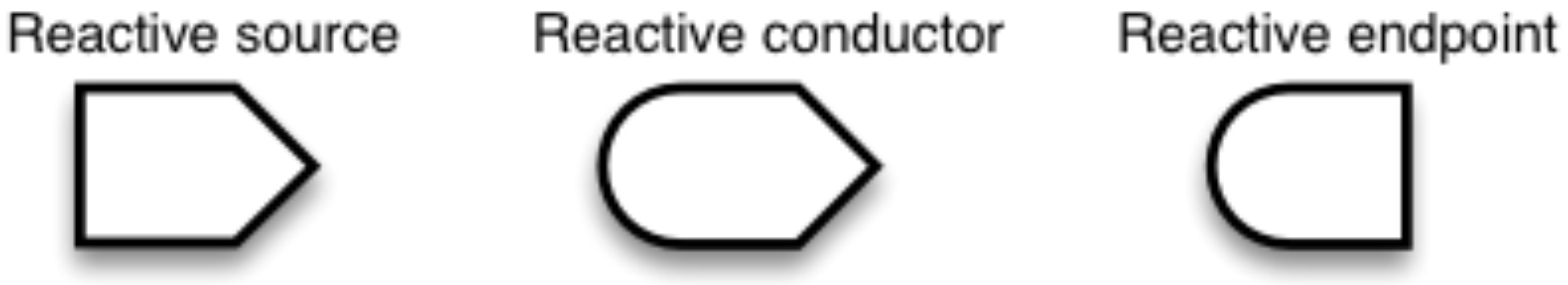
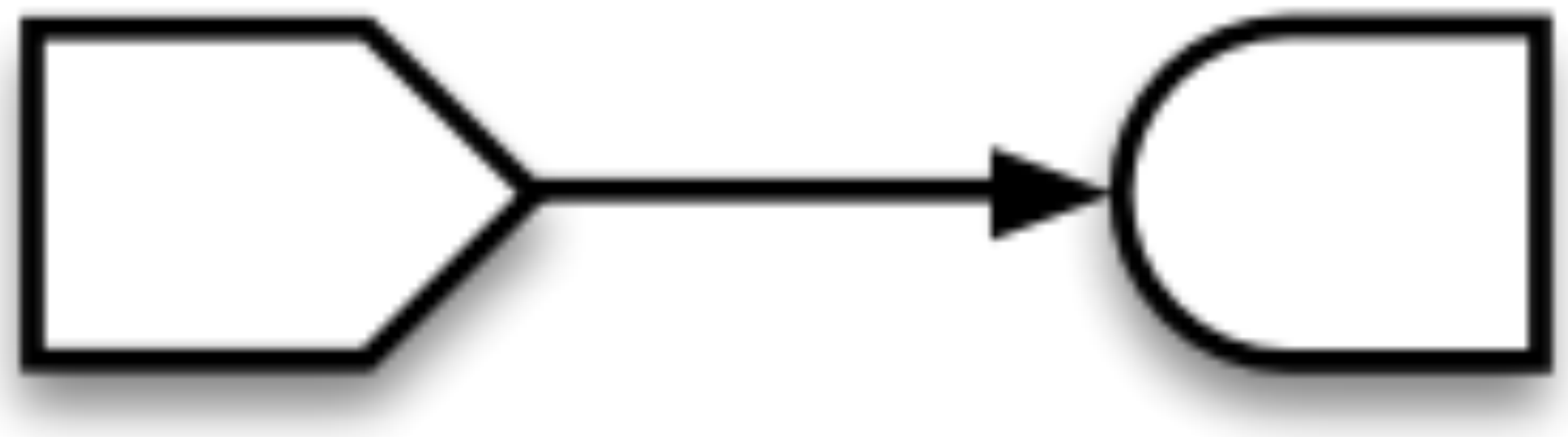
BUILDING WEB APPLICATIONS IN R WITH SHINY

