

Técnicas y Herramientas Modernas I- Introducción a lenguajes de programación de alto nivel (paradigma de objetos) – Programación en R

Franco Font^{a,1,*}, Carlos Font^a, Ignacio Ortiz Maldonado^{a,2}, Nicolas Guntsche^{a,2}, Sebastián Chiapetta^{a,2}

^aCentro universitario M5500 Mendoza

^bCentro universitario M5500 Mendoza

Abstract

Este módulo de trabajo se realizó con el objetivo de aprender y practicar sobre el lenguaje de programación estadístico R. A medida que se resuelven los diferentes ejercicios se utilizan diferentes herramientas y recursos del R para ser resueltos.

Keywords: vector, serie, iteración, modelo matemático

EJERCICIO N 1: Generar un vector secuencia

Se compararon dos códigos que tienen la misma finalidad para ver cuánto tiempo toman cada uno, con el fin de observar su rendimiento y eficiencia. Ambos códigos sirven para generar un vector con una secuencia numérica.

##Codigo generado con for:

```
A<- c()
start_time<-Sys.time()
for (i in 1:50000) {A[i]<-(i*2)}
head (A)
```

```
## [1]  2  4  6  8 10 12
```

```
tail(A)
```

```
## [1] 99990 99992 99994 99996 99998 100000
```

```
end_time<- Sys.time()
end_time-start_time
```

```
## Time difference of 0.463351 secs
```

##Codigo generado con R

*Corresponding author

Email addresses: francofont7@gmail.com (Franco Font), carlosfont32@gmail.com (Carlos Font), ortizmaldonadoignacio@gmail.com (Ignacio Ortiz Maldonado), nicoguntsche@gmail.com (Nicolas Guntsche), sebachiapetta20@gmail.com (Sebastián Chiapetta)

¹Esta es la primera nota del autor.

²

```
start_time2<- Sys.time()
A<-seq(1,100000,2)
head(A)
```

```
## [1] 1 3 5 7 9 11
```

```
tail(A)
```

```
## [1] 99989 99991 99993 99995 99997 99999
```

```
end_time2 <- Sys.time()
end_time2 - start_time2
```

```
## Time difference of 0.04779696 secs
```

EJERCICIO N 2: Implementacion de serie de Fibonacci

En matemáticas, la sucesión o serie de Fibonacci es la siguiente sucesión infinita de números naturales: “0, 1, 1, 2, 3, 5, 8... 89, 144, 233... 00 La sucesión comienza con los números 0 y 1,2 a partir de estos, «cada término es la suma de los dos anteriores», es la relación de recurrencia que la define. A los elementos de esta sucesión se les llama números de Fibonacci.

$$f_0 = 0; f_1 = 1; f_{n+1} = f_n + f_{n-1}$$

Como consigna de este ejercicio se debe buscar cuantas iteraciones se deben realizar para que el valor de la serie alcance un número mayor a 1.000.000

```
f0<-0
f1<-1
it<-0
f2<-0
vec<- c(f0,f1)
while(f2<=1000000){
  it<-(it+1)
  f2<-(f0+f1)
  vec<- c(vec,f2)
  f0<-f1
  f1<-f2
}
it
```

```
## [1] 30
```

```
tail(vec)
```

```
## [1] 121393 196418 317811 514229 832040 1346269
```

Se puede observar que el algoritmo requiere al menos 30 iteraciones para poder superar el millón.

EJERCICIO N 3:Ordenamiento de un vector por el método burbuja

En el siguiente ejercicio se realiza el ordenamiento de los valores numéricos de un vector mediante el método burbuja vs. el método sort nativo de R.

```

library(microbenchmark)
x<-sample(1:100,100)
mbm<-microbenchmark(
  ##método de ordenamiento directo o burbuja
  "burbuja"={
    burbuja<-function(x){
      n<-length(x)
      for(j in 1:(n-1)){
        for(i in 1:(n-j)){
          if(x[i]>x[i+1]){
            temporal<-x[i]
            x[i]<-x[i+1]
            x[i+1]<-temporal
          }
        }
      }
      return(x)
    }
  },
  ##método de R sort
  "sort"={
    sort(x)
  }
)
mbm

