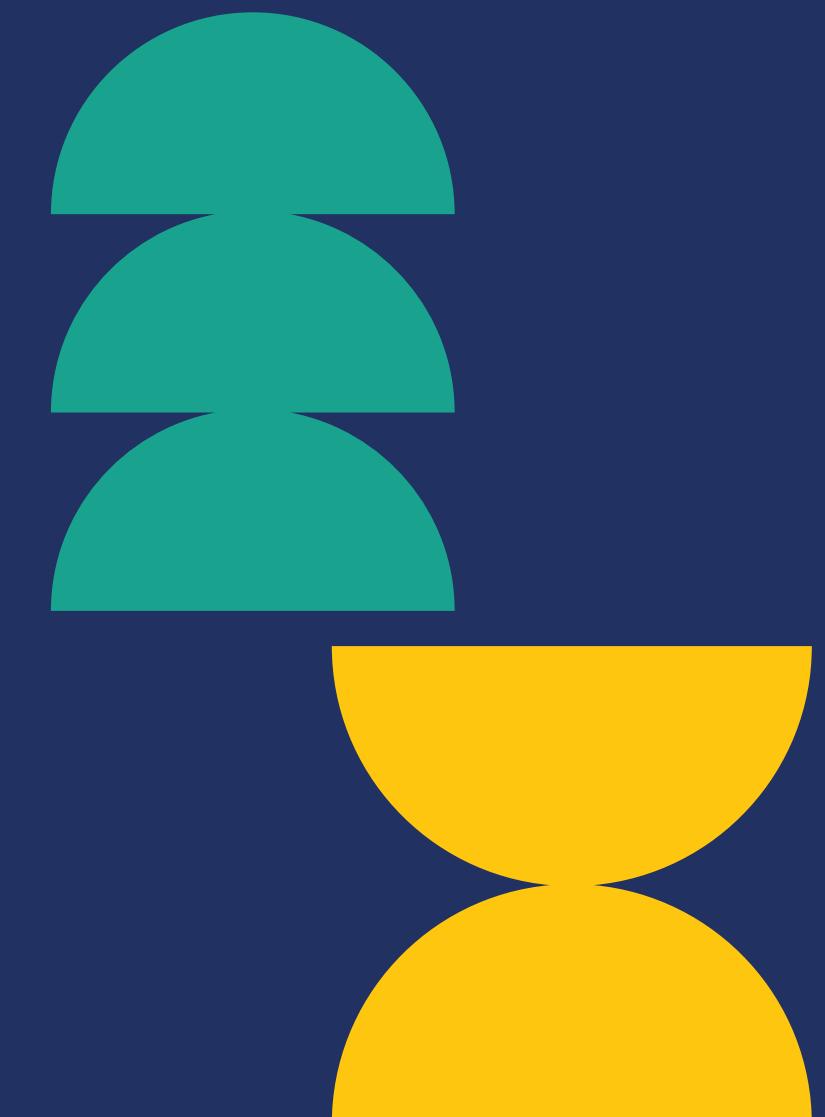


## Aims

1. HiTOP-TBI  
Apply the transdiagnostic HiTOP-TBI framework to examine relationships between psychopathology dimensions and cognition



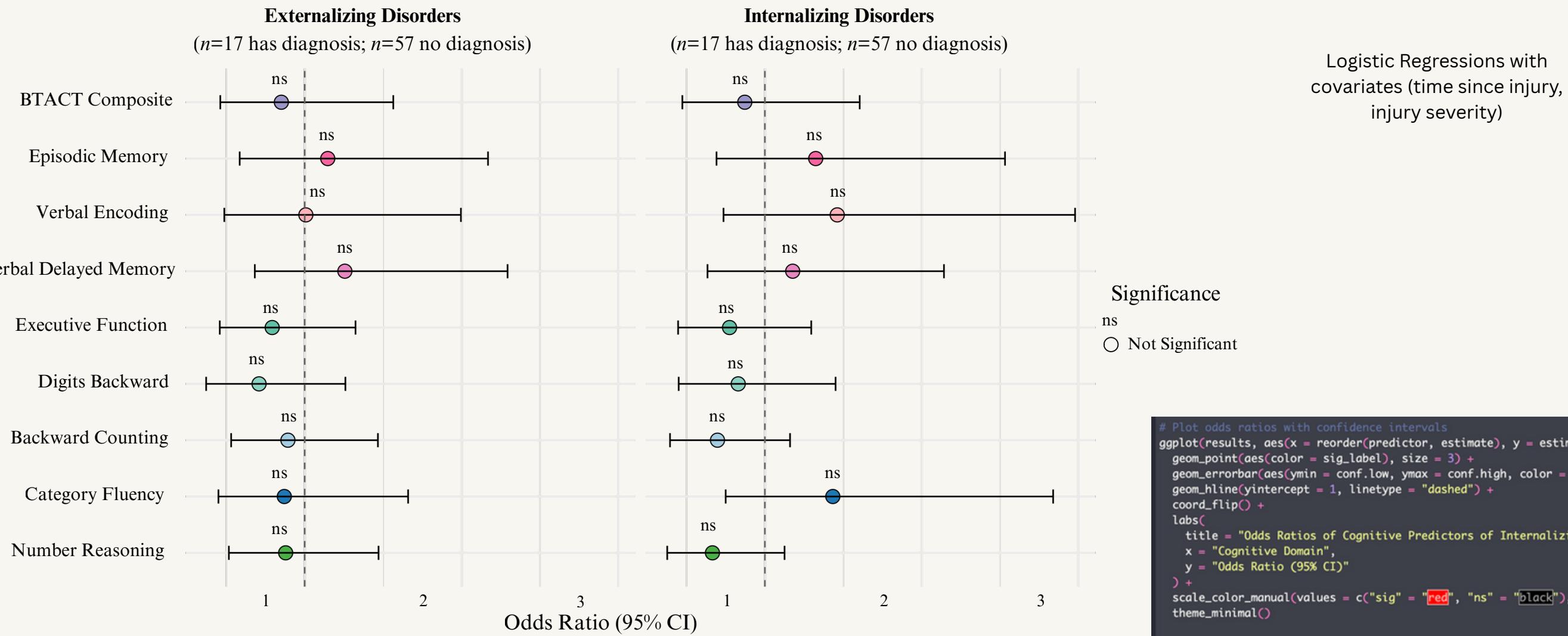
- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



Here, we are applying the shape aesthetic to visually distinguish corrected and uncorrected regression estimates. This is a strong way to summarise the data for the reader!

# Charting: Advanced Graphics

Lollipop Plots: Summarising Many Results (e.g., of several regressions)



Aims

## 2. DSM/ICD

Compare cognitive performance of individuals with vs. without DSM/ICD diagnoses using MINI structured interview

