



# Intro to RShiny

Martin Paleico

# What is RShiny?

- R package that allows for constructing interactive web applications
- Simple to program (with some R knowledge)
- No HTML knowledge required
- Showcasing paper results, interactive presentations, thesis, tools, etc
- Server software for easy hosting also available
- Easy to install, but it also already runs directly on RStudio instances (File - New File - Shiny Web App)
- Try our new RStudio notebooks on [jupyter.hpc.gwdg.de](http://jupyter.hpc.gwdg.de) !

## File Setup

- Either single app.R, or separate ui.R and server.R
- plus an optional global.R
- and an optional www folder where things such as images can be stored
- all stored in a single folder
- Name of the folder will be the name of the app
- In R, you can start your app with `runApp("folder-name")`, will open a new tab in your default web browser
- Apps can be started in "showcase" mode which displays the app's code to the end-user:

```
1 runApp("MyApp", display.mode = "showcase")  
2 #or have DESCRIPTION and Readme.md files
```

# Code Setup

```
1 library(shiny)
2 #can define global stuff here
3 ui <- fluidPage(...) #this is an object we are creating an assigning to UI
4 server <- function(input, output) { ... } #whereas this is an R function
5 shinyApp(ui = ui, server = server)
```

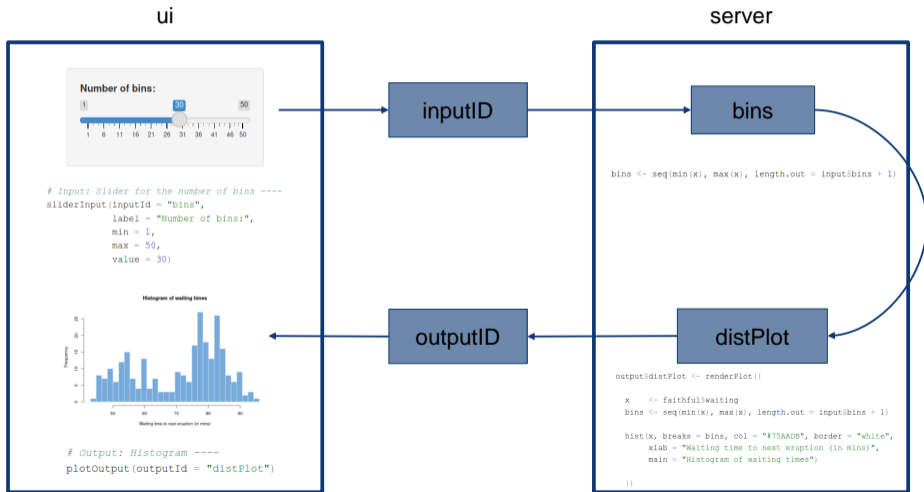
- ui is what the user can see and interact with
- server defines what gets displayed for the user to see, what happens when the user interacts with ui
- Example from `runExample("01_hello")`

Hello Shiny!



```
app.R 1 view source
library(shiny)
# Define UI for app that draws a histogram ----
ui <- fluidPage()
# App title ----
titlePanel("Hello Shiny!"),
# Sidebar layout with input and output definitions ----
sidebarLayout(
  # Sidebar panel for inputs ----
  sidebarPanel(),
  # Input slider for the number of bins ----
  sliderInput(inputId = "bins",
             labels = "Number of bins",
             min = 1,
             max = 50,
             value = 30)
),
```

# Diagram



## ui

```
1 ui ← fluidPage(  
2   titlePanel("Hello"),  
3   sidebarLayout(sidebarPanel(sliderInput(inputId = "bins", ...)),  
4                   mainPanel(plotOutput(outputId = "distPlot"))  
5   )  
6 )
```

- Create a "fluid page"
- with a given title
- and a two pane layout
- which we can fill with inputs and outputs, each with a named ID

# server

```
1 server ← function(input, output) {  
2   output$distPlot ← renderPlot({  
3     x ← faithful$waiting #retrieve a dataset  
4     #retrieve values from slider  
5     bins ← seq(min(x), max(x), length.out = input$bins + 1)  
6     hist(x, breaks = bins)  
7   })  
8 }
```

- We reference previous input and output IDs

## Some special features

Can stack HTML objects:

```
1 mainPanel(  
2     h1("First level title"),  
3     h2("Second level title"),  
4     h3("Third level title"),  
5     h4("Fourth level title"),  
6     h5("Fifth level title"),  
7     h6("Sixth level title")  
8 )
```

