Tarea3_E1.R

Usuario

2025-10-02

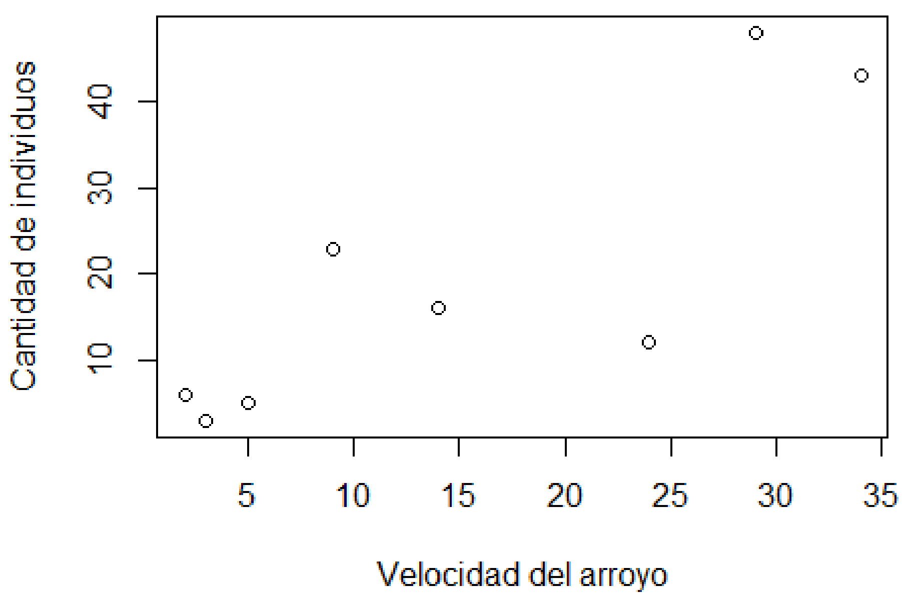
```
# Tarea 3
# JEGR
# 02/10/2025

library(ggplot2)
library(corrplot)

## corrplot 0.95 loaded

library(gt)
library(Hmisc)

##
## Adjuntando el paquete: 'Hmisc'

## The following objects are masked from 'package:gt':
##
##     html, latex

## The following objects are masked from 'package:base':
##
##     format.pval, units

#
=========================================================================
# Ejercicio 1 -----------------------------------------------------------
--
#
=========================================================================

# Creacion de dataset
speed <- c(2, 3, 5, 9, 14, 24, 29, 34)
abundance <- c(6, 3, 5, 23, 16, 12, 48, 43)

mayfly <- data.frame(speed, abundance)
View (mayfly)

print(mayfly)

##   speed abundance
## 1     2         6
## 2     3         3
## 3     5         5
## 4     9        23
```

```
## 5     14          16
## 6     24          12
## 7     29          48
## 8     34          43
```

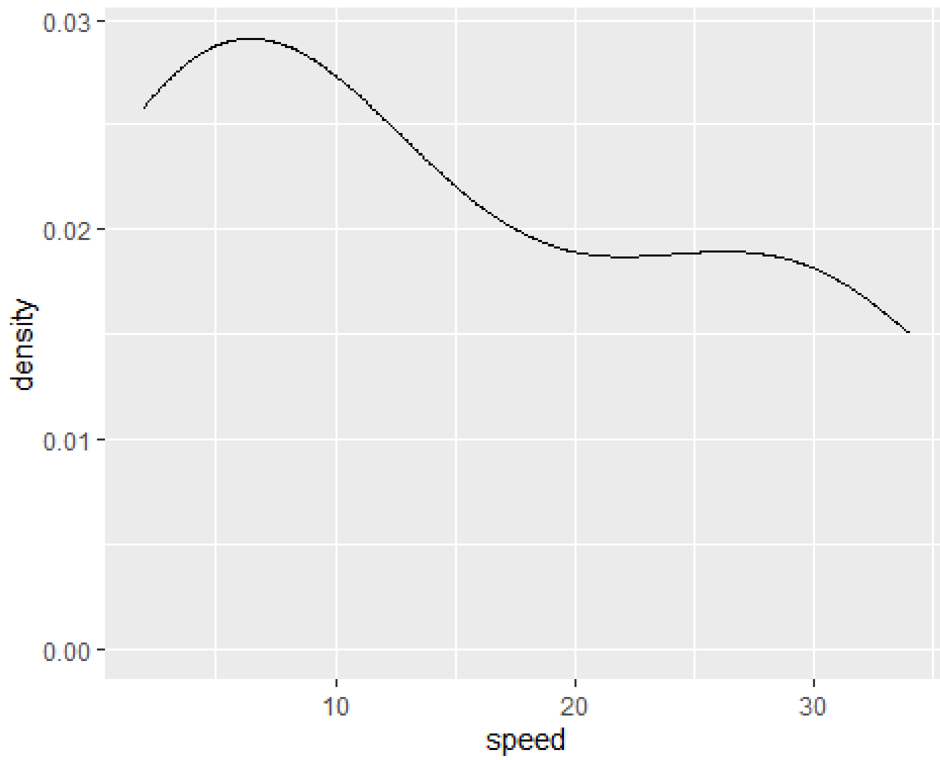
```r
# Tabla

mayfly %>%
  gt()
```

speed	abundance
2	6
3	3
5	5
9	23
14	16
24	12
29	48
34	43

```r
# Grafico de dispersion
plot(mayfly,
     xlab = "Velocidad del arroyo",
     ylab = "Cantidad de individuos")
```



```r
# Grafico densidad
ggplot(mayfly, aes(x = speed, color = abundance))+   # dist. datos en una
grafica de densidad
  geom_density()

## Warning: The following aesthetics were dropped during statistical
transformation:
## colour.
## ℹ This can happen when ggplot fails to infer the correct grouping
structure in
##   the data.
## ℹ Did you forget to specify a `group` aesthetic or to convert a
numerical
##   variable into a factor?
```



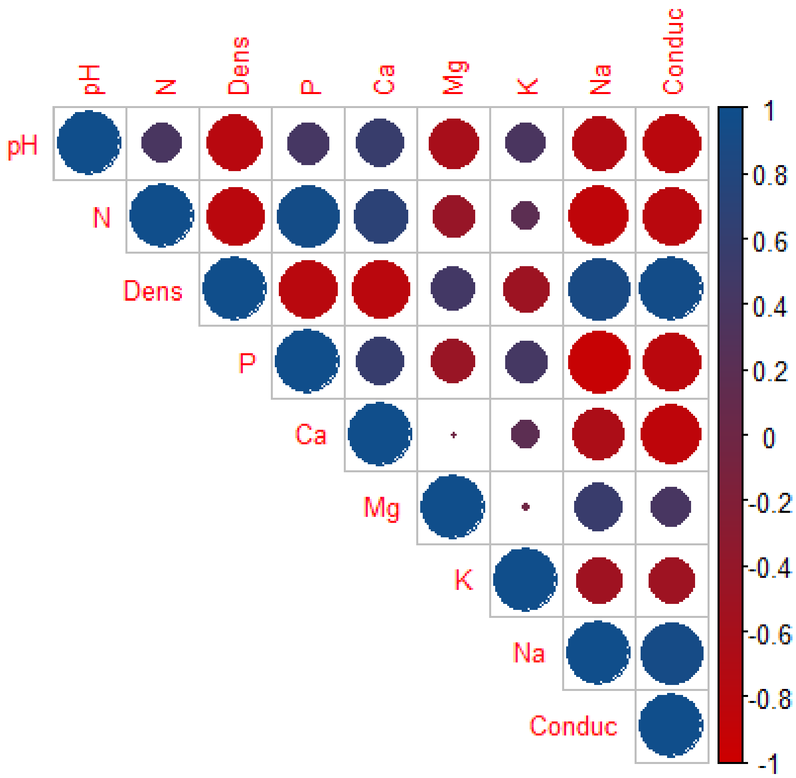

```r
# Prueba de normalidad
shapiro.test(mayfly$speed)      # Datos normales, p-value = 0.2572

##
##  Shapiro-Wilk normality test
##
## data:  mayfly$speed
## W = 0.89444, p-value = 0.2572

shapiro.test(mayfly$abundance)  # Datos normales, p-value = 0.1046