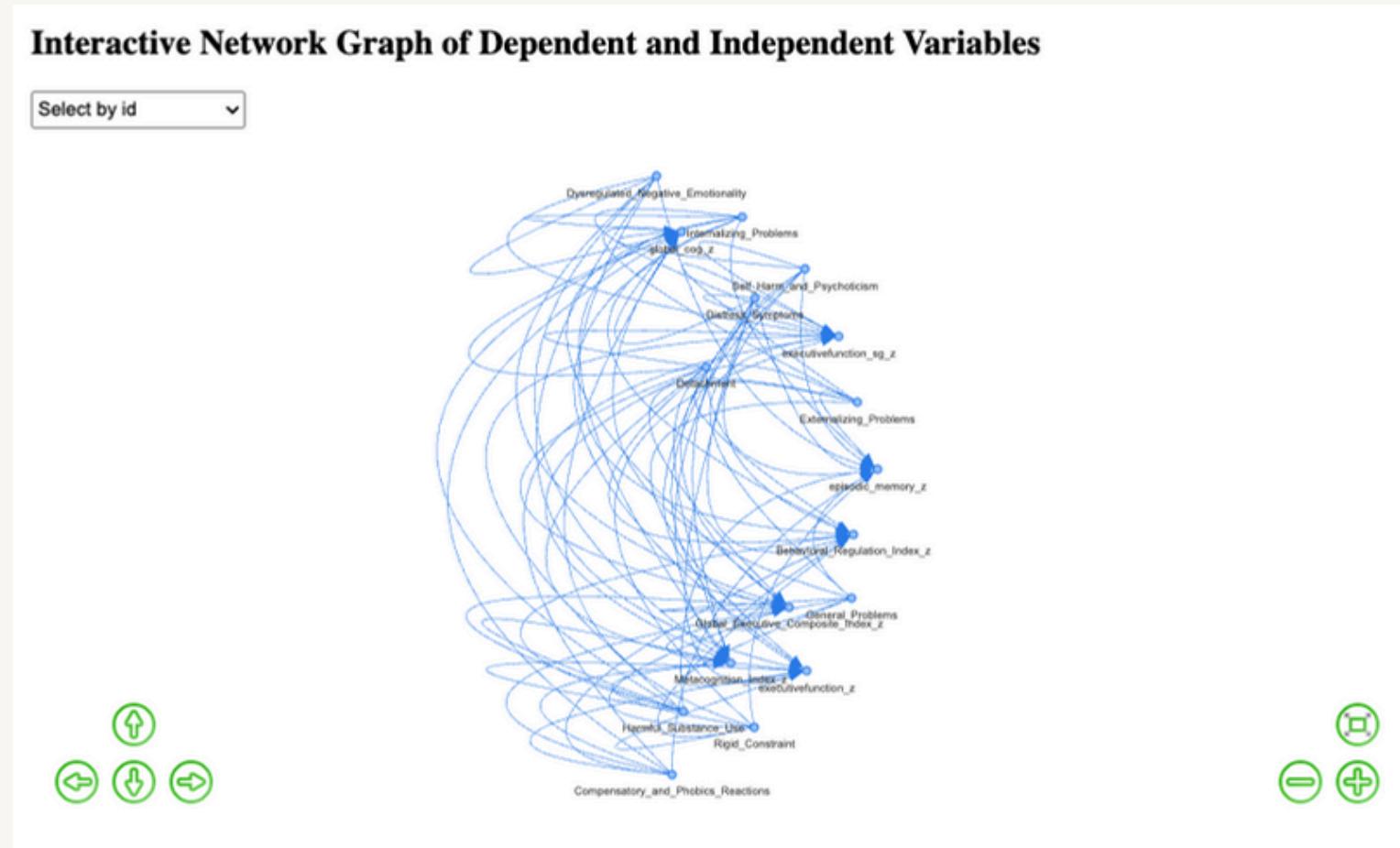
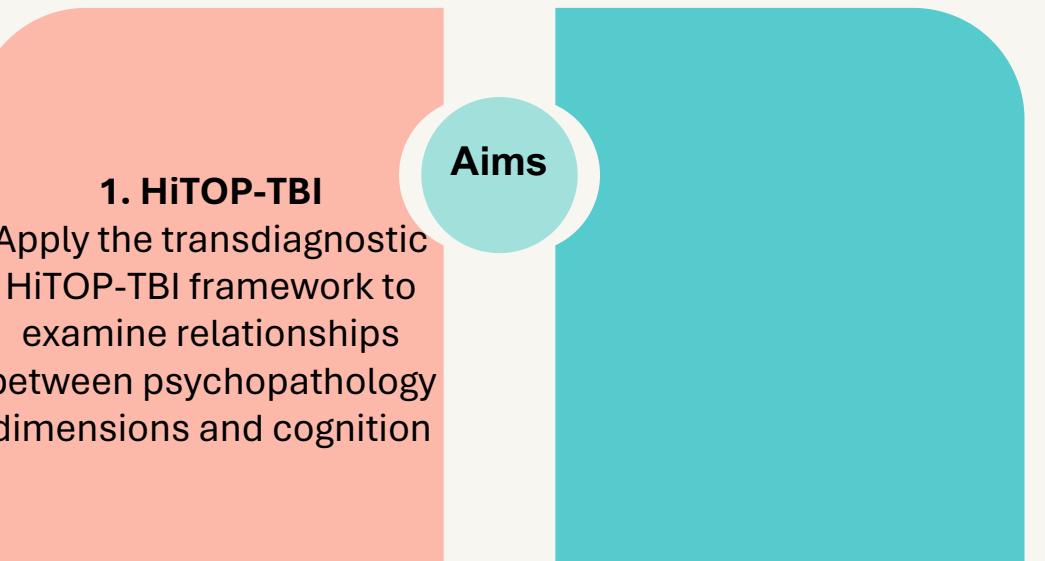


- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics

# Charting: Advanced Graphics

## Interactive Plots: VisNetwork

<https://datastorm-open.github.io/visNetwork/>

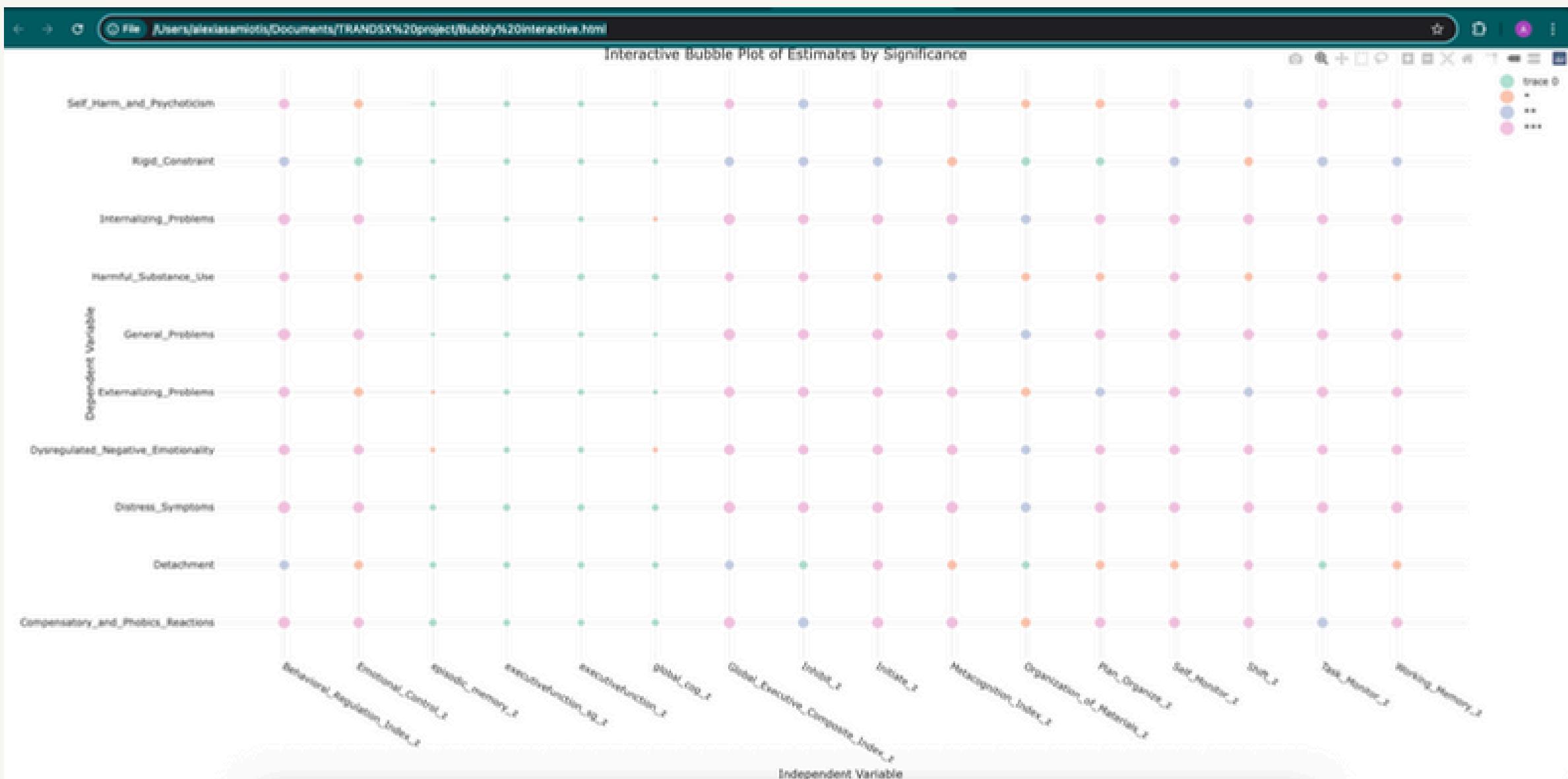
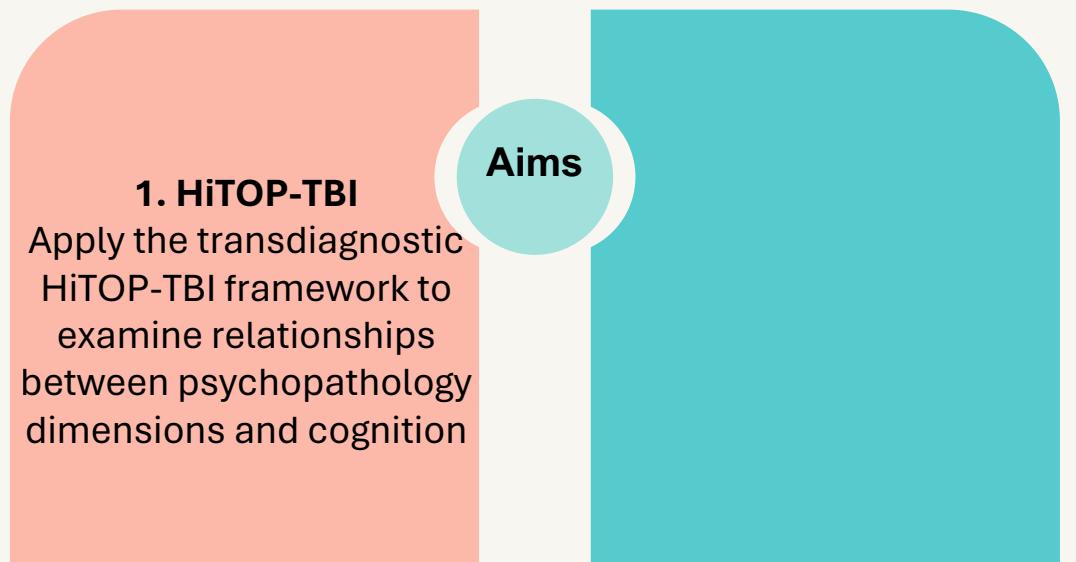
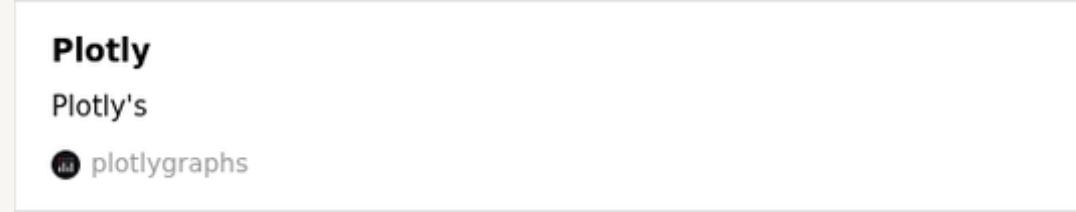


- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



# Charting: Advanced Graphics

## Interactive Plots: plotly



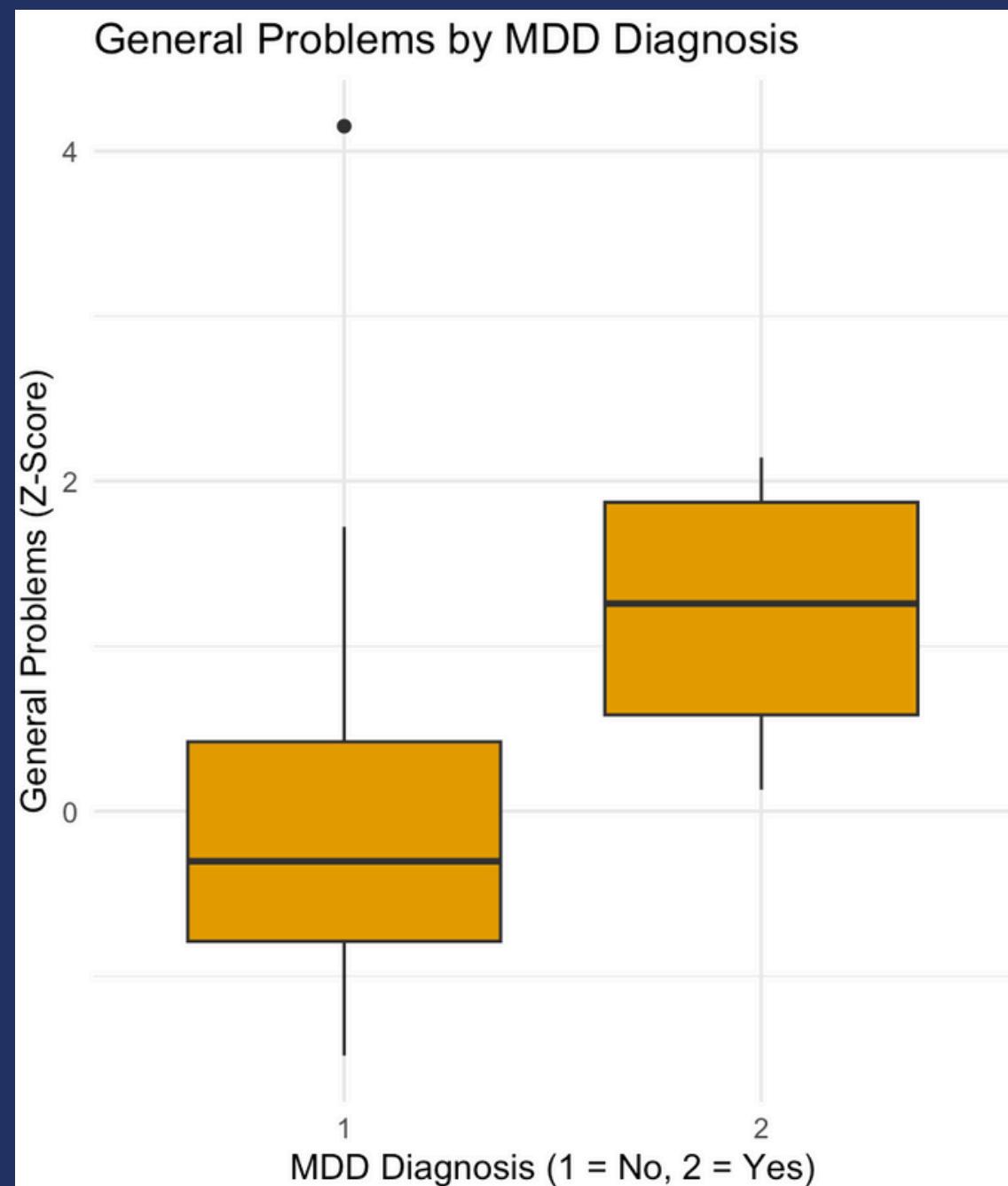
- Visualisation
- Introducing ggplot2
- Graphics in R
- Advanced graphics



## Bonus: visualising relationships

We can do all the same techniques discussed while considering other important variables.

Interesting to present data this way as it shows how diagnostic labels don't neatly fit around all individuals with severe psychopathology after TBI.



This is consistent with some of the findings from my PhD - dimensional/continuous scales better capture psych problems than DSM diagnoses.