Reactive elements

Reactive objects



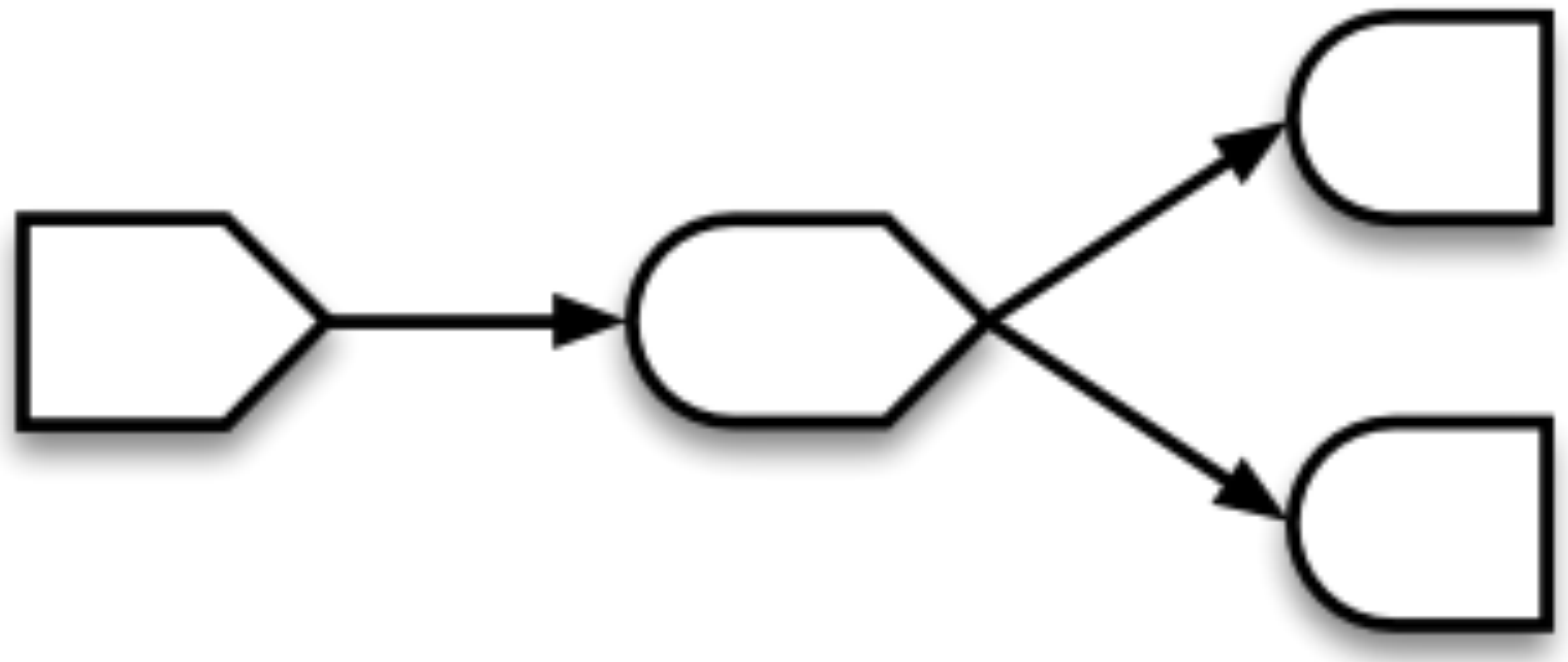
Reactive sources and endpoints

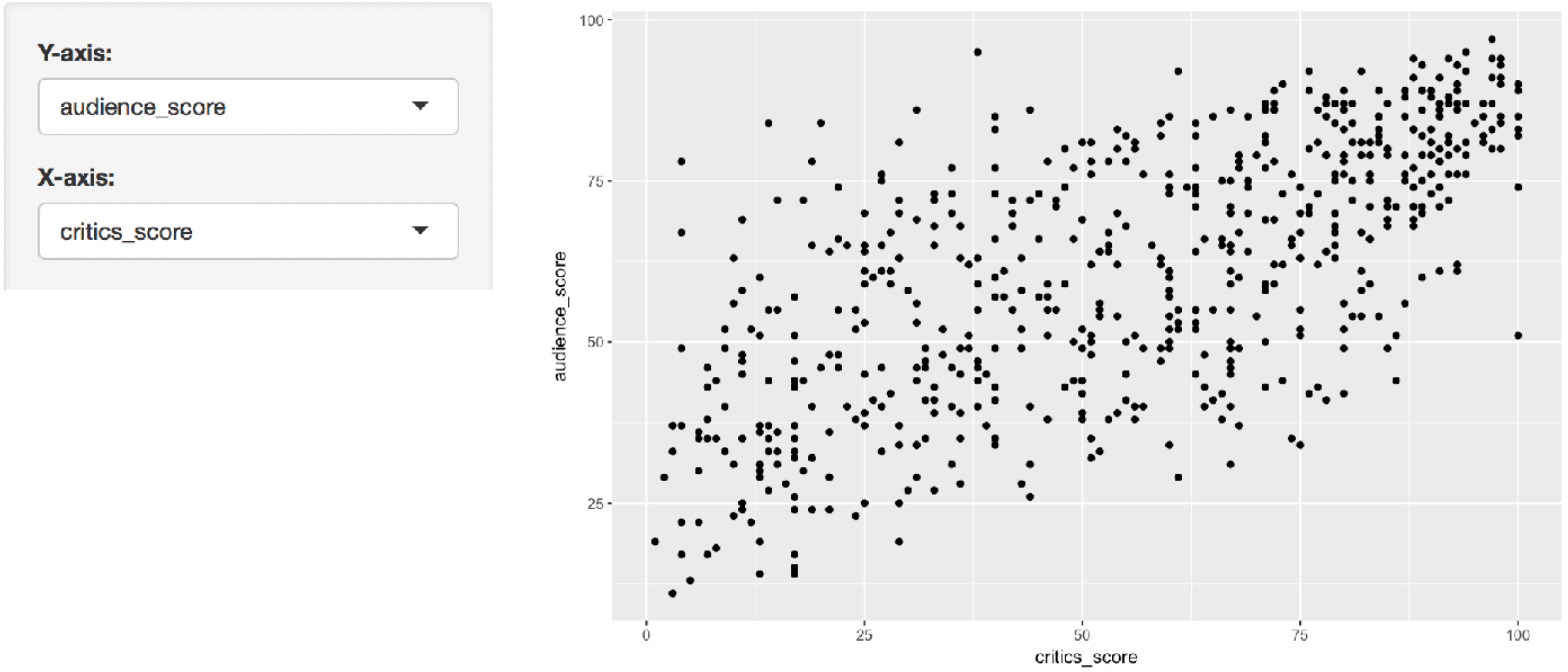


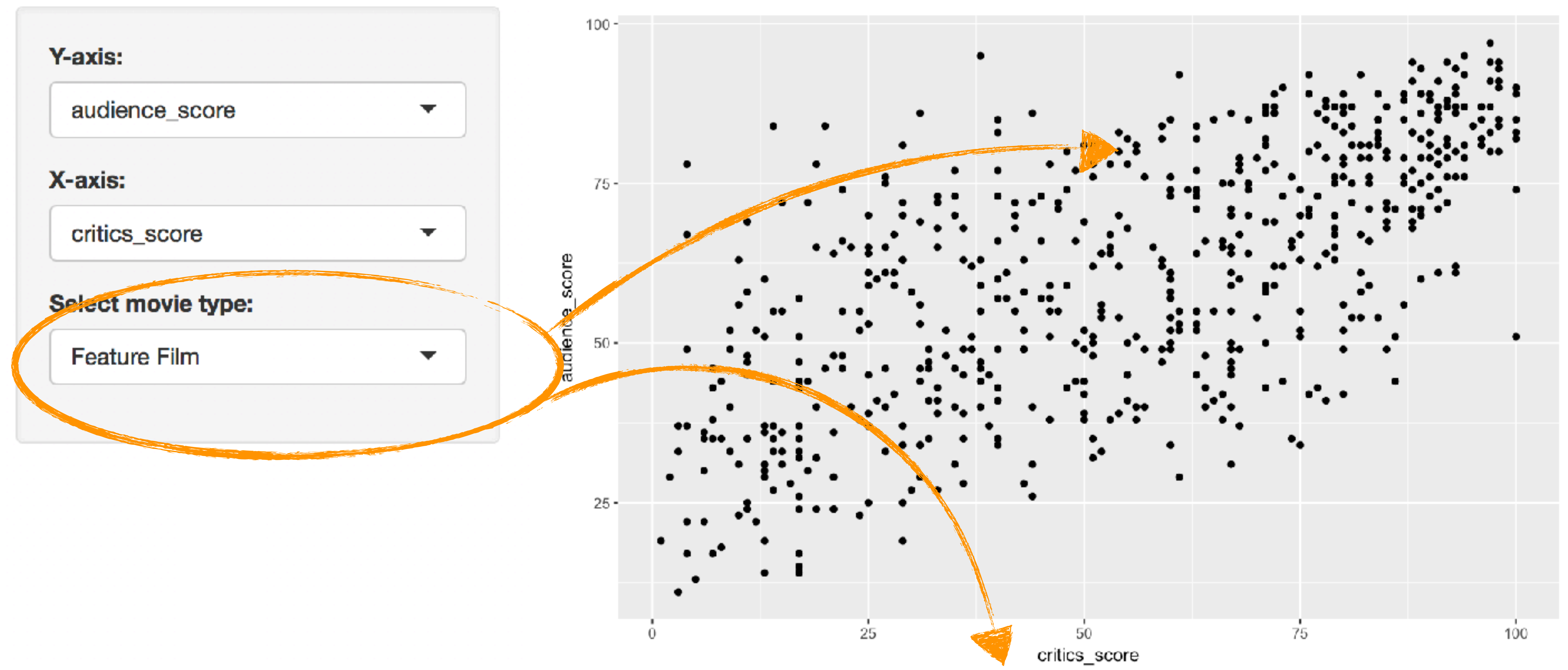
- **Reactive source:** User input that comes through a browser interface, typically
- **Reactive endpoint:** Something that appears in the user's browser window, such as a plot or a table of values
- One reactive source can be connected to multiple endpoints, and vice versa

Reactive conductors

- **Reactive counductor:** Reactive component between a source and an endpoint
- A conductor can both be a dependent (child) and have dependents (parent)
  - Sources can only be parents (they can have dependents)
  - Endpoints can only be children (they can be dependents)




Y-axis:
audience_score
