

```
## server.R -----
require(shiny)

x <- precip
a <- extendrange(x, f=0.05)

shinyServer(function(input, output){
  output$hist.reactive <- renderPlot({
    bks <- seq(a[1], a[2],
                length.out=input$nclass+1)
    hist(x,
          breaks=bks)
  })
})

## ui.R -----
shinyUI(fluidPage(
  sidebarPanel(
    sliderInput(inputId="nclass",
                label="Número de classes:",
                min=1, max=30, step=1, value=10)),
  mainPanel(
    plotOutput("hist.reactive"))
))
```

The diagram illustrates the data flow between the UI and Server components. A blue circle labeled '1' is positioned at the bottom left, corresponding to the 'sliderInput' in the UI code. A blue curved arrow originates from this circle and points upwards towards the 'hist.reactive' variable in the Server code. A red circle labeled '2' is positioned at the bottom right, corresponding to the 'plotOutput' in the UI code. A red curved arrow originates from this circle and points upwards towards the 'renderPlot' function in the Server code.