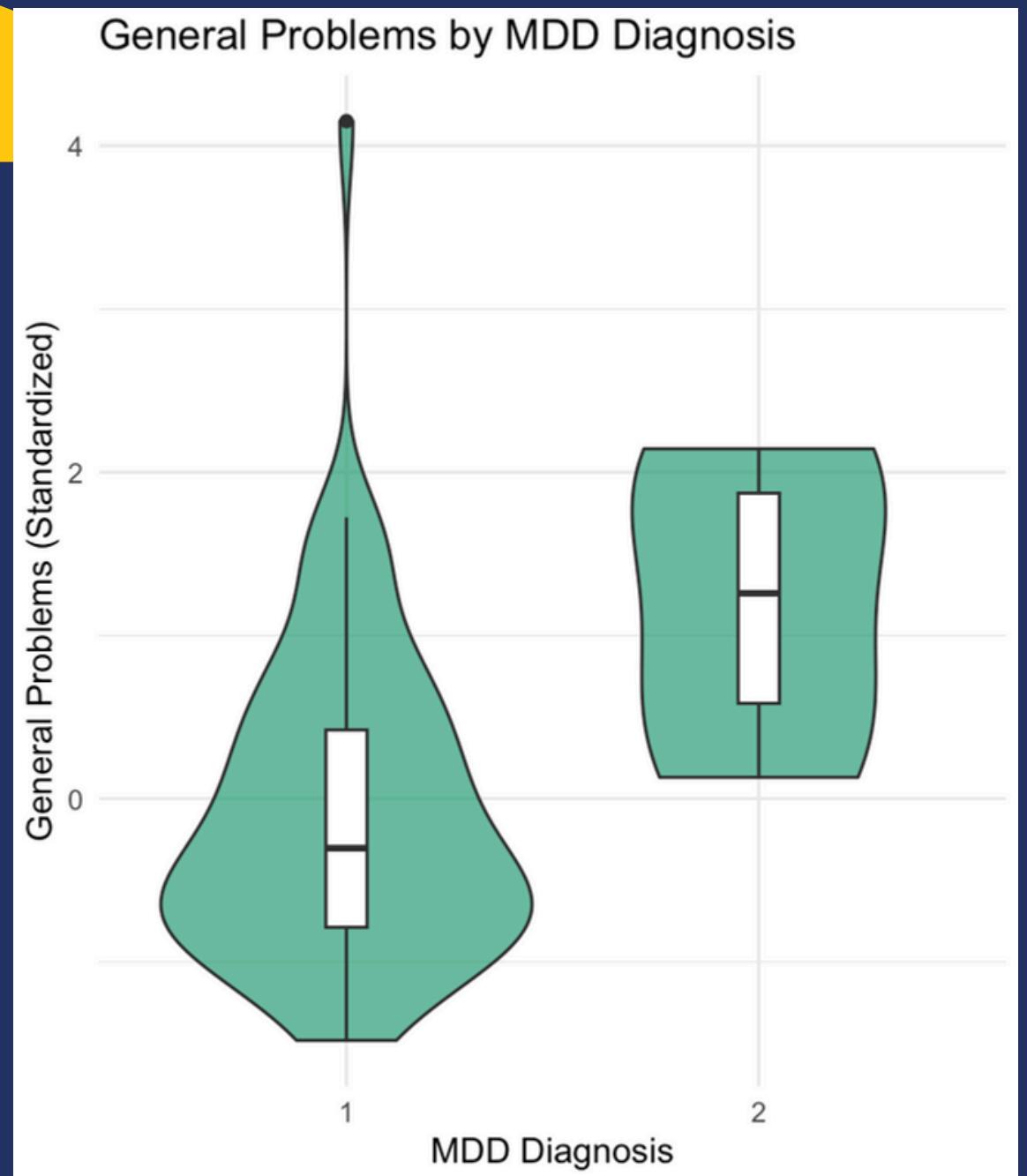
There is one outlier with an extreme (high psychopathology severity) score in the no diagnosis group

```
Synergistic Visualisation: Using Two or More Variables
Boxplots by Group – Compare Across Diagnoses or Gender
Example: By MDD diagnosis

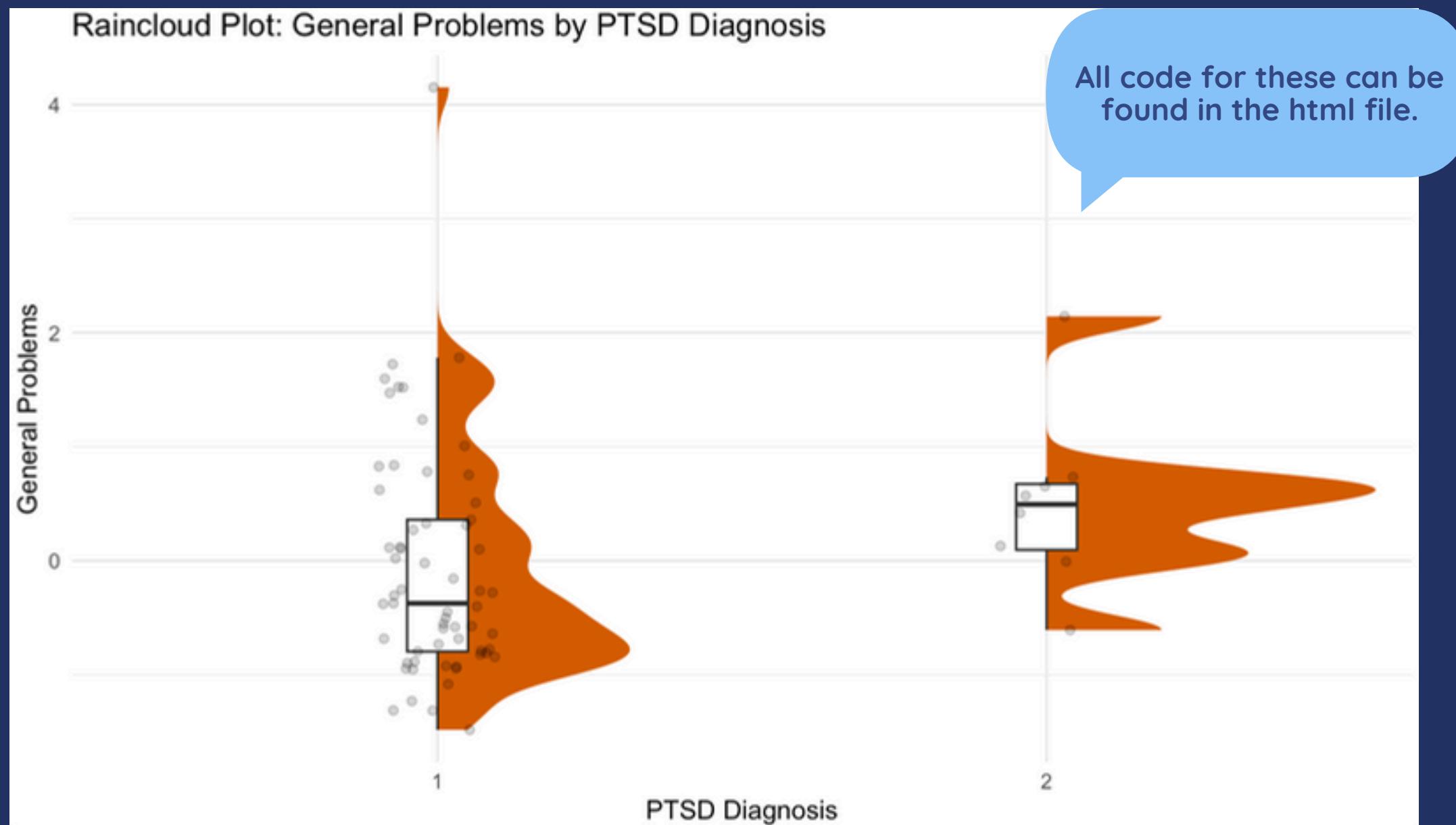
```{r}
df_clean$General_Problems <- scale(df_clean$a1_composite_score)
ggplot(df_clean, aes(x = factor(mini_mdd_current), y = General_Problems)) +
  geom_boxplot(fill = "#E69F00") +
  labs(title = "General Problems by MDD Diagnosis",
       x = "MDD Diagnosis (1 = No, 2 = Yes)",
       y = "General Problems (Z-Score)") +
  theme_minimal()
```
```

## Bonus: visualising relationships

We can do all the same techniques discussed while considering other important variables.



```
Violin Plot – Combine Density + Boxplot
```{r}
ggplot(df_clean, aes(x = factor(mini_mdd_current), y = General_Problems)) +
  geom_violin(fill = "#009E73", alpha = 0.7) +
  geom_boxplot(width = 0.1, fill = "white") +
  labs(title = "General Problems by MDD Diagnosis",
       x = "MDD Diagnosis",
       y = "General Problems (Standardized)") +
  theme_minimal()
```
```



```
Raincloud Plot (Optional, Modern & Informative)
```{r}
library(ggdist)
ggplot(df_clean, aes(x = factor(mini_ptsd_current), y = General_Problems)) +
  stat_halfeye(adjust = 0.5, width = 0.6, .width = 0, fill = "#D55E00") +
  geom_boxplot(width = 0.1, outlier.shape = NA) +
  geom_jitter(width = 0.1, alpha = 0.2) +
  labs(title = "Raincloud Plot: General Problems by PTSD Diagnosis",
       x = "PTSD Diagnosis",
       y = "General Problems") +
  theme_minimal()
```
```

# Thank you!

Associations between transdiagnostic psychopathology dimensions and cognitive functioning

after traumatic brain injury: An application of the HiTOP-TBI model

Alexia Samiotis<sup>\*1,2</sup>, Jai Carmichael<sup>1,2</sup>, Jao-Yue Carmintati<sup>1,2</sup>, Amelia J Hicks<sup>1</sup>, Jennie

Ponsford<sup>1,2</sup>, Kate Rachel Gould<sup>1,2</sup>, Gershon Spitz<sup>1,2,3</sup>

