

Single Day Event

Automation and integration of data
visualization using
R ESQUISSE & R SHINY

Vijayakumar Radhakrishnan,
Precision for Medicine



Vijayakumar Radhakrishnan

Senior Manager, Statistical Programming, Precision for Medicine

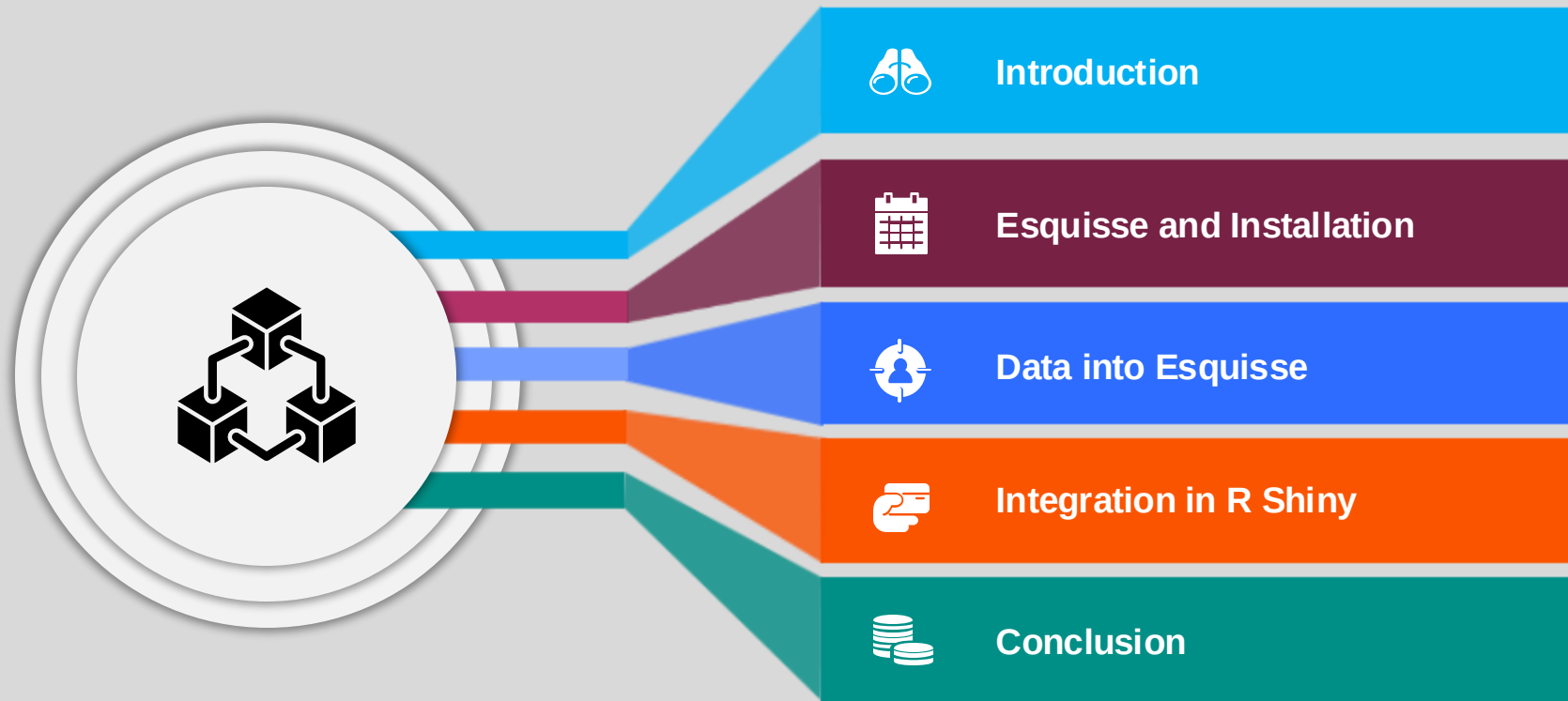
Vijayakumar has been Senior Manager of Statistical Programming at Precision for Medicine since 2022, and he brings a plethora of knowledge to the post. Prior to joining Precision for Medicine, he spent over 11 years with ICON Clinical Research in several statistical programming jobs. He has actively participated in guiding big and complicated submissions for a wide range of clients, including significant biopharmaceuticals, biotech companies, and CROs.