X-axis:
critics_score



The plot displays the relationship between the audience and critics' scores of 591 **Feature Film** movies.

1. **ui:** Add a UI element for the user to select which type(s) of movies they want to plot.

```
# Select which types of movies to plot
selectInput(inputId = "selected_type",
            label = "Select movie type:",
            choices = levels(movies$title_type),
            selected = "Feature Film")
```

2. **server:** Filter for chosen title type and save the new data frame as a reactive expression.

```
# Create a subset of data filtering
movies_subset <- reactive({
  req(input$selected_type)
  filter(movies, title_type %in% in
})
```

Creates a **cached expression** that knows it is out of date when input changes

3. **server:** Use `movies_subset` (which is reactive) for plotting.

```
# Create scatterplot
output$scatterplot <- renderPlot({
    ggplot(data = movies_subset(),
           aes_string(x = input$x, y = input$
      geom_point()
})
```

Cached - only re-run when inputs change

3. **ui & server:** Use movies_subset (which is reactive) for printing number of observations.

```
# ui - Lay out where text should appear on app
mainPanel(
  …
  # Print number of obs plotted
  uiOutput(outputId = "n"),
  …
  )
```

```
# server - Print number of movies plotted
output$n <- renderUI({
  HTML(paste0("The plot displays the relationship between the <br>
              audience and critics' scores of <br>",
              nrow(movies_subset()),
              " <b>", input$selected_type, "</b> movies."))
})
```

Let's practice!