## Supervisors

- Prof Jennie Ponsford
- Dr Gershon Spitz
- Dr Jai Carmichael

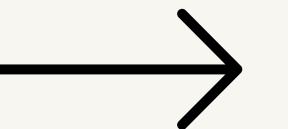
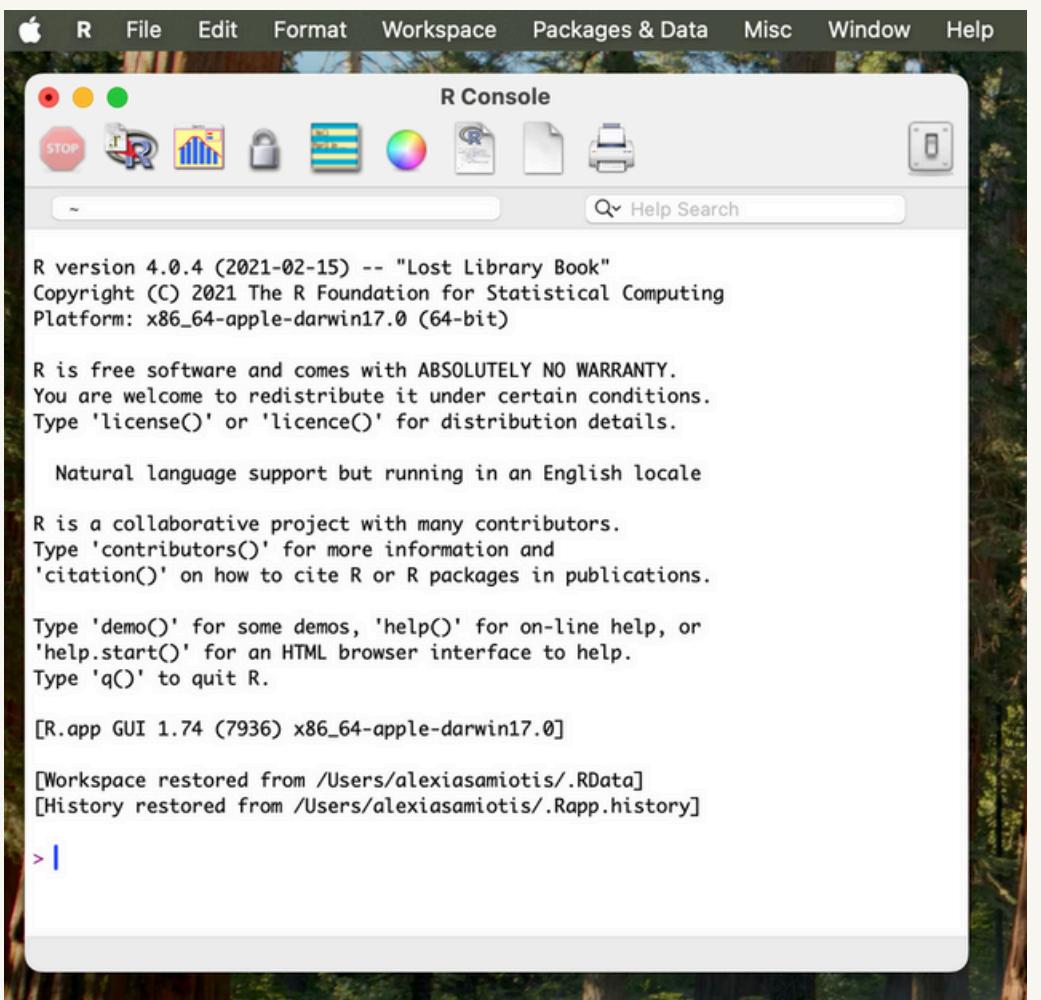
Questions or need any code?  
[alexia.samiotis1@monash.edu](mailto:alexia.samiotis1@monash.edu)  
or find me at the desks at MERRC  
Thursdays/Fridays 😊

Link to code



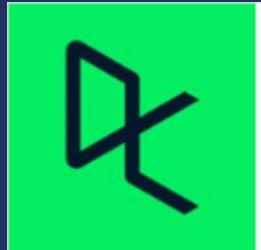
# R Crash Course

R - computer programming language



A screenshot of a DataCamp article page. The title is "How to Install R on Windows, Mac OS X, and Ubuntu Tutorial". The page content includes a green header with the DataCamp logo, a brief description, and a timestamp of "Mar 11, 2020". The main content area shows a person sitting at a laptop with code snippets and styling elements.

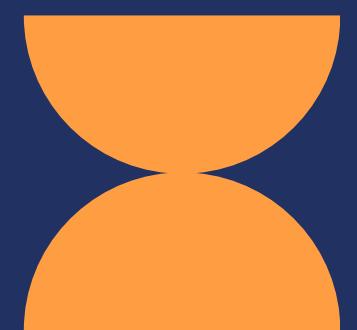
Downloading R and R Studio to Mac if needed



How to Install R on Windows, Mac OS X, and Ubuntu Tutorial

This is a beginner guide that is designed to save yourself a headache and valuable time if you deci...

DataCamp / Mar 11, 2020



# R Crash Course

R Studio - An application that provides an interface that looks the same on Mac/Windows and is more user friendly

The screenshot shows the R Studio interface with several panes:

- R script**: The leftmost pane displays an R Markdown document with code and text.
- R environment**: The top-right pane shows the global environment with objects like .RData and .Rhistory.
- Files in directory**: The bottom-right pane shows a file browser with numerous files related to "affectiveAbility" and "AlcoholProblems".
- R console**: The bottom-left pane shows the R command-line interface with standard R startup messages.



Downloading to R Studio to Mac if needed



How to Install R on Windows, Mac OS X, and Ubuntu Tutorial

This is a beginner guide that is designed to save yourself a headache and valuable time if you deci...

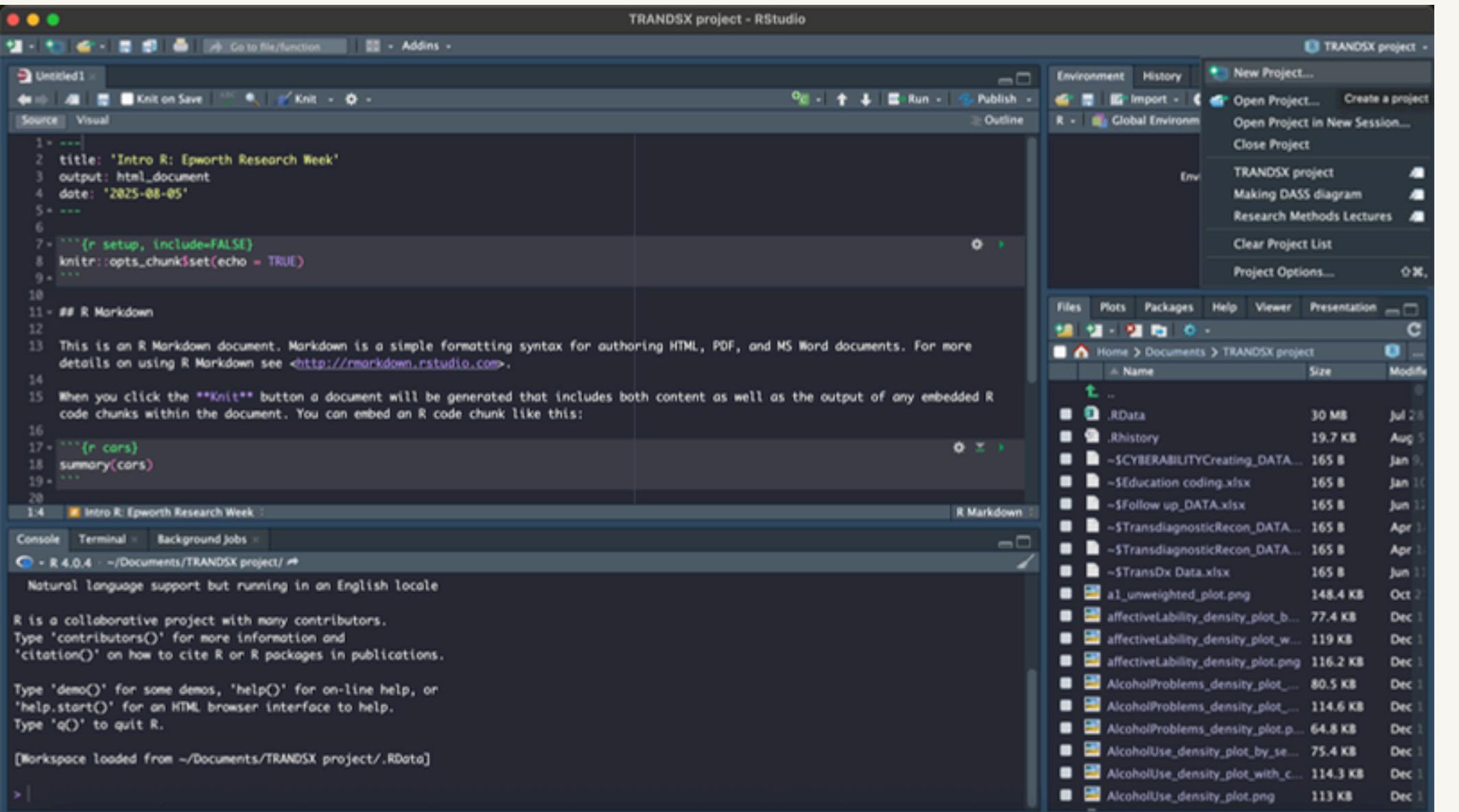
DataCamp / Mar 11, 2020

Link

# R Crash Course

## Setting Up

### 1. New Project



```
1: ---
2: title: 'Intro R: Epworth Research Week'
3: output: html_document
4: date: '2025-08-05'
5: ---
6:
7: ```{r setup, include=FALSE}
8: knitr::opts_chunk$set(echo = TRUE)
9: ```
10:
11: ## R Markdown
12:
13: This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
14:
15: When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
16:
17: ```{r cors}
18: summary(cors)
19: ```

1:4 Intro R: Epworth Research Week : R Markdown

Console Terminal Background Jobs
R 4.0.4 --> /Documents/TRANSX project/ #>
Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/Documents/TRANSX project/.RData]
```



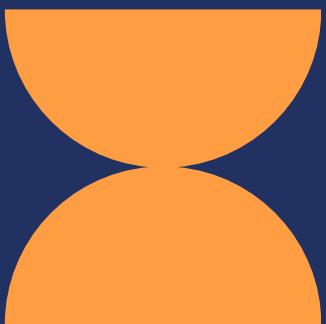
Downloading to R Studio to Mac if needed



How to Install R on Windows, Mac OS X, and Ubuntu Tutorial

This is a beginner guide that is designed to save yourself a headache and valuable time if you deci...