In addition to honing his skills in statistical programming, data analytics, and data visualisation, he is highly committed to nurturing talent and cutting-edge software solutions. He is also actively involved in various industrial forums, including Phuse, ConSPIC and PharmaSUG.

Vijayakumar has an MBA in pharmaceutical management and a master's degree in bioinformatics.



Agenda





Introduction

**R Esquisse: R program that
provides a user-friendly interface**

Data Visualizations with ggplot2

Quickly Analyze and obtain insights

**New to R or prefer a visual approach
to making charts**

Drag - Drop - Plot canvas

**Customize plot elements;
Multi-language compatibility**

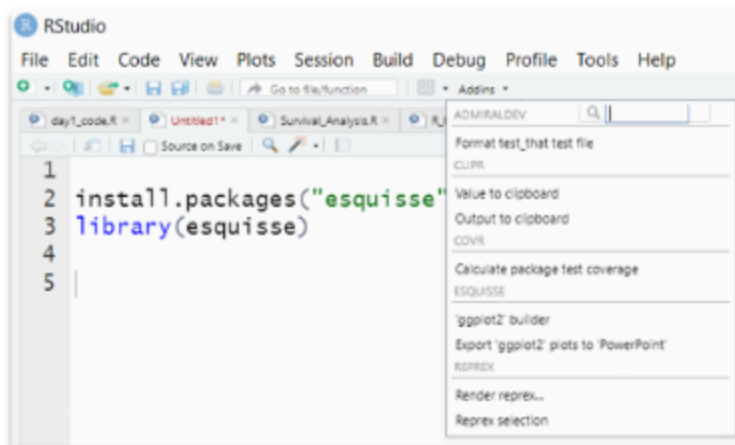




Installation

Methodology

```
install.packages("esquisse")
```



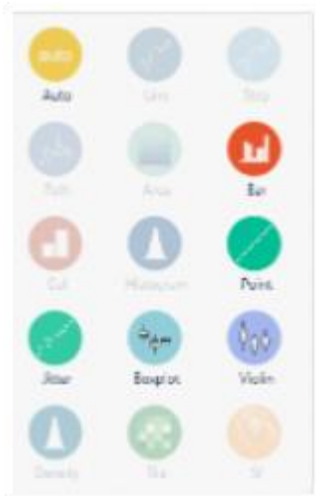
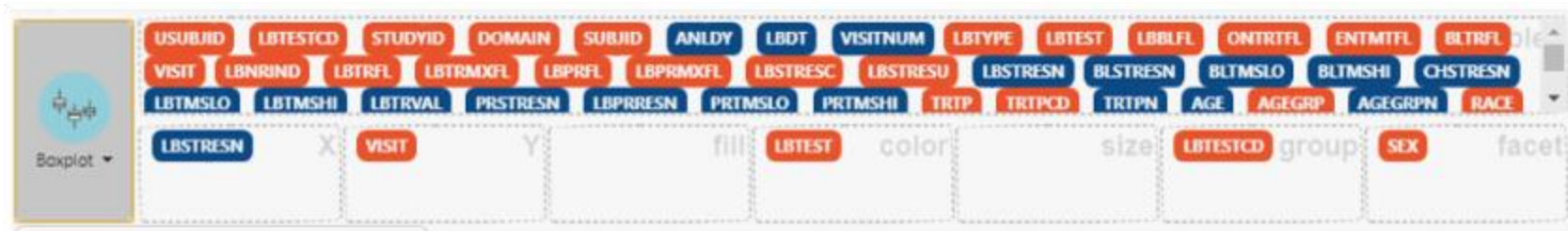