```

```

## Unit: microseconds
##      expr      min       lq      mean   median      uq      max neval
## burbuja 4507.4 6238.20 16310.840 8845.15 14421.05 173906.6   100
##      sort   136.9  207.95   917.825  290.80   399.05  18022.9   100

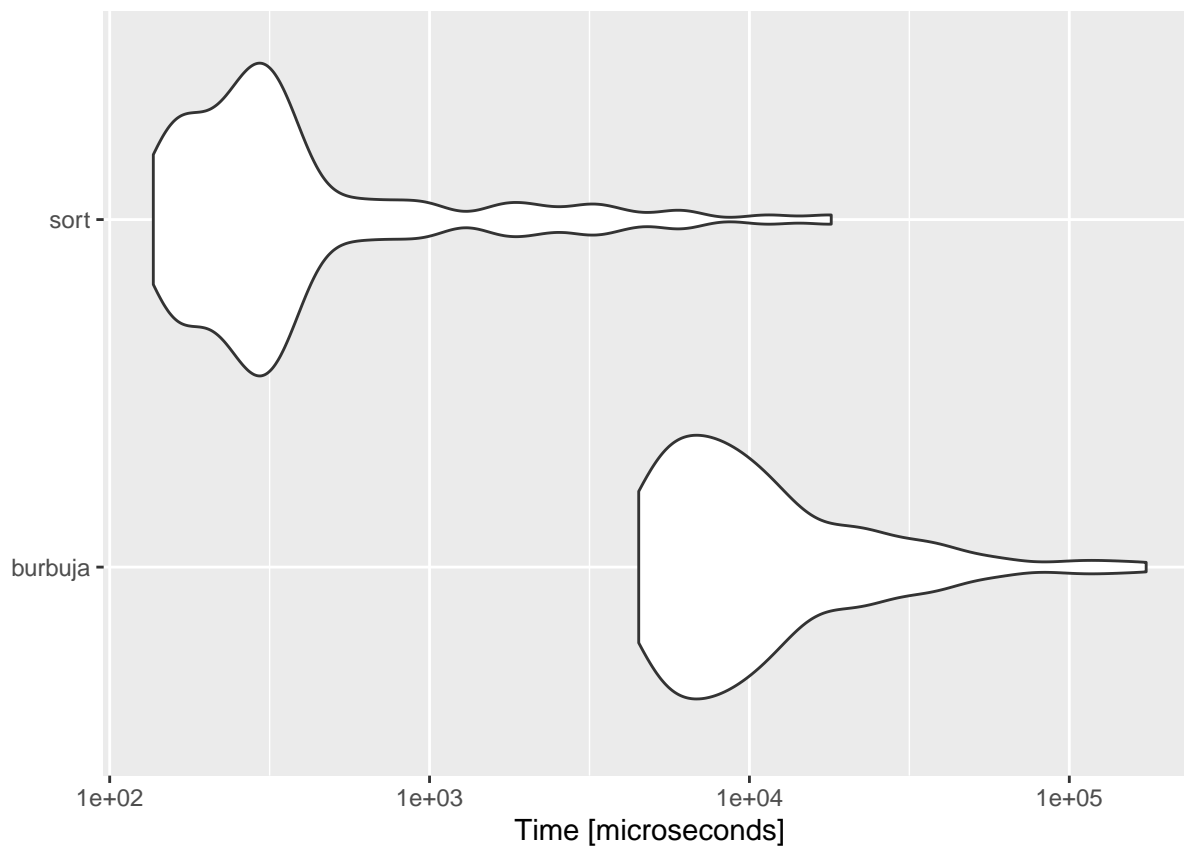
```

```

library(ggplot2)
autoplot(mbm)

```

Coordinate system already present. Adding new coordinate system, which will replace the existing one



EJERCICIO N 4: Progresión geométrica de los casos de Covid-19

En el siguiente ejercicio se resuelve, mediante un modelo matemático, la incógnita del virus en la pandemia, determinando así cuántos días serán necesarios para que se contagien 40 millones de habitantes. Además haremos uso de los datos tomados del archivo “casos” de los contagios en Argentina. Como dato inicial se utiliza según el 04/07/2020 una cantidad inicial de contagios de 1175.

```
f1<- 1175
f2<-0
dia<-0
vector<- c(f1)
F<-1.62

while(f2<=40000000){
  dia<-dia+1
  f2<-F*f1
  vector<-c(vector,f2)
  f1<-f2
}
dia
```