By DataCamp / Mar 11, 2020



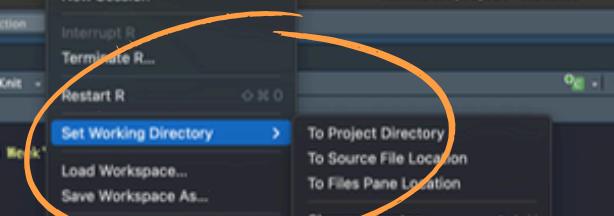
# R Crash Course



## Setting Up

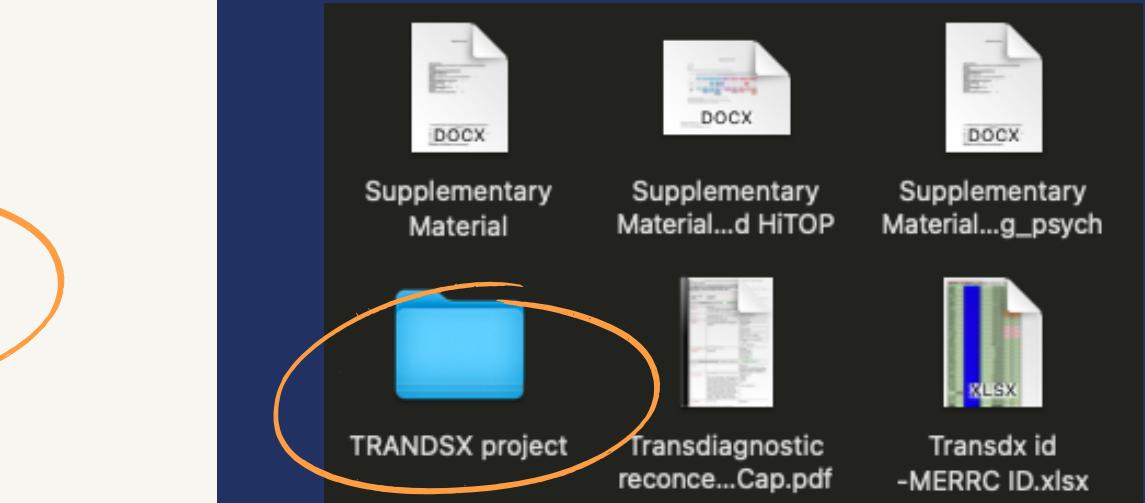
### 1. New Project

### 2. Set Working Directory



```
Macintosh HD > Users > alexiasamiotis > Documents > TRANDSX project
```

The screenshot shows the RStudio interface. In the top-left corner, the 'Session' menu is open, and the 'Set Working Directory' option is highlighted with an orange circle. The main workspace shows an R Markdown file named 'Untitled1.Rmd'. The code in the file includes R code chunks and a note about R Markdown. The bottom-left corner shows the R console output, which includes the path to the working directory: 'R 4.0.4 - ~/Documents/TRANDSX project/'. The bottom-right corner shows the file browser with a list of files in the 'TRANDSX project' folder.



Create a new folder on your computer itself (NOT OneDrive or iCloud)

Set this as your project directory. Mine is in my Documents folder.

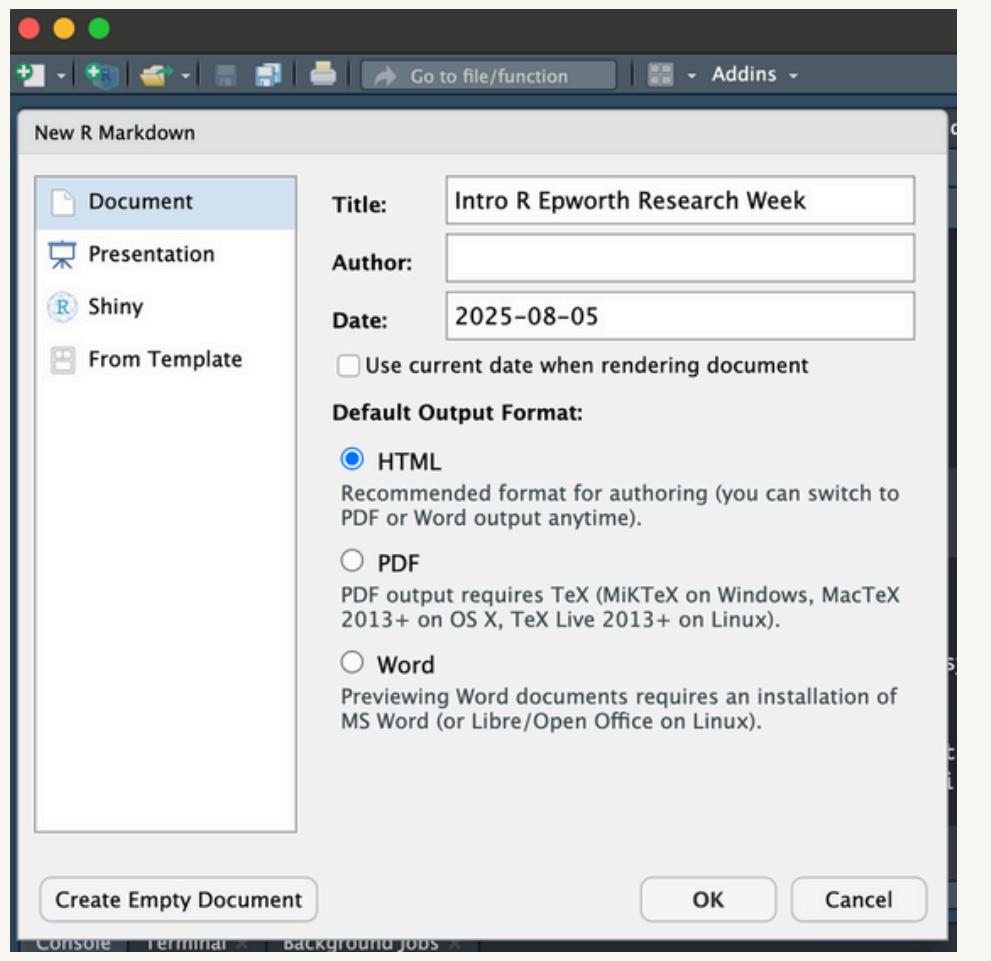
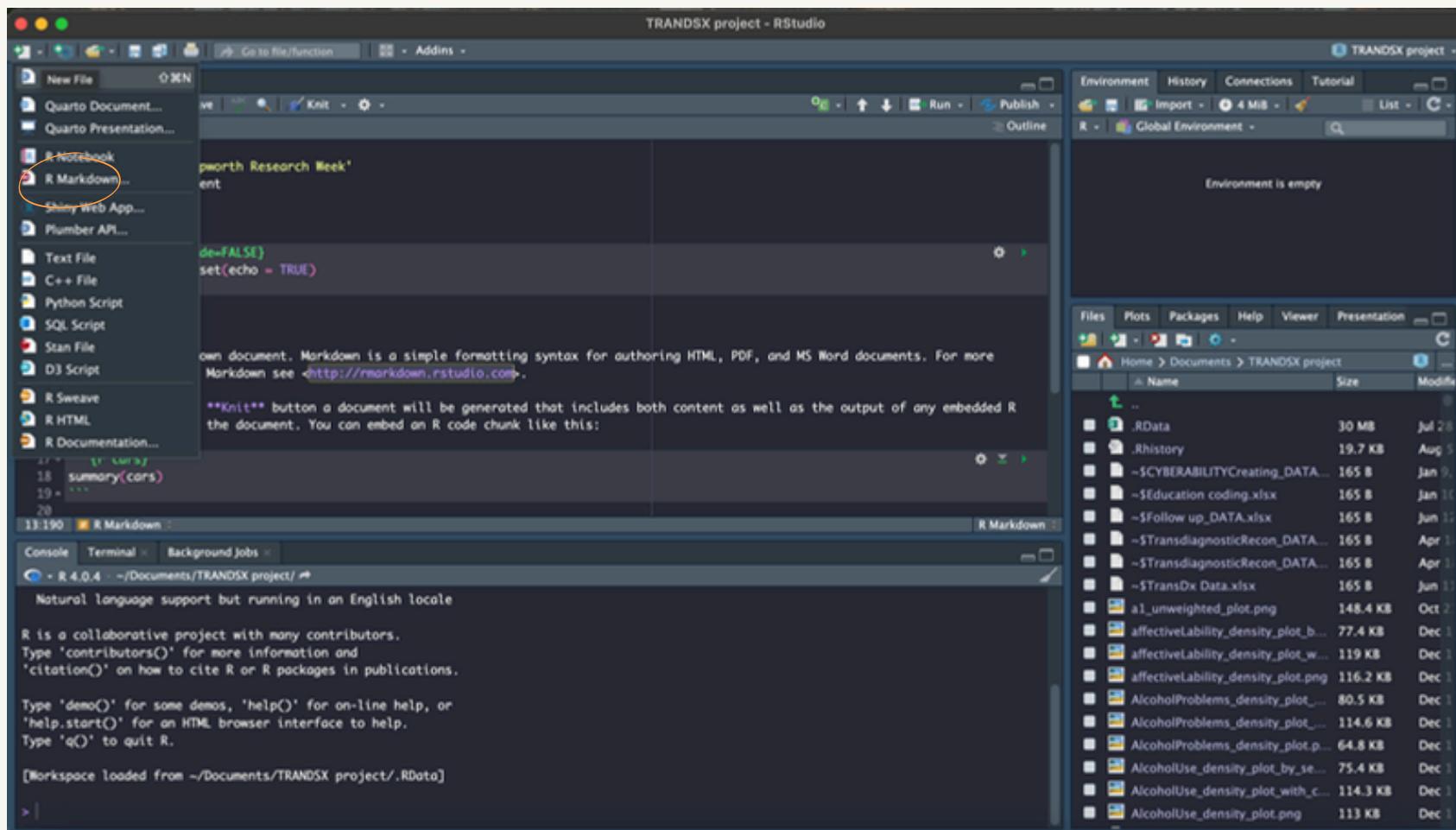
# R Crash Course