IMPORT Data into R Esquisse

There are various options for loading data into R Esquisse



IMPORT Data into R Esquisse

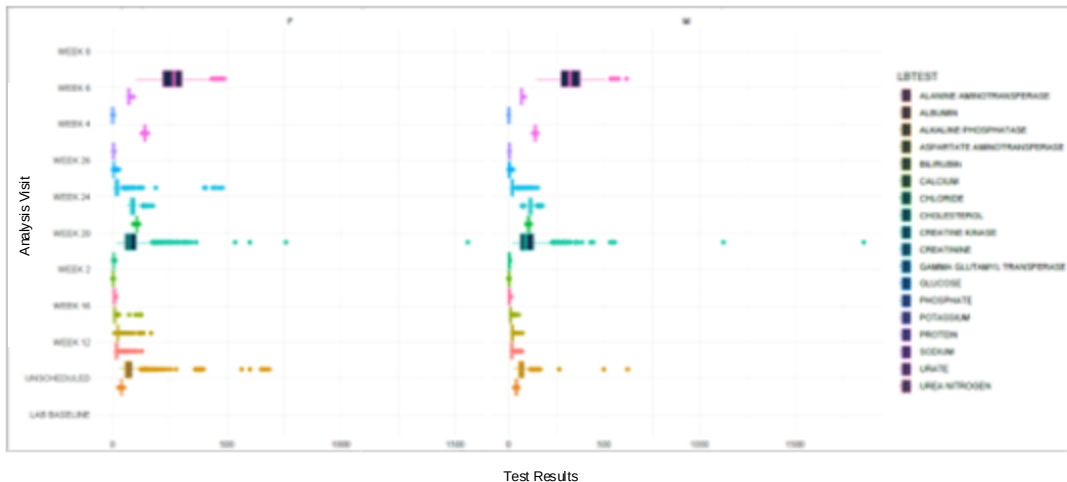


Drag to all of the containers labelled X, Y, FILL, COLOUR, SIZE, GROUP, and FACETO.
Changes which you made in each container will have impact in figures
Automatically prefers to load the figure transformation



Lab Abnormalities Report

Lab Abnormalities
Safety Population



Develop



Analysis Visit and Test results



Label for each axis



Customize the title section



Adjust the legend preference and position

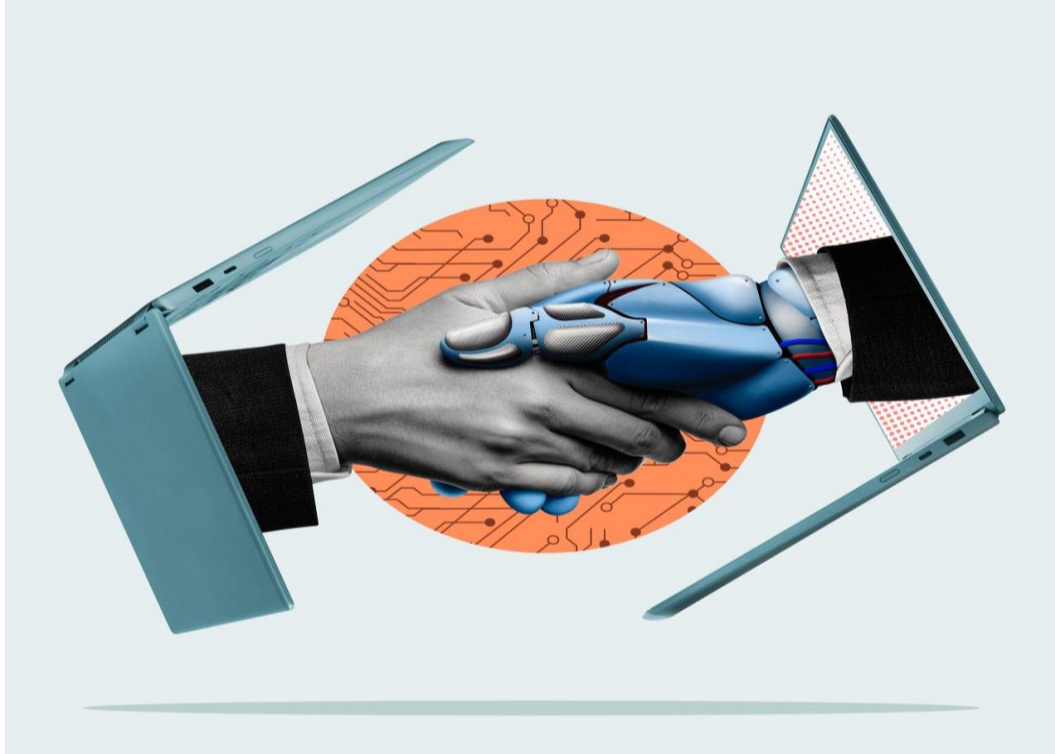


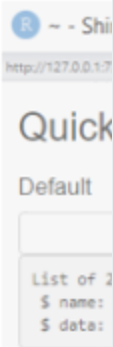
Apply colors to differentiate and highlight each of the analysis


```
ggplot(ADLBC) +  
  aes(  
    x = LBSTRESN,  
    y = VISIT,  
    colour = LBTEST,  
    group = LBTESTCD  
  ) +  
  geom_boxplot(fill = "#112446") +  
  scale_color_hue(direction = 1) +  
  labs(  
    x = "Test Results",  
    y = "Analysis Visit",  
    title = "Lab Abnormalities",  
    subtitle = "Safety Population"  
  ) +  
  theme_minimal() +  
  theme(plot.title = element_text(size = 20L, face = "bold")) +  
  facet_wrap(vars(SEX), scales = "free_x")
```