##
##  Shapiro-Wilk normality test
```

```
##
## data:  mayfly$abundance
## W = 0.85403, p-value = 0.1046

# Si significativo, p-value = 0.008393
cor.test(mayfly$speed, mayfly$abundance,
         method = "pearson")

##
##  Pearson's product-moment correlation
##
## data:  mayfly$speed and mayfly$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

#
========================================================================
# Ejercicio 2 ----------------------------------------------------------
--
#
========================================================================

# Creacion del dataset
soil <- data.frame(
  Gp    = c("T0", "T0", "T0", "T0", "T1", "T1", "T1"),
  Block= 1:7,
  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)
)

print(soil)

##   Gp Block   pH     N Dens   P    Ca   Mg    K   Na Conduc
## 1 T0     1 5.40 0.188 0.92 215 16.35 7.65 0.72 1.14   1.09
## 2 T0     2 5.65 0.165 1.04 208 12.25 5.15 0.71 0.94   1.35
## 3 T0     3 5.14 0.260 0.95 300 13.02 5.68 0.68 0.60   1.41
## 4 T0     4 5.14 0.169 1.10 248 11.92 7.88 1.09 1.01   1.64
## 5 T1     5 5.14 0.164 1.12 174 14.17 8.12 0.70 2.17   1.85
```

```
## 6 T1     6 5.10 0.094 1.22 129  8.55 6.92 0.81 2.67   3.18
## 7 T1     7 4.70 0.100 1.52 117  8.74 8.16 0.39 3.32   4.16

# Matriz de correlacion
cor_matrix <- cor(soil[,3:11],method = "pearson")

# Test de normalidad para cada par
cor_results <- rcorr(as.matrix(soil[,3:11]))

cor_matrix        # Son los coeficientes de los valores de R

##                 pH          N       Dens          P          Ca
Mg
## pH      1.0000000  0.3881145 -0.7736913  0.4206120  0.56848734 -
0.61115331
## N       0.3881145  1.0000000 -0.7926628  0.9410159  0.69412870 -
0.43103915
## Dens   -0.7736913 -0.7926628  1.0000000 -0.7865731 -0.79809646
0.45828088
## P       0.4206120  0.9410159 -0.7865731  1.0000000  0.57439198 -
0.45099416
## Ca      0.5684873  0.6941287 -0.7980965  0.5743920  1.00000000 -
0.01009406
