Practica—Correlación-de-Pearson-02-10-25.R

angel

2025-12-01

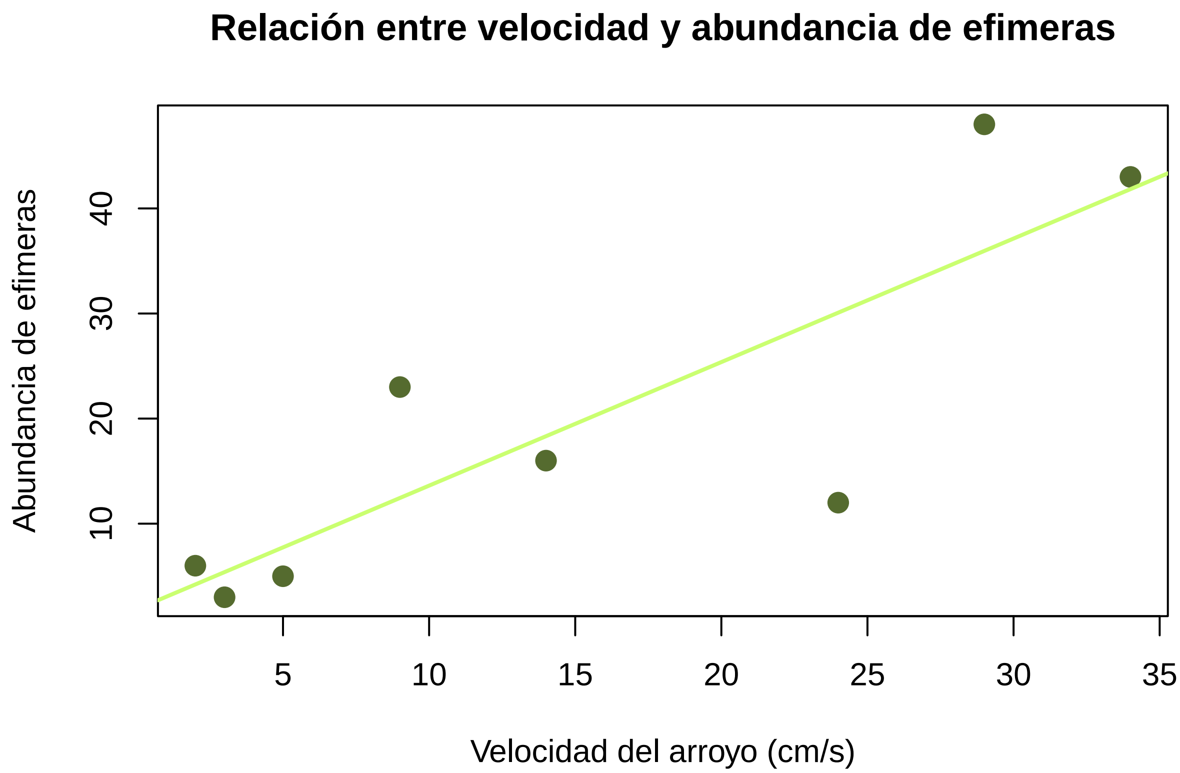
```r
################################################################################
# Practica - Correlación de Pearson
# FLOR ANGELI CRUZ ROSALES
# DR. MARCO A. GONZALEZ TAGLE
# 02/10/2025
################################################################################
options(repos = c(CRAN = "https://cloud.r-project.org"))
#Ejercicio 1: Efimeras y velocidad


resp <- data.frame(speed <- c(2, 3, 5, 9, 14, 24, 29, 34),
    abundance <- c(6, 3, 5, 23, 16, 12, 48, 43)
  )
speed <- c(2, 3, 5, 9, 14, 24, 29, 34)
abundance <- c(6, 3, 5, 23, 16, 12, 48, 43)

# Crear nuevas columnas con los rangos (1 a 8)
print("Datos de velocidad:")
```

```
## [1] "Datos de velocidad:"
```

```r
print(speed)
```

```
## [1]  2  3  5  9 14 24 29 34
```

```r
plot(speed,abundance,
       main = "Relación entre velocidad y abundancia de efimeras",
       xlab = "Velocidad del arroyo (cm/s)",
       ylab = "Abundancia de efimeras",
       pch = 16,col = "#556B2F", cex=1.5)
# Linea de temdencia
abline(lm(abundance ~ speed),
         col ="#CAFF70",
         lwd = 2)
```




```r
# Verificación de normalidad
shapiro_speed <- shapiro.test(speed)
shapiro_abundance <- shapiro.test(abundance)


# Correlación
if(shapiro_speed$p.value > 0.05 & shapiro_abundance$p.value > 0.05){
  cor_result <- cor.test(speed,abundance,method = "pearson")}else
  {cor_resul <- cor.test(speed,abundance,method = "spearman")}

# Resultados finales
print(cor_result)
```

```
##
##  Pearson's product-moment correlation
##
## data:  speed and abundance
## t = 3.8568, df = 6, p-value = 0.008393
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3442317 0.9711386
## sample estimates:
##       cor
## 0.8441408
```

```r
# Ejercicio 2: DATOS DEL SUELO
# data.frame= tabla de datos
# Columna "Gp"
# Columna "Block"
# Columna "pH"
# Columna "N"
suelo <- data.frame(Gp = c("T0", "T0", "T0", "T0", "T1", "T1", "T1"),
    Block = c(1, 2, 3, 4, 1, 2, 3),
    pH = c(5.40, 5.65, 5.14, 5.14, 5.14, 5.10, 4.70),
    N = c(0.188, 0.165, 0.260, 0.169, 0.164, 0.094, 0.100),
    Dens = c(0.92, 1.04, 0.95, 1.10, 1.12, 1.22, 1.52),
    P = c(215, 208, 300, 248, 174, 129, 117),
    Ca = c(16.35, 12.25, 13.02, 11.92, 14.17, 8.55, 8.74),
    Mg = c(7.65, 5.15, 5.68, 7.88, 8.12, 6.92, 8.16),
    K = c(0.72, 0.71, 0.68, 1.09, 0.70, 0.81, 0.39),