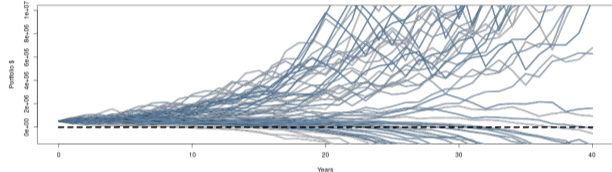
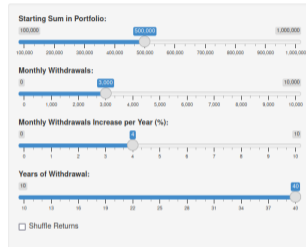
Images, need to be in a folder named www:

```
1 img(src = "my_image.png", height = 72, width = 72)
```



# Example App

Hello Shiny!



Monthly Withdrawals at End of Period 14403  
Average Portfolio Value and SD 10613 k 22379 k  
Failed Portfolios 23.42%

Data from <https://www.slickcharts.com/sp500/returns>, S&P 500 yearly returns since 1926

app.R

show with app

```
library(shiny)

display_as_int <- function(input) {
  return (format(round(input, 0), nsmall = 0))
}

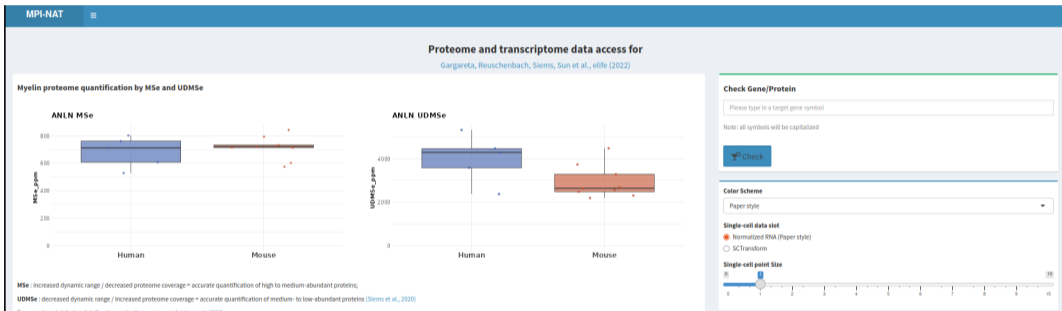
data <- read.csv("history.csv") #returns a dataframe object
data <- data[order(data$year),] #resort by ascending year

# Define UI for app that draws a histogram ----
ui <- fluidPage()

# App title ----
titlePanel("Portfolio Simulation")
```

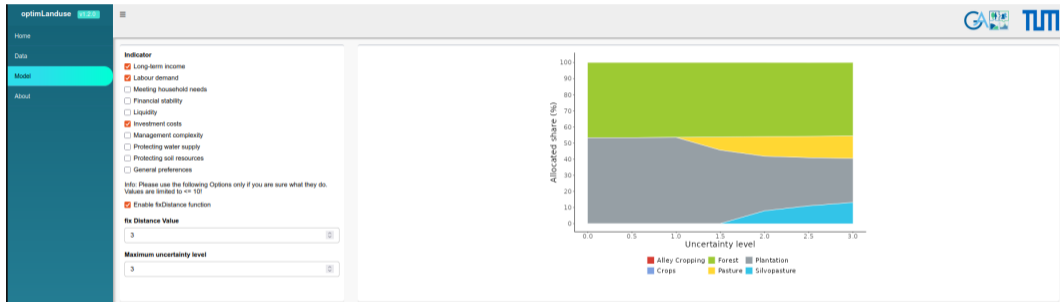
[rshiny.gwdg.de/apps/sample-apps/portfolio\\_sim/](https://rshiny.gwdg.de/apps/sample-apps/portfolio_sim/), Inspired by [firecalc.com](https://firecalc.com)

# Example App 1



CNS Myelin: Genome visualizer  
<https://rshiny.gwdg.de/apps/CNS-myelin/>

## Example App 2



OptimLandUse: optimal land allocation  
<https://rshiny.gwdg.de/apps/optimLanduse/>

# Example App 3

The screenshot displays the CLARUS web application interface. At the top, there is a blue header with the CLARUS logo and navigation links: Menu, Home, Select Data, and Interact. Below the header, the interface is divided into several sections:

