

```

fichero = read.csv2("prueba.csv")

len = function(list){
  count = 0
  for (element in list){
    count = count + 1
  }
  count
}

distancias = fichero$Numeros

len(distancias)

## [1] 12

bubble = function(list){
  n = len(list)
  for (i in 2:n){
    for (j in 1:(n-1)){
      if (list[j] > list[j+1]){
        temp = list[j]
        list[j] = list[j+1]
        list[j+1] = temp
      }
    }
  }
  list
}

distanciasordenadas = bubble(distancias)
distanciasordenadas

## [1] 13 15 16 20 20 22 27 29 30 33 34 42

rank = function(list){
  ordered_list = bubble(list)
  ordered_list[len(ordered_list)] - ordered_list[1]
}

rango = rank(distanciasordenadas)
rango

## [1] 29

absolute_freq = function(list){

```

---

```

ordered_list = bubble(list)
n = len(ordered_list)
elements = vector()
frequencies = vector()
i = 1
while (i <= n){
  actual_element = ordered_list[i]
  elements = append(elements, actual_element)
  actual_freq = 0
  j = i
  while(j <= n & actual_element == ordered_list[j]){
    actual_freq = actual_freq + 1
    j = j+1
  }
  frequencies = append(frequencies, actual_freq)
  i = j
}
rbind(elements, frequencies)
}
frecuencia_abs = absolute_freq(distancias)
frecuencia_abs

##           [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
## elements    13   15   16   20   22   27   29   30   33   34   42
## frequencies    1    1    1    2    1    1    1    1    1    1    1

relative_freq = function(list){
  f_abs = absolute_freq(list)
  elements = f_abs[1,]
  abs_fvalues = f_abs[2,]
  rbind(elements,abs_fvalues/len(list))
}

frecuencia_rel = relative_freq(distancias)
frecuencia_rel

##           [,1]           [,2]           [,3]           [,4]           [,5]           [,6]
## elements 13.00000000 15.00000000 16.00000000 20.0000000 22.00000000 27.00000000
##           0.08333333 0.08333333 0.08333333 0.1666667 0.08333333 0.08333333
##           [,7]           [,8]           [,9]           [,10]           [,11]
## elements 29.00000000 30.00000000 33.00000000 34.00000000 42.00000000
##           0.08333333 0.08333333 0.08333333 0.08333333 0.08333333

acum_absolute_freq = function(list){
  f_abs = absolute_freq(list)

```

---

```

    elements = f_abs[1,]
    abs_fvalues = f_abs[2,]
    acum_abs_fvalues = vector()
    acum = 0
    for (i in 1:len(elements)){
        acum = acum + abs_fvalues[i]
        acum_abs_fvalues = append(acum_abs_fvalues, acum)
    }
    rbind(elements, acum_abs_fvalues)
}
frecuencia_abs_acum = acum_absolute_freq(distancias)
frecuencia_abs_acum

##           [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
## elements      13  15  16  20  22  27  29  30  33  34  42
## acum_abs_fvalues  1   2   3   5   6   7   8   9  10  11  12

acum_relative_freq = function(list){
    f_rel = relative_freq(list)
    elements = f_rel[1,]
    rel_fvalues = f_rel[2,]
    acum_rel_fvalues = vector()
    acum = 0
    for (i in 1:len(elements)){
        acum = acum + rel_fvalues[i]
        acum_rel_fvalues = append(acum_rel_fvalues, acum)
    }
    rbind(elements, acum_rel_fvalues)
}
frecuencia_rel_acum = acum_relative_freq(distancias)
frecuencia_rel_acum

##           [,1]      [,2]      [,3]      [,4] [,5]      [,6]
## elements      13.00000000 15.0000000 16.00 20.0000000 22.0 27.0000000
## acum_rel_fvalues 0.08333333 0.1666667 0.25 0.4166667 0.5 0.5833333
##           [,7]      [,8]      [,9]      [,10] [,11]
## elements      29.0000000 30.00 33.0000000 34.0000000 42
## acum_rel_fvalues 0.6666667 0.75 0.8333333 0.9166667 1

mean = function(list){
    total = 0
    n = len(list)
    for (i in 1:n){
        total = total + list[i]
    }
}

```

---

```

        mean = total / n
        mean
    }

media = mean(distancias)
media

## [1] 25.08333

mode = function(list){
    frequencies = absolute_freq(list)
    elements = frequencies[1,]
    freq_values = frequencies[2,]
    actual_mode = 0
    actual_mode_val = 0
    for (i in 1:len(elements)){
        if (freq_values[i] > actual_mode_val){
            actual_mode_val = freq_values[i]
            actual_mode = elements[i]
        }
    }
    actual_mode
}

moda = mode(distancias)
moda

## [1] 20

median = function(list){
    n = len(list)
    if (n%%2 == 0){
        median = (list[n/2] + list[(n/2)+1]) / 2
    }
    else{
        median = list[(n+1)/2]
    }
    median
}

mediana = median(distancias)
mediana

## [1] 24.5

```

---

```

standard_desv = function(list){
  mean = mean(list)
  n = len(list)
  add = 0
  for (i in 1:n){
    add = add + ((list[i] - mean)^2)
  }
  sqrt(add/n)
}

desviacion = standard_desv(distancias)
desviacion

## [1] 8.47996

variance = function(list){
  desv = standard_desv(list)
  var = desv^2
  var
}

varianza = variance(distancias)
varianza

## [1] 71.90972

quant = function(list, c){
  ordered_list = bubble(list)
  n = len(list)
  if (c < 0){
    quant = 0
  }
  else{
    if((n*c)%1 == 0){
      quant = (ordered_list[(n*c)] + ordered_list[(n*c) + 1]) / 2
    }
    else {
      int_prod = floor(n*c)
      quant = ordered_list[int_prod + 1]
    }
  }
  quant
}

cuartil1 = quant(distancias,0.25)

```

---

```
cuartil2 = quant(distancias,0.5)
cuartil3 = quant(distancias,0.75)
cuartil1

## [1] 18

cuartil2

## [1] 24.5

cuartil3

## [1] 31.5
```