## Mg     -0.6111533 -0.4310391  0.4582809 -0.4509942 -0.01009406
1.00000000
## K       0.3709419  0.1859458 -0.4912862  0.4397625  0.18456449 -
0.01344459
## Na     -0.7114380 -0.8524815  0.8950210 -0.9322460 -0.65215650
0.55987093
## Conduc -0.8013901 -0.7888124  0.9577017 -0.8002884 -0.84959432
0.39241421
##                  K          Na      Conduc
## pH      0.37094191 -0.7114380 -0.8013901
## N       0.18594583 -0.8524815 -0.7888124
## Dens   -0.49128624  0.8950210  0.9577017
## P       0.43976248 -0.9322460 -0.8002884
## Ca      0.18456449 -0.6521565 -0.8495943
## Mg     -0.01344459  0.5598709  0.3924142
## K       1.00000000 -0.5176140 -0.5066074
## Na     -0.51761397  1.0000000  0.9230713
## Conduc -0.50660743  0.9230713  1.0000000

cor_results$P     # Valores de p de las variables

##                 pH           N         Dens            P          Ca
Mg
## pH              NA 0.389598749 0.0412492796 0.347396647 0.18297197
0.1448353
## N       0.38959875          NA 0.0335058897 0.001571912 0.08359083
0.3342845
## Dens    0.04124928 0.033505890           NA 0.035894252 0.03145189
```

```
0.3010543
## P      0.34739665 0.001571912 0.0358942516          NA 0.17743202
0.3097948
## Ca     0.18297197 0.083590834 0.0314518872 0.177432022         NA
0.9828647
## Mg     0.14483526 0.334284451 0.3010542750 0.309794820 0.98286466
NA
## K      0.41268477 0.689757431 0.2628616908 0.323480126 0.69198279
0.9771778
## Na     0.07301106 0.014803213 0.0064764582 0.002212297 0.11240576
0.1912051
## Conduc 0.03024205 0.035005360 0.0006907938 0.030643788 0.01551290
0.3839018
##                K          Na       Conduc
## pH     0.4126848 0.073011065 0.0302420518
## N      0.6897574 0.014803213 0.0350053599
## Dens   0.2628617 0.006476458 0.0006907938
## P      0.3234801 0.002212297 0.0306437882
## Ca     0.6919828 0.112405762 0.0155129011
## Mg     0.9771778 0.191205072 0.3839018130
## K             NA 0.234092556 0.2459325415
## Na     0.2340926          NA 0.0030236919
## Conduc 0.2459325 0.003023692           NA

# Variables que nos interesan analizar con correlacion
vars <- c("N", "Dens", "P", "Ca", "Mg", "K", "Na")