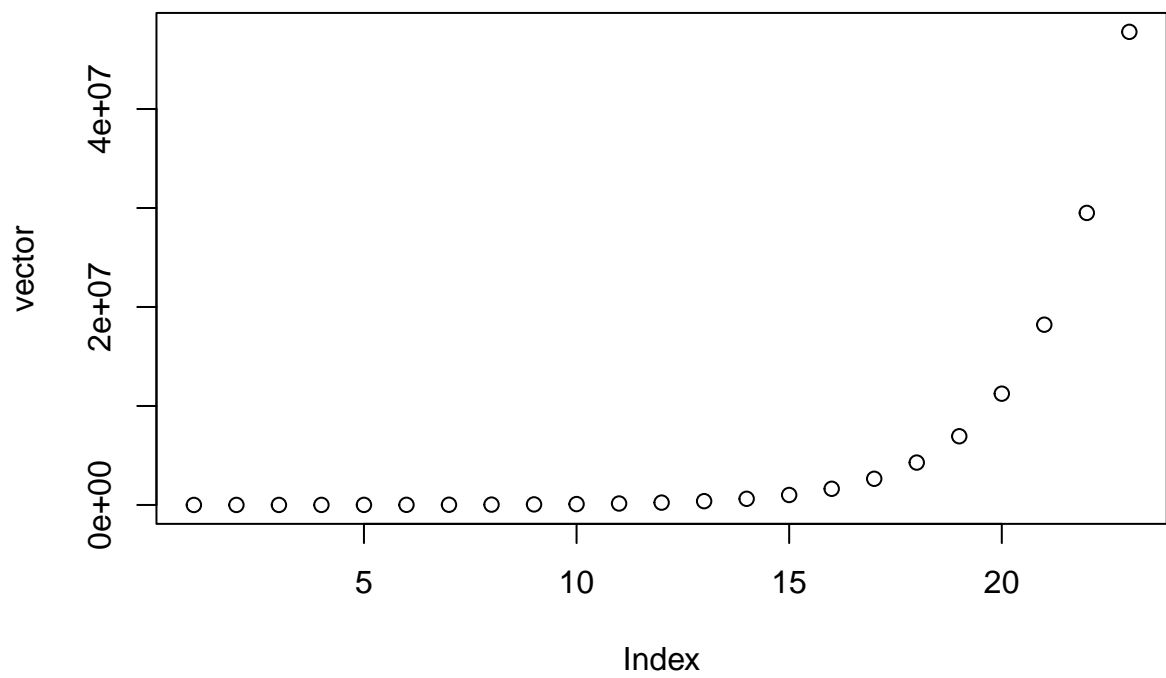
```
## [1] 22
```

```
vector
```

```
## [1] 1175.000 1903.500 3083.670 4995.545 8092.784
```

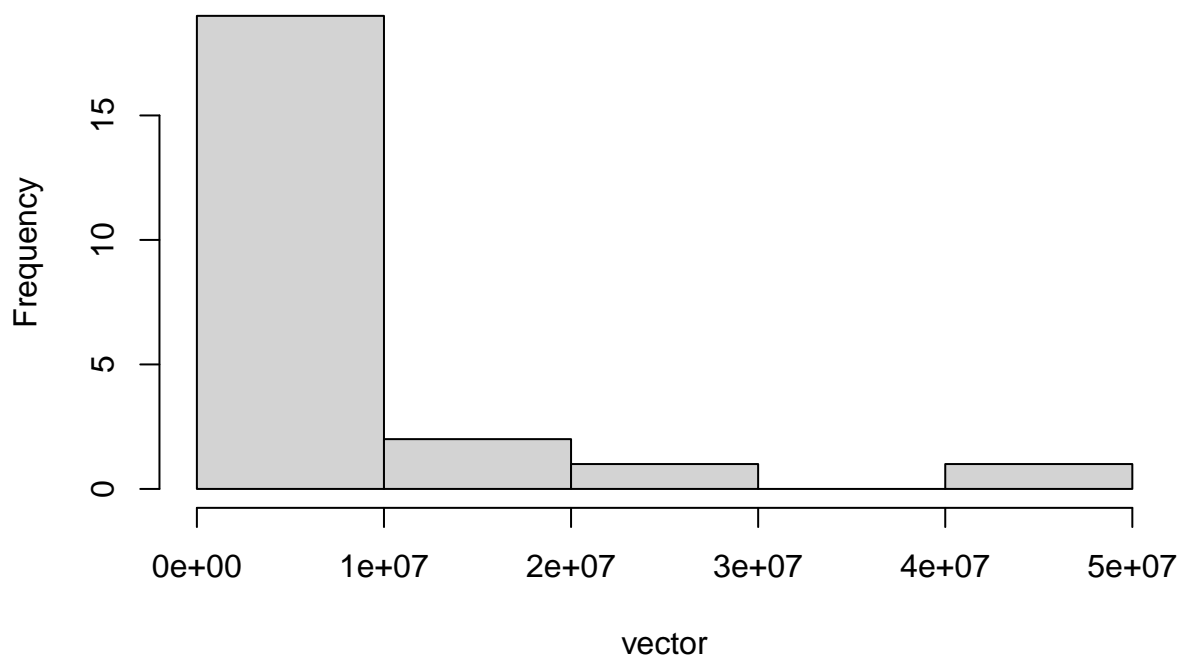
```
## [6] 13110.309 21238.701 34406.696 55738.847 90296.933
## [11] 146281.031 236975.270 383899.937 621917.898 1007506.995
## [16] 1632161.332 2644101.358 4283444.201 6939179.605 11241470.960
## [21] 18211182.956 29502116.388 47793428.549
```

```
plot(vector)
```



```
hist(vector)
```

Histogram of vector



Se puede observar en el vector que almacenó la cantidad de casos en Argentina por día, según el modelo matemático, al día 22 recién se superará la cantidad de infectados buscada.

```
plot(density(vector))
```