## Setting Up

### 1. New Project

### 2. Set Working Directory

### 3. Open new Markdown file



Now we create a new file within R. I prefer Markdown.

### R Markdown

Turn your analyses into high quality documents, reports, presentations and dashboards with R Markdown. Use a productive notebook interface to weave together narrative text and code to produce elegantly formatted...

[rstudio.com](https://rstudio.com)

You can knit it at the end to a nice looking file, like the one I made for today.

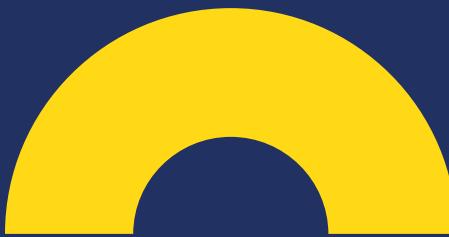


## Extra material: Importing data

Associations between transdiagnostic psychopathology dimensions and cognitive functioning

after traumatic brain injury: An application of the HiTOP-TBI model

Alexia Samiotis<sup>\*1,2</sup>, Jai Carmichael<sup>1,2</sup>, Jao-Yue Carmintati<sup>1,2</sup>, Amelia J Hicks<sup>1</sup>, Jennie Ponsford<sup>1,2</sup>, Kate Rachel Gould<sup>1,2</sup>, Gershon Spitz<sup>1,2,3</sup>



## Discrete Data

Represent the frequency of different results in the data.

So how do we get this to R Studio?

You should first have a worksheet where you entered the data

result or  
data value



DSM  
Diagnosis

MDD

frequency



Number of Individuals  
with TBI

5

AUD

12

•  
•  
•  
•  
•

•  
•  
•  
•  
•

# 1. Installing packages

```
17 - ## Installing Packages
18 These include functions that you will need to execute basic commands like importing your data or creating basic plots, exporting data. You only have to install these ONCE, not every time you open R (thankfully!). In fact, if you try and install them more than once you will effectively just be updating the package (which is fine just not necessary).
19
20 - ```{r packages}
21 install.packages("readxl") # needed to import data from excel sheets
22 install.packages("tidyverse") # meta-package with other packages I often use
23 install.packages("tidyverse") # for data tidying
24 install.packages("lubridate") # for data wrangling
25 install.packages("writexl") # for exporting to excel
26 install.packages("ggplot2") # visualisation, plots
27 install.packages("psych") # psych stats stuff :(
28 install.packages("openxlsx") # used sometimes when readxl doesn't work
29 -
30 ````
```

I like my R Studio interface with these colours, you can change yours to have a white background if you like

Tools > Options > Environment > General and choose a colour theme from the "Colour theme" dropdown menu

# 2. Opening packages

```
31 - ## Load Relevant Libraries
32 You can think of installing packages like downloading an app, once it's done it will be there. This doesn't automatically mean the app will open, you still have to click on it. In R, you have to tell the package to open by loading it in the environment. You do have to load packages EVERY time you use R Studio. If you stay in the same session and don't exit R, you don't have to load them.
33
34 - ```{r libraries, echo=FALSE}
35 library(readxl)
36 library(tidyverse)
37 library(tidyr)
38 library(lubridate)
39 library(writexl)
40 library(ggplot2)
41 library(psych)
42 library(openxlsx)
43 ````
```



Link to code

# 3.Importing and naming your data

```
45 ## Import the dataset
46 This is unpublished data that I have collected as part of my PhD looking at mental health after Traumatic Brain Injury. As you can
see there >1000 variables, many of which will be included in the analyses for my studies. Today we will only be focusing on a few of
these. I will show you how to call on the variables you would like to use.
47
48 ````{r transdiagnostic data}
49 df <- read_excel("HiTOP_followup_data.xlsx") # I called this df but can call it something more meaningful if you like.
50 view(df) # lets have a look at what our imported dataset looks like
51 ````
```



Naming: what would you like to name the dataframe? name <-

I like my R Studio interface with these colours, you can change yours to have a white background if you like

Tools > Options > Environment >  
General and choose a colour theme from the "Colour theme" dropdown menu

Link to code

[https://github.com/AlexiaSam/R-Intro/blob/main\\_Epworth Research Week Intro.md](https://github.com/AlexiaSam/R-Intro/blob/main_Epworth%20Research%20Week%20Intro.md)



# 3.Importing and naming your data

This is the function to import the data (found in the `readxl` package we loaded earlier)

Type the name of your excel file (must be verbatim)

```
45 ## Import the dataset
46 This is unpublished data that I have collected as part of my PhD looking at mental health after Traumatic Brain Injury. As you can see there >1000 variables, many of which will be included in the analyses for my studies. Today we will only be focusing on a few of these. I will show you how to call on the variables you would like to use.
47
48 #'r transdiagnostic data}
49 df <- read_excel("HiTOP_followup_data.xlsx") # I called this df but can call it something more meaningful if you like.
50 view(df) # lets have a look at what our imported dataset looks like
51 #'
```

Naming: what would you like to name the dataframe? `name <-`

I like my R Studio interface with these colours, you can change yours to have a white background if you like

Tools > Options > Environment > General and choose a colour theme from the "Colour theme" dropdown menu

[Link to code](#)



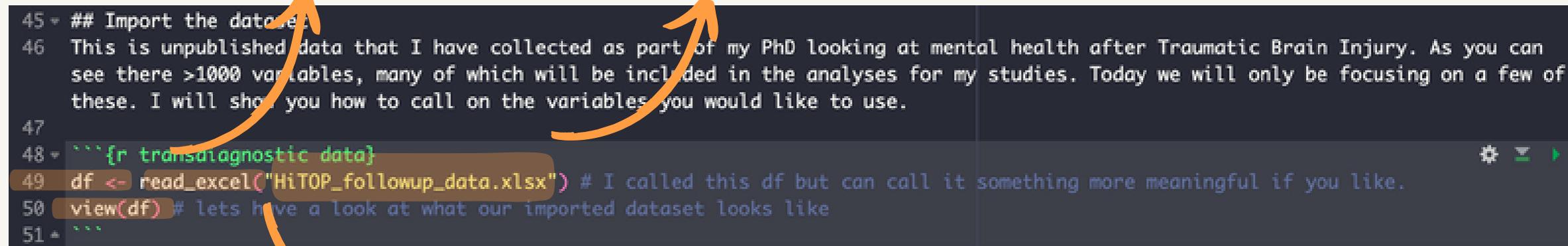
[https://github.com/AlexiaSam/R-Intro/blob/main\\_Epworth Research Week Intro.md](https://github.com/AlexiaSam/R-Intro/blob/main_Epworth%20Research%20Week%20Intro.md)

# 3.Importing and naming your data

This is the function to import the data (found in the `readxl` package we loaded earlier)