# Correlaciones de pH con las demas variables
resultados <- data.frame()

for (v in vars){
  test <- cor.test(soil$pH, soil[[v]], method = "pearson")
  resultados <- rbind(resultados,
                      data.frame(Conjunto = paste("pH -", v),
                                 t = round(test$estimate, 3),
                                p_value = round(test$p.value, 4)))
}

# Tabla de resultados de las correlacions y los r
print(resultados)

##        Conjunto      t p_value
## cor      pH - N  0.388  0.3896
## cor1 pH - Dens -0.774  0.0412
## cor2     pH - P  0.421  0.3474
## cor3    pH - Ca  0.568  0.1830
## cor4    pH - Mg -0.611  0.1448
## cor5     pH - K  0.371  0.4127
## cor6    pH - Na -0.711  0.0730
```

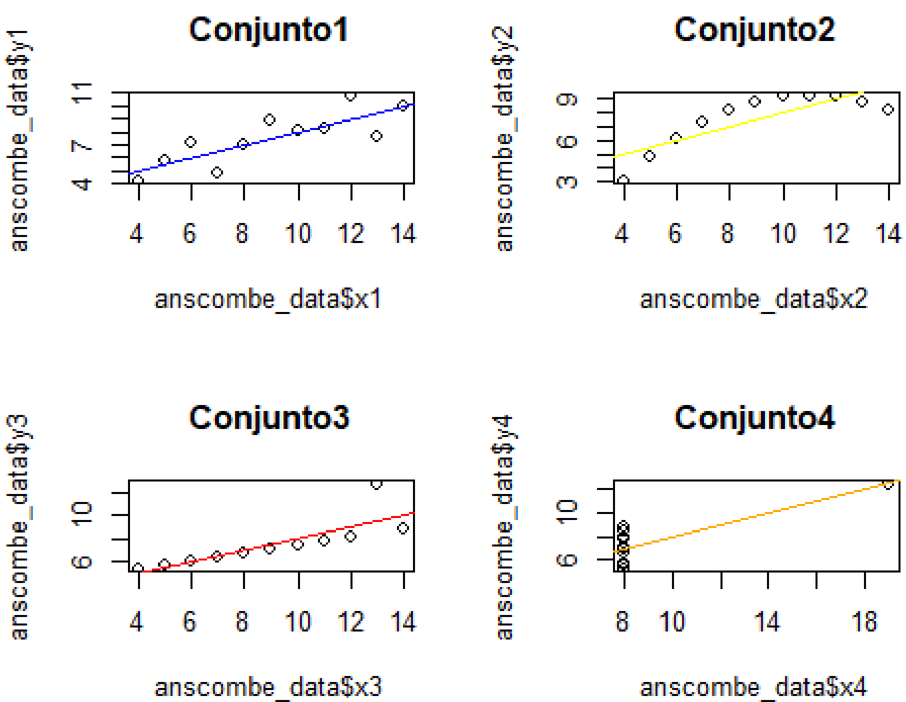
```
# Correlograma
corrplot(cor_matrix, method = "circle", type = "upper",
         tl.cex = 0.8, col =
colorRampPalette(c("red3","dodgerblue4"))(200))
```




```
#
==========================================================================
# Ejercicio 3 ------------------------------------------------------------
--
#
==========================================================================


# Base de datos incorporada a R
anscombe

##    x1 x2 x3 x4    y1   y2    y3    y4
## 1  10 10 10  8  8.04 9.14  7.46  6.58
## 2   8  8  8  8  6.95 8.14  6.77  5.76
## 3  13 13 13  8  7.58 8.74 12.74  7.71
## 4   9  9  9  8  8.81 8.77  7.11  8.84
## 5  11 11 11  8  8.33 9.26  7.81  8.47
## 6  14 14 14  8  9.96 8.10  8.84  7.04
## 7   6  6  6  8  7.24 6.13  6.08  5.25
## 8   4  4  4 19  4.26 3.10  5.39 12.50
## 9  12 12 12  8 10.84 9.13  8.15  5.56
## 10  7  7  7  8  4.82 7.26  6.42  7.91
## 11  5  5  5  8  5.68 4.74  5.73  6.89
```

```r
data("anscombe")

anscombe_data <- anscombe

# resumen estadistico
summary(anscombe_data)

