ui.R example

Chi-Lin Yu

```r
### Distribution App : ui.R
library(shiny)
library(ggplot2)
library(MASS)
# Define UI for random distribution app ----
ui <- fluidPage(
  titlePanel("Distribution Demo"),
  # Sidebar layout with input and output definitions
  sidebarLayout(
    # Sidebar panel for inputs
    sidebarPanel(
      # Input: Select the random distribution type
      radioButtons("dist", "Distribution type:",
                   c("Normal" = "norm",
                     "t Dist" = "t",
                     "Uniform" = "unif",
                     "Log-normal" = "lnorm",
                     "Exponential" = "exp",
                     "Bivariate-Normal" = "mvn"
      )),
      # Spacing
      br(),
      conditionalPanel(
        condition = "input.dist == 't'",
        sliderInput('df',"degree of freedom",
                    value = 1,
                    min = 1,
                    max = 30
      )),
      # Input: Slider for the number of observations
      sliderInput("n",
                  "Number of observations:",
                  value = 500,
                  min = 1,
                  max = 1000)

    ),
    # Main panel for displaying outputs
    mainPanel(
      # Output: plot, summary, and table
      tabsetPanel(type = "tabs",
                  tabPanel("Plot", plotOutput("plot")),
                  tabPanel("Summary", verbatimTextOutput("summary")),
                  tabPanel("Table", tableOutput("table"))
      )
    )
  )
)
```

```
### Distribution App: server.R
library(shiny)
library(ggplot2)
library(MASS)
server <- function(input, output) {
  # define the function for generating random numbers
  d <- reactive({
    dist <- switch(input$dist,
                   norm = rnorm,
                   t = rt,
                   unif = runif,
                   lnorm = rlnorm,
                   exp = rexp,
                   mvn = mvrnorm,
                   rnorm)
    if (input$dist == "t"){
      dist(input$n,df = input$df)
    } else if (input$dist == "mvn") {
      mvrnorm(input$n, mu = c(0, 0),
              Sigma = matrix(c(1, 0.5, 0.5, 1),2))
    } else {
      dist(input$n)
    }
  })
  # for plots
  output$plot <- renderPlot({
    dist <- input$dist
    n <- input$n
    if (input$dist == "mvn") {
      par(mar = rep(0, 4))
      p <- persp(kde2d(d()[,1], d()[,2],n = input$n),xlab = "X",
                 phi = 45, theta = 30,shade = .5, border = NA)
    } else {
      hist(d(),main = paste("r", dist, "(", n, ")", sep = ""),
           col = "grey", border = "white", xlab = "Values")
    }
  })
  # for summary table
  output$summary <- renderPrint({
    summary(d())
  })
  # for showing data
  output$table <- renderTable({
    d()
  })
}
```