- Selected patient:** A dropdown menu showing "Patient 0 (in Test Data, True label: 0, Predicted label: 0, GNNs prediction confidence: 24.11)".
- Nodes displayed:** A slider control with a value of 1, indicating the number of nodes displayed. A note below states: "(their near neighbours will also be displayed in the graph)".
- Confusion matrix:** A 2x2 matrix showing True Positives (125), True Negatives (0), False Positives (0), and False Negatives (125). The overall performance is noted as Sensitivity: 1, Specificity: 1. Below the matrix are buttons for "Download Results", "Print", and "Refresh". A note below the buttons says: "The download might take a few minutes."
- Graph visualization:** A network graph with nodes labeled N01, N02, N03, N04, N05, N06, and N07. Node N01 is highlighted in red, and its neighbors (N02, N03, N04, N05, N06) are highlighted in blue.
- Sort nodes:** A dropdown menu set to "Name (A to Z)".
- Color edges:** A dropdown menu set to "Salinity".
- Color nodes:** A dropdown menu set to "One color (default)".
- Modify Graph:** A section with buttons for "Clear", "Reset", and "Refresh", and radio buttons for "Delete Node", "Add a new Node", and "Delete Edge".

Clarus: eXplainable Graph NNs  
<https://rshiny.gwdg.de/apps/clarus/>

# Outlook

- There's a Python library now which seems to be advanced
- Alternative hosting server software in [ShinyProxy](#) (containers per app)
- Always looking for more apps to host!
- Course Q2 2025?

## Bonus: Datavzrd

- New tool for interactive reports in HTML
- For tabular data, based on YAML config files
- e.g.: Genomic data
- Server-free! Portable! No hosting!

My oscar report / movies ▾

Filter...

oscars 184 rows  
movies 184 rows  
movies-plot  
oscar-plot

Multiple datasets

Heatmaps

plot/sort/  
hide/search

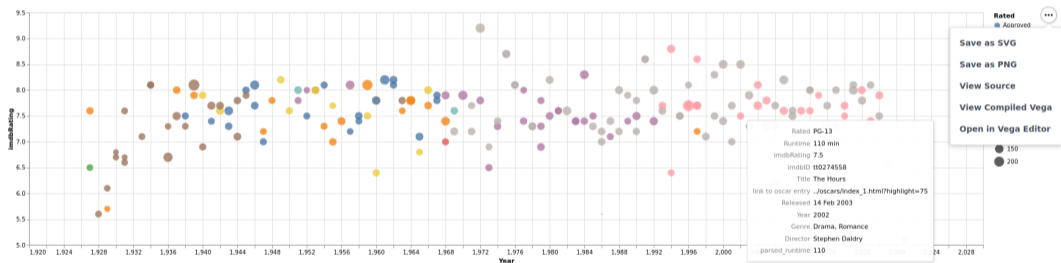
Linkouts/QR  
codes/copy

Title	Year	Rating	Released	Runtime	Genre	Director	Imdb rating	ImdbID	link to oscar entry	QR	Copy
7th Heaven ▾	1927	Unrated	6 May 1927	1h 50m	Drama	Frank Borzage	7.8	m0018379	link to oscar entry	QR	Copy
Coquette ▾	1929	Unrated	06 Apr 1929	1h 16m	Drama, Romance	Sam Taylor	7.5	m0019788	link to oscar entry	QR	Copy
The Divorcee ▾	1930	Passed	19 Apr 1930	1h 24m	Romance, Drama	Robert Z. Leonard	7.5	m0020827	link to oscar entry	QR	Copy
Min and Bill ▾	1930	Passed	29 Nov 1930	1h 9m	Comedy, Drama	George W. Hill	7.5	m0021148	link to oscar entry	QR	Copy
The Sin of Madelon Claudet ▾	1931	Passed	24 Oct 1931	1h 15m	Drama	Edgar Selwyn	7.5	m0022386	link to oscar entry	QR	Copy
Morning Glory ▾	2010	PG-13	10 Nov 2010	1h 47m	Comedy, Drama, ...	Roger Michell	7.5	m1126618	link to oscar entry	QR	Copy

Export options

Download excel sheet  
Download CSV  
Unhide columns  
Show/Hide Line Numbers  
Export table page as SVG  
Clear filters

# Bonus: Datavzrd



Plot definition with Vega "visualization grammar"

# Bonus: Datavzrd