##        x1             x2             x3             x4          y1
##  Min.   : 4.0   Min.   : 4.0   Min.   : 4.0   Min.   : 8   Min.   :
4.260
##  1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 6.5   1st Qu.: 8   1st Qu.:
6.315
##  Median : 9.0   Median : 9.0   Median : 9.0   Median : 8   Median :
7.580
##  Mean   : 9.0   Mean   : 9.0   Mean   : 9.0   Mean   : 9   Mean   :
7.501
##  3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.:11.5   3rd Qu.: 8   3rd Qu.:
8.570
##  Max.   :14.0   Max.   :14.0   Max.   :14.0   Max.   :19   Max.
:10.840
##        y2              y3              y4
##  Min.   :3.100   Min.   : 5.39   Min.   : 5.250
##  1st Qu.:6.695   1st Qu.: 6.25   1st Qu.: 6.170
##  Median :8.140   Median : 7.11   Median : 7.040
##  Mean   :7.501   Mean   : 7.50   Mean   : 7.501
##  3rd Qu.:8.950   3rd Qu.: 7.98   3rd Qu.: 8.190
##  Max.   :9.260   Max.   :12.74   Max.   :12.500

# Correlacion de las distintas variables
cor(anscombe_data$x1, anscombe_data$y1)

## [1] 0.8164205

cor(anscombe_data$x2, anscombe_data$y2)

## [1] 0.8162365

cor(anscombe_data$x3, anscombe_data$y3)

## [1] 0.8162867

cor(anscombe_data$x4, anscombe_data$y4)

## [1] 0.8165214

# Graficas
par(mfrow=c(2,2))
plot(anscombe_data$x1, anscombe_data$y1,
     main="Conjunto1")
abline(lm(y1 ~ x1, data = anscombe_data), col = "blue")

plot(anscombe_data$x2, anscombe_data$y2,
```

```r
     main = "Conjunto2")
abline(lm(y2 ~ x2, data = anscombe_data), col = "yellow")

plot(anscombe_data$x3, anscombe_data$y3,
     main = "Conjunto3")
abline(lm(y3 ~ x3, data = anscombe_data), col = "red")

plot(anscombe_data$x4, anscombe_data$y4,
     main = "Conjunto4")
abline(lm(y4 ~ x4, data = anscombe_data), col = "orange")
```




```r
par(mfrow=c(1,1))

# Cuadro
# Calculo de correlaciones y sus modelos de regresion
resultados_ans <- data.frame()

for (i in 1:4)  {
  x <- anscombe_data[[paste0("x", i)]]
  y <- anscombe_data[[paste0("y", i)]]

  # Correlacion
  r <- cor(x, y)

  # Regresion lineal
  modelo <- lm(y ~ x)
```

```
  resumen <- summary(modelo)

  resultados_ans <- rbind(resultados_ans, data.frame(
    Conjunto = paste("Conjunto", i),
    Media_x  = mean(x),
    Media_y  = mean(y),
    Var_x    = var(x),
    Var_y    = var(y),
    Correlacion = round(r, 3),
    Pendiente = round(coef(modelo)[2],3),
    Intercepto = round(coef(modelo)[1],3),
    R2      = round(resumen$r.squared, 3)
  ))
}

print(resultados_ans)

##      Conjunto Media_x  Media_y Var_x    Var_y Correlacion Pendiente
Intercepto
## x  Conjunto 1       9 7.500909    11 4.127269       0.816       0.5
3.000
## x1 Conjunto 2       9 7.500909    11 4.127629       0.816       0.5
3.001
## x2 Conjunto 3       9 7.500000    11 4.122620       0.816       0.5
3.002
## x3 Conjunto 4       9 7.500909    11 4.123249       0.817       0.5
3.002
##       R2
## x  0.667
## x1 0.666
## x2 0.666
## x3 0.667
```