app.R

victoriahernandez

2024-04-12

library(shiny)  
library(ggplot2)  
library(rsconnect)

##   
## Attaching package: 'rsconnect'

## The following object is masked from 'package:shiny':  
##   
## serverInfo

library(readr)  
library(magrittr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

data <- read\_csv("train.csv")

## Rows: 1460 Columns: 81

## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities, LotConf...  
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond, Ye...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

# UI  
ui <- fluidPage(  
 titlePanel("Interactive Price vs. Living Area Chart"),  
 sidebarLayout(  
 sidebarPanel(  
 helpText("Interactive chart displaying the relationship between sale price and living area."),  
 # Checkbox group for neighborhoods  
 checkboxGroupInput(  
 "neighborhood",  
 "Neighborhood:",  
 choices = unique(data$Neighborhood),  
 selected = unique(data$Neighborhood)  
 ),  
 # Options for log transformation  
 checkboxInput("logScale", "Log-transform Sale Price", value = FALSE),  
 checkboxInput("logGrLivArea", "Log-transform Living Area", value = FALSE)  
 ),  
 mainPanel(  
 plotOutput("priceLivingAreaPlot")  
 )  
 )  
)  
  
# Server  
server <- function(input, output) {  
 filtered\_data <- reactive({  
 if (is.null(input$neighborhood) || identical(input$neighborhood, "")) {  
 dat <- data  
 } else {  
 dat <- data %>% filter(Neighborhood %in% input$neighborhood)  
 }  
 dat  
 })  
   
 # Render the plot  
 output$priceLivingAreaPlot <- renderPlot({  
 plot\_data <- filtered\_data()  
   
 # Apply log transformations if selected  
 if (input$logScale) {  
 plot\_data$SalePrice <- log(plot\_data$SalePrice)  
 }  
 if (input$logGrLivArea) {  
 plot\_data$GrLivArea <- log(plot\_data$GrLivArea)  
 }  
   
 # Generate the plot  
 ggplot(plot\_data, aes(x = GrLivArea, y = SalePrice)) +  
 geom\_point(alpha = 0.5) +  
 labs(  
 x = ifelse(input$logGrLivArea, "Log of Living Area (sq ft)", "Living Area (sq ft)"),  
 y = ifelse(input$logScale, "Log of Sale Price ($)", "Sale Price ($)"),  
 title = "Sale Price vs. Living Area"  
 ) +  
 theme\_minimal()  
 })  
}  
  
# Run the app  
shinyApp(ui = ui, server = server)

## PhantomJS not found. You can install it with webshot::install\_phantomjs(). If it is installed, please make sure the phantomjs executable can be found via the PATH variable.