    Na = c(1.14, 0.94, 0.60, 1.01, 2.17, 2.67, 3.32),
    Conduc = c(1.09, 1.35, 1.41, 1.64, 1.85, 3.18, 4.16)
  )
# Analisis de correlaciones con pH
variables <- c("N", "Dens", "P", "Ca", "Mg", "K", "Na", "Conduc")
# Como hacer tabla de resultados

resultados <- data.frame(Variable= character(),
                         Coeficiente_r=numeric(),
                         Valor_p= numeric(),
                         Significancia=character(),
                         stringsAsFactors = FALSE)

# llenar con los resultados de correlación

variables <- c("N", "Dens", "P", "Ca", "Mg", "K", "Na", "Conduc")
for (variable in variables) {
cor_test <- cor.test(suelo$pH,suelo[[variable]],method = "pearson")
  }

# Mostrar tabbla con los resultados
print("Tabla de Correlaciones con pH:")
```

```
## [1] "Tabla de Correlaciones con pH:"
```

```r
print(resultados)
```

```
## [1] Variable      Coeficiente_r Valor_p       Significancia
## <0 rows> (o 0- extensión row.names)
```

```r
# Matriz de correlación completa
variables_completas <- c("pH", "N", "Dens", "P", "Ca", "Mg", "K", "Na", "Conduc")
matriz_cor <- cor(suelo[variables_completas])
print(round(matriz_cor,3))
```

```
##            pH      N   Dens      P     Ca     Mg      K     Na Conduc
## pH      1.000  0.388 -0.774  0.421  0.568 -0.611  0.371 -0.711 -0.801
```

```
## N       0.388  1.000 -0.793  0.941  0.694 -0.431  0.186 -0.852 -0.789
## Dens   -0.774 -0.793  1.000 -0.787 -0.798  0.458 -0.491  0.895  0.958
## P       0.421  0.941 -0.787  1.000  0.574 -0.451  0.440 -0.932 -0.800
## Ca      0.568  0.694 -0.798  0.574  1.000 -0.010  0.185 -0.652 -0.850
## Mg     -0.611 -0.431  0.458 -0.451 -0.010  1.000 -0.013  0.560  0.392
## K       0.371  0.186 -0.491  0.440  0.185 -0.013  1.000 -0.518 -0.507
## Na     -0.711 -0.852  0.895 -0.932 -0.652  0.560 -0.518  1.000  0.923
## Conduc -0.801 -0.789  0.958 -0.800 -0.850  0.392 -0.507  0.923  1.000
```

```
##########################################################################################################
# 1. INSTALAR EL PAQUETE
install.packages("corrplot")
```

```
## Installing package into 'C:/Users/angel/AppData/Local/R/win-library/4.5'
## (as 'lib' is unspecified)

## package 'corrplot' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\angel\AppData\Local\Temp\Rtmpi27AVb\downloaded_packages
```

```
