gauseano.R  
URIEL ROJAS CHURQUIPA QUISPE

2025-09-17

# =============================================================================  
# APP COMPLETA SHINY - Distritos, Provincias y GRF  
# =============================================================================

library(shiny)

## Warning: package 'shiny' was built under R version 4.4.3

library(DT)

## Warning: package 'DT' was built under R version 4.4.3

##   
## Adjuntando el paquete: 'DT'

## The following objects are masked from 'package:shiny':  
##   
## dataTableOutput, renderDataTable

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.4.3

##   
## Adjuntando el paquete: 'dplyr'

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

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

library(leaflet)

## Warning: package 'leaflet' was built under R version 4.4.3

library(gstat)

## Warning: package 'gstat' was built under R version 4.4.3

library(raster)

## Warning: package 'raster' was built under R version 4.4.3

## Cargando paquete requerido: sp

## Warning: package 'sp' was built under R version 4.4.3

##   
## Adjuntando el paquete: 'raster'

## The following object is masked from 'package:dplyr':  
##   
## select

simular\_grf <- function(modelo = "Exp", var = 1, scale = 0.3, n = 50) {  
 # Crear grilla de puntos  
 grd <- expand.grid(x = 1:n, y = 1:n)  
 coordinates(grd) <- ~x + y  
 gridded(grd) <- TRUE  
   
 # Definir modelo de covarianza  
 vgm\_model <- vgm(psill = var, model = modelo, range = scale \* n, nugget = 0)  
   
 # Simulación  
 g <- gstat(formula = z ~ 1, locations = ~x + y, dummy = TRUE, beta = 0,  
 model = vgm\_model, nmax = 20)  
 sim <- predict(g, grd, nsim = 1)  
   
 rasterFromXYZ(as.data.frame(sim)[, c("x", "y", "sim1")])  
}  
  
# =============================================================================  
# UI  
# =============================================================================  
ui <- fluidPage(  
 titlePanel("📊 Análisis de Distritos y Provincias - Puno"),   
   
 tabsetPanel(  
 tabPanel("Distritos",  
 DTOutput("tabla\_distritos"),  
 downloadButton("descargar\_distritos", "⬇ Descargar Distritos")  
 ),  
   
 tabPanel("Provincias",  
 DTOutput("tabla\_provincias\_detalle"),  
 downloadButton("descargar\_provincias", "⬇ Descargar Provincias")  
 ),  
   
 tabPanel("Resumen General",  
 DTOutput("datos\_completos"),  
 downloadButton("descargar\_resumen", "⬇ Descargar Resumen")  
 ),  
   
 tabPanel("Mapa de Calor (Coroplético)",  
 sidebarLayout(  
 sidebarPanel(  
 selectInput("variable\_mapa", "Selecciona variable para el mapa:",  
 choices = c("Superficie (ha)" = "superficie\_total\_ha",  
 "Productores Totales" = "productores\_total",  
 "Agricultores que necesitan préstamo" = "agricultores\_prestamo"),  
 selected = "superficie\_total\_ha")  
 ),  
 mainPanel(  
 leafletOutput("mapa\_calor", height = 600)  
 )  
 )  
 ),  
   
 # 🚀 NUEVA PESTAÑA PARA GRF  
 tabPanel("Gaussian Random Field",  
 sidebarLayout(  
 sidebarPanel(  
 selectInput("modelo\_grf", "Modelo de Covarianza:",  
 choices = c("Exponencial" = "Exp",  
 "Gaussiano" = "Gau",  
 "Matérn" = "Mat"),  
 selected = "Gau"),  
 sliderInput("scale\_grf", "Escala (rango de correlación):",  
 min = 0.1, max = 1, value = 0.3, step = 0.1),  
 sliderInput("var\_grf", "Varianza:", min = 0.5, max = 3,  
 value = 1, step = 0.5)  
 ),  
 mainPanel(  
 plotOutput("plot\_grf", height = 600) # Mostramos el campo simulado  
 )  
 )  
 )  
 )  
)  
  
# =============================================================================  
# SERVER  
# =============================================================================  
server <- function(input, output, session) {  
   
 # =================== GRF =====================  
 output$plot\_grf <- renderPlot({  
 set.seed(123)  
 r <- simular\_grf(modelo = input$modelo\_grf,  
 var = input$var\_grf,  
 scale = input$scale\_grf,  
 n = 60)  
 plot(r, main = paste("Gaussian Random Field -", input$modelo\_grf))  
 })  
   
 # 🔹 Aquí deberías agregar el server original de distritos, provincias y resumen  
 # (yo te puse el módulo del GRF para integrarlo a tu app actual)  
}  
  
# =============================================================================  
# RUN APP  
# =============================================================================  
shinyApp(ui = ui, server = server)