Integration in Rshiny





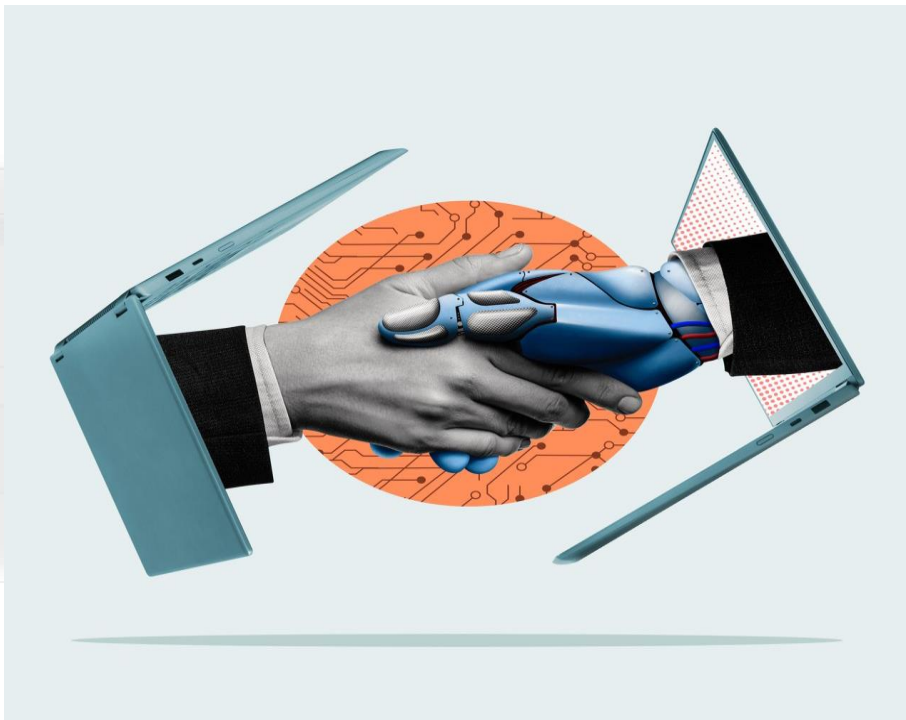
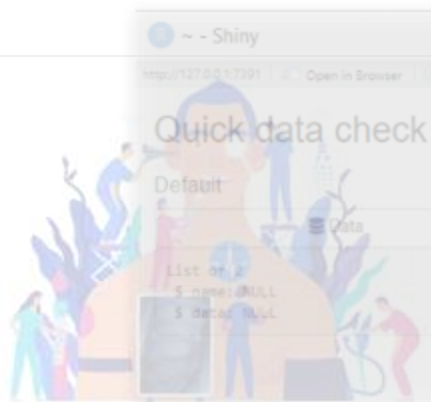
```

#Page(
  "Choose data module",
  {
    n = 4, tagsH4("Default"),
    img_chooseDataUI,
    resDataUI(id = "choose1"),
    stimTextOutput(outputId = "res1")})
function(input, output, session)
  <- callModule(
    dataServer, id = "choose1",
    onstart = FALSE)
  res1 <- renderPrint({
    activeValuesToList(res_dat1)})
  }, server)

```



Integration in Rshiny





Integration in Rshiny

```
ui <- fluidPage(  
  tags$div("Choose data"),  
  fluidRow(  
    column(  
      width = 4, tags$  
      # Using chooseDataUI  
      chooseDataUI(id,  
        verbatimTextOut)  
    )  
  )  
)  
server <- function(in)  
{  
  res_data1 <- callModule(  
    chooseDataServer,  
    launchOnStart = F,  
    output$res1 <- renderText(  
      reactiveValue  
    )  
  }  
  shinyApp(ui, server)
```