# 2. CARGAR EL PAQUETE
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.5.2

## corrplot 0.95 loaded
```

```
# 3. CREAR LA MATRIZ DE CORRELACIÓN
names(suelo)
```

```
##  [1] "Gp"     "Block"  "pH"     "N"      "Dens"   "P"      "Ca"     "Mg"
##  [9] "K"      "Na"     "Conduc"
```

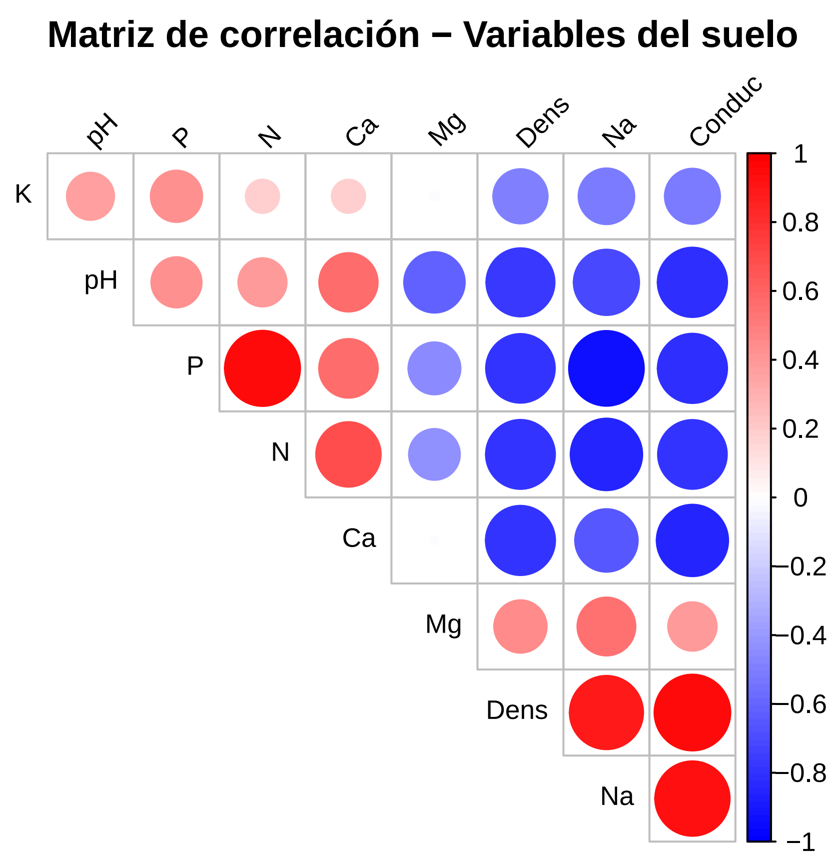
```
variables_orden <- c("K","pH","P","N","Ca","Mg","Dens","Na","Conduc")
matriz_cor <- cor(suelo[variables_orden])
colnames(matriz_cor <- rownames(matriz_cor) <- variables_orden)
```

```
## NULL
```

```
# 3. CREAR LA MATRIZ DE CORRELACIÓN

matriz_cor <- cor(suelo[variables_orden])
colnames(matriz_cor) <- rownames(matriz_cor) <- variables_orden


# Grafico de correlaciones
# Instalar el paquete
install.packages("corrplot")
```

```
## Warning: package 'corrplot' is in use and will not be installed
```

```r
# Cargar el paquete
library(corrplot)

corrplot(matriz_cor,
         method = "circle",
           type = "upper",
           tl.cex = 0.8,
           tl.col = "black",
           tl.srt= 45,
           title = "Matriz de correlación - Variables del suelo",
           mar= c(0,0,2,0),
           col = colorRampPalette(c("blue","white","red"))(100),
           diag= FALSE,
           order = "original",)
```




```r
################################################################################
# Ejercicio 3 Cuadro de anscombe
# Limpiar y configurar gráficos

graphics.off() # Cierra todos los gráficos anteriores
par(mar = c(2, 2, 2, 1)) # Márgenes mínimos: (abajo, izquierda, arriba, derecha)
par(mfrow = c(2, 2)) # Cuadrícula 2x2
par(mgp=c(1, 0.5, 0)) # Espacio entre ejes y etiquetas reducido
# Cargar datos
data(anscombe)
```

```r
# Configurar área de gráficos (2 filas, 2 columnas)
par(mfrow = c(2, 2))
# Crear los 4 gráficos
for (i in 1:4) {
    x <- anscombe[, i] # Columnas x1, x2, x3, x4
    y <- anscombe[, i + 4] # Columnas y1, y2, y3, y4
# Graficos
plot(x, y,
     main = paste("Conjunto", i),
     xlab = paste("x", i),
     ylab = paste("y", i),
     pch = 16,
     col = "#FF69B4",
     cex = 1.5,
     xlim=c(3,19),
     ylim=c(3,13))
# Línea de regresión
abline(lm(y ~ x), col = "purple", lwd = 2)
# Estadísticas
r <- round(cor(x, y), 3)
legend("topleft",
legend = paste("r =", r),
bty = "n",
text.col = "purple",
cex=1.2)
}

# Restaurar configuración normal
par(mfrow = c(1, 1))
# par(mfrow = c(2, 2)) = "Modo cuadrícula"
# par(mfrow = c(1, 1)) = "Modo normal"
```