Type the name of your excel file (must be verbatim)

```
45 ## Import the dataset
46 This is unpublished data that I have collected as part of my PhD looking at mental health after Traumatic Brain Injury. As you can see there >1000 variables, many of which will be included in the analyses for my studies. Today we will only be focusing on a few of these. I will show you how to call on the variables you would like to use.
47
48 ```{r transdiagnostic data}
49 df <- read_excel("HiTOP_followup_data.xlsx") # I called this df but can call it something more meaningful if you like.
50 view(df) # lets have a look at what our imported dataset looks like
51 ```

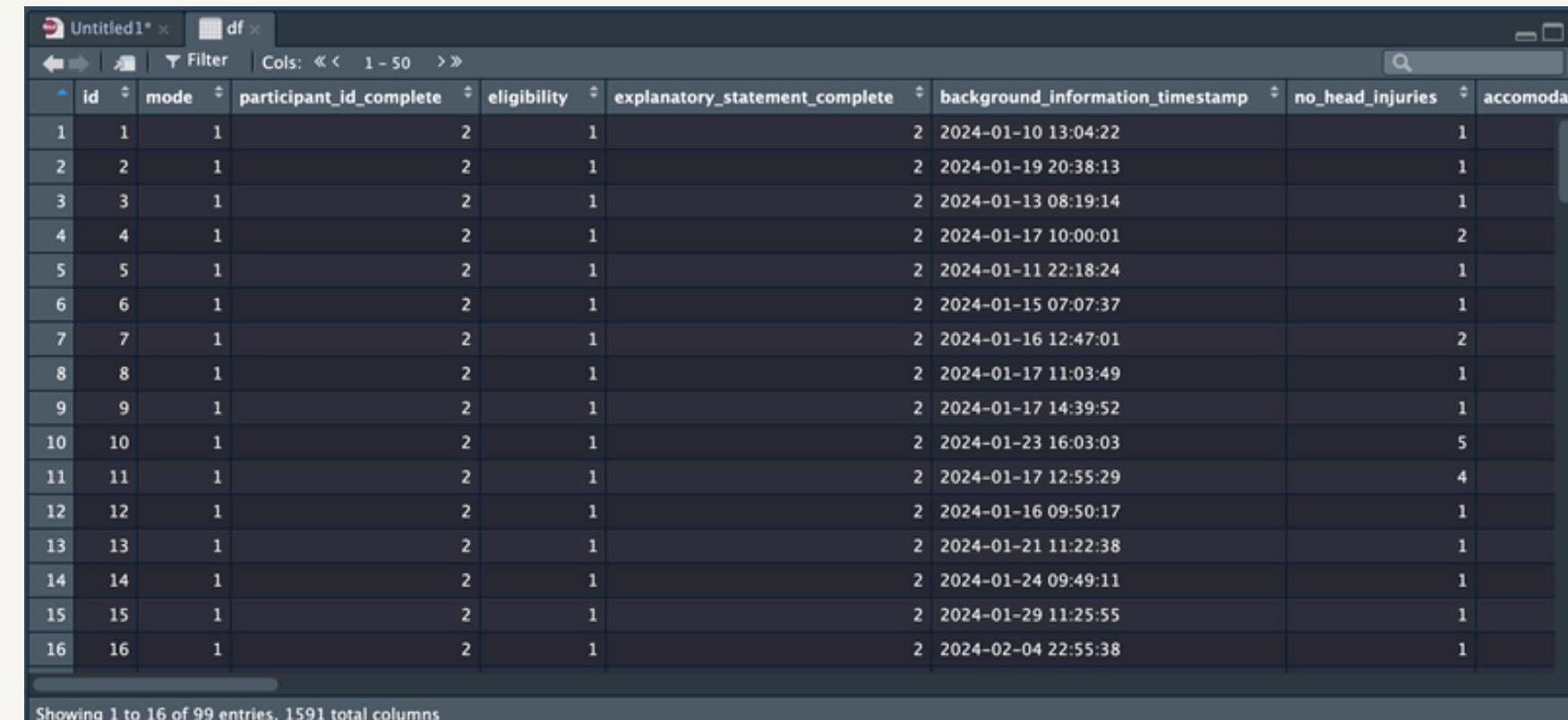

```

Let's have a look at our dataframe

Naming: what would you like to name the dataframe? `name <-`

I like my R Studio interface with these colours, you can change yours to have a white background if you like

Tools > Options > Environment > General and choose a colour theme from the "Colour theme" dropdown menu



| 1  | 1  | 1 |  | 2 | 1 |  | 2 | 2024-01-10 13:04:22 |  |  | 1 |  |  |  |  |
|----|----|---|--|---|---|--|---|---------------------|--|--|---|--|--|--|--|
| 2  | 2  | 1 |  | 2 | 1 |  | 2 | 2024-01-19 20:38:13 |  |  | 1 |  |  |  |  |
| 3  | 3  | 1 |  | 2 | 1 |  | 2 | 2024-01-13 08:19:14 |  |  | 1 |  |  |  |  |
| 4  | 4  | 1 |  | 2 | 1 |  | 2 | 2024-01-17 10:00:01 |  |  | 2 |  |  |  |  |
| 5  | 5  | 1 |  | 2 | 1 |  | 2 | 2024-01-11 22:18:24 |  |  | 1 |  |  |  |  |
| 6  | 6  | 1 |  | 2 | 1 |  | 2 | 2024-01-15 07:07:37 |  |  | 1 |  |  |  |  |
| 7  | 7  | 1 |  | 2 | 1 |  | 2 | 2024-01-16 12:47:01 |  |  | 2 |  |  |  |  |
| 8  | 8  | 1 |  | 2 | 1 |  | 2 | 2024-01-17 11:03:49 |  |  | 1 |  |  |  |  |
| 9  | 9  | 1 |  | 2 | 1 |  | 2 | 2024-01-17 14:39:52 |  |  | 1 |  |  |  |  |
| 10 | 10 | 1 |  | 2 | 1 |  | 2 | 2024-01-23 16:03:03 |  |  | 5 |  |  |  |  |
| 11 | 11 | 1 |  | 2 | 1 |  | 2 | 2024-01-17 12:55:29 |  |  | 4 |  |  |  |  |
| 12 | 12 | 1 |  | 2 | 1 |  | 2 | 2024-01-16 09:50:17 |  |  | 1 |  |  |  |  |
| 13 | 13 | 1 |  | 2 | 1 |  | 2 | 2024-01-21 11:22:38 |  |  | 1 |  |  |  |  |
| 14 | 14 | 1 |  | 2 | 1 |  | 2 | 2024-01-24 09:49:11 |  |  | 1 |  |  |  |  |
| 15 | 15 | 1 |  | 2 | 1 |  | 2 | 2024-01-29 11:25:55 |  |  | 1 |  |  |  |  |
| 16 | 16 | 1 |  | 2 | 1 |  | 2 | 2024-02-04 22:55:38 |  |  | 1 |  |  |  |  |

Showing 1 to 16 of 99 entries, 1591 total columns

## 4. Calling on specific variables

**\$** is used to call the variable

```
53 - ## Calling on specific variables
54 I'm sure you don't want to go searching through 1000 variables, so instead you can call on the variable you need and then work with
55 it. You can also perform functions on a vector of variables you call on (1). Alternatively, you can filter a subset of variables that
56 you would like to work with (2). I'll show you both.
57
58 ``{r calling on variables}
59 df$mini_mdd_current # what does our Major Depressive Disorder variable contain? Let's have a quick look by printing it here.
60 df$mini_alcohol_current # AUD diagnoses
61 df$mini_ptsd_current # PTSD diagnoses
62 ````
```

Which variable do you want to call?  
You must type verbatim the name  
that is in your dataframe.

Which dataframe contains the variable you would like to call on? Mine is `df`

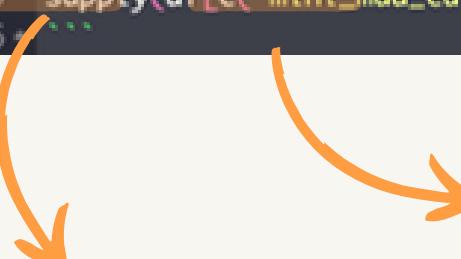
# What does the output look like?

# How should we present our diagnostic data?

5 people have MDD  
13 people have AUD  
8 people have PTSD

```
73 + ###_1. Show the data
74
75 Check what they contain by calling on each variable individually (option 1)
76 This gives you the frequency of each diagnosis. For this dataset, 1 = NO diagnosis, 2 = YES diagnosis
77 + `r`
78 table(df$mini_mdd_current)
79 table(df$mini_alcohol_current)
80 table(df$mini_ptsd_current)
81 +
82
83 OR Calling on variables altogether as a vector (option 2)
84 + `r`
85 sapply(df[c("mini_mdd_current", "mini_alcohol_current", "mini_ptsd_current")], table)
86 +
```

```
Console Terminal × Background Jobs ×
R 4.0.4 ~/Documents/TRANDSX project/ ↗
> table(df$mini_mdd_current)
1 2
66 5
> table(df$mini_alcohol_current)
1 2
65 13
> table(df$mini_ptsd_current)
1 2
70 8
> sapply(df[c("mini_mdd_current", "mini_alcohol_current", "mini_ptsd_current")], table)
mini_mdd_current mini_alcohol_current mini_ptsd_current
1 66 65 70
2 5 13 8
> |
```



Variables called on in **vector** =  
c("variable", "variable")

Same function from before, now just calling it all at once