R ~ - Shiny

http://127.0.0.1:7391 | Open in Browser

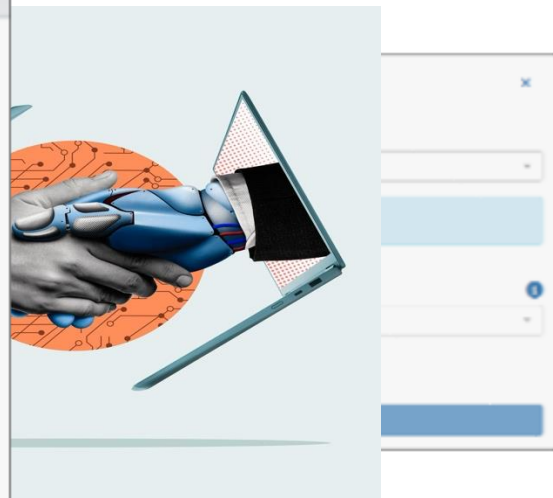
Quick data check

Default

Data

List of 2

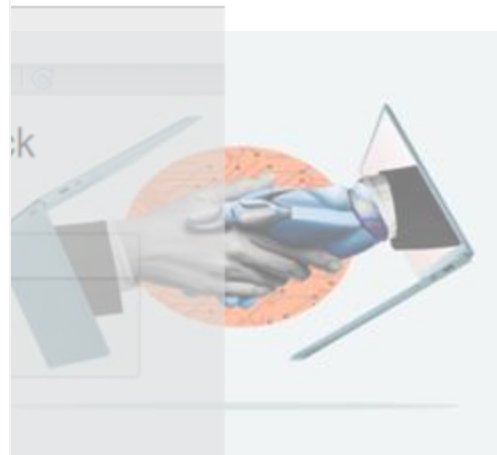
- \$ name: NULL
- \$ data: NULL





Integration in Rshiny

```
ui <- fluidPage(  
  tags$h2("choose data"),  
  fluidRow(  
    column(  
      width = 4, tags  
      # Using chooseD  
      chooseDataUI(id  
      verbatimTextOut  
    )  
  )  
)  
  
server <- function(in  
  res_dat1 <- callMod  
  chooseDataServer,  
  launchonstart = F  
  output$res1 <- rend  
  str(reactiveValue  
)  
)  
shinyApp(ui,server)]
```





Integration in Rshiny

```
ui <- fluidPage(  
  tags$h2("choose data module"),  
  fluidRow(  
    column(  
      width = 4, tags$h4("Default"),  
      # Using chooseDataUI  
      chooseDataUI(id = "choose1"),  
      verbatimTextOutput(outputId = "res1"))))  
  
server <- function(input, output, session)  
{  
  res_dat1 <- callModule(  
    chooseDataServer, id = "choose1",  
    launchOnStart = FALSE)  
  output$res1 <- renderPrint(  
    str(reactiveValuesToList(res_dat1)))  
}  
  
shinyApp(ui, server)|
```

Select a dataset

Choose a data.frame :

List of data frames

No data selected Use a data.frame :

Select variables to keep :

Nothing selected

Legend discrete continuous

Quick data check

Default

Data

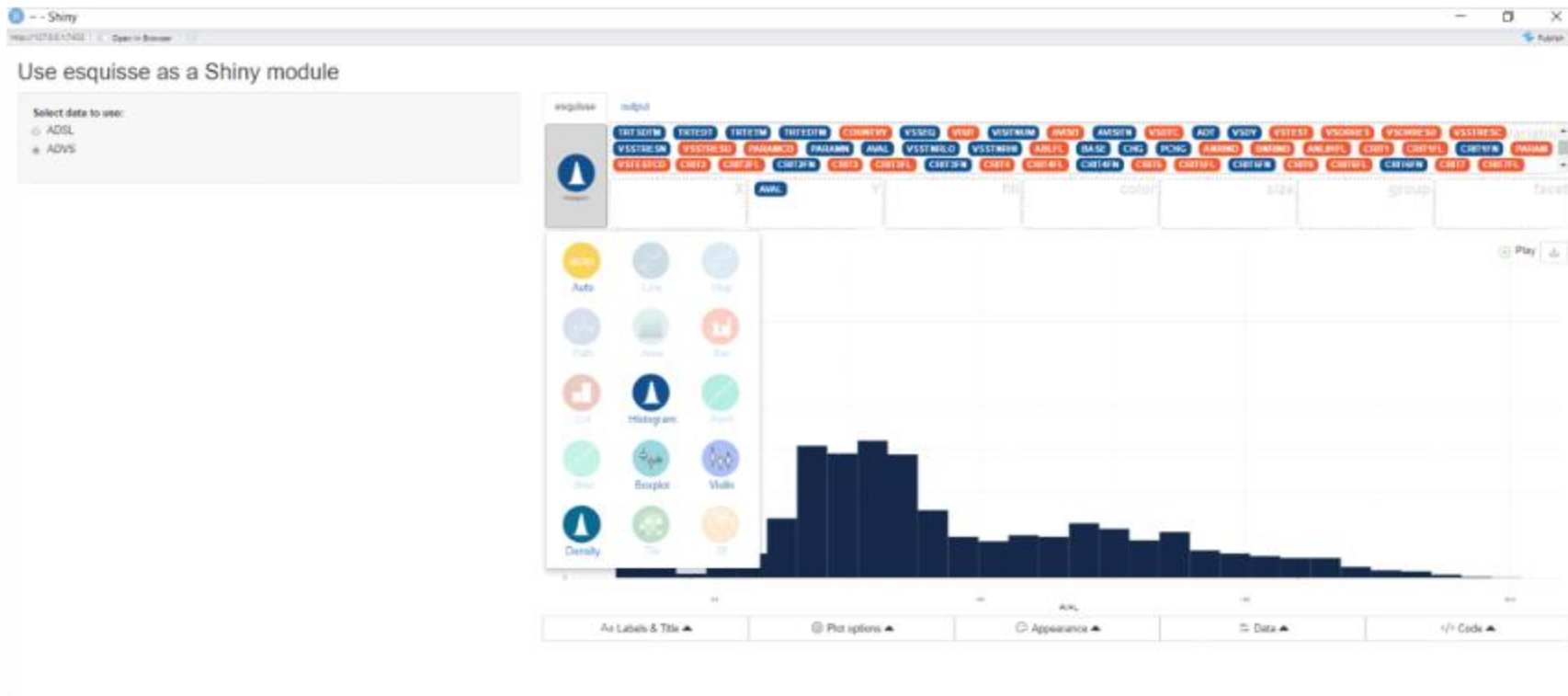
List of 2

\$ name: NULL

\$ data: NULL



Integration in Rshiny



Conclusion



Overall, R Esquisse is a R program that provides a user-friendly interface for making data visualizations with ggplot2.



This package is especially useful for stakeholders who want to conduct quick interactive analysis and acquire insights from data sources, whether they are new to R or prefer a more visual approach to plotting.



When we integrate it with R shiny, we can easily create data visualizations with a full GUI experience, and there are numerous features available to integrate it with R shiny.



Resulting in more interactive data visualizations with the source data without any prior knowledge of figures. Can be used by different entities like programmers, statisticians and clinicians.





References

1. Meyer F, Perrier V. Shiny Usage. GitHub. Accessed September 3, 2024.
<https://mpn.metworx.com/packages/esquisse/0.3.1/articles/shiny-usage.html>
2. Esquisse Package in R Programming. Geeks for Geeks. Updated December 8, 2021. Accessed September 3, 2024.
<https://www.geeksforgeeks.org/esquisse-package-in-r-programming/>
3. Radečić D. R Esquisse: How to Explore Data in R Through a Tableau-like Drag-and-Drop Interface. Appsilon. Accessed September 3, 2024. <https://www.appsilon.com/post/r-esquisse-drag-n-drop-charts>
4. Scheithauer G. Analyze Data quickly with Esquisse. Medium. Published April 15, 2022. Accessed September 3, 2024.
<https://towardsdatascience.com/how-to-analyze-data-quickly-without-tableau-but-close-66eb1f2fffd2>



Contact Us

Vijayakumar Radhakrishnan

(Vijayakumar.radhakriashnan@precisionformedicine.com)

Thank You



**The Global Healthcare
Data Science Community**

Contact Channels

📞 UK +44 1843 609600
✉ office@phuse.global
🌐 phuse.global

Social Media

🐦 [@phusetwitla](https://twitter.com/phusetwitla)
📘 [/phusebook](https://www.facebook.com/phusebook)
▶ [/phusetube](https://www.youtube.com/channel/UCphusetube)
🏢 [/company/phuse](https://www.linkedin.com/company/phuse)