

Trabajando con R y Python

antonio

7/7/2020

Reticulate

```
#utilizamos una libreria de Python en R
library(reticulate)
use_python("C:/Users/anton/.conda/pkgs/python-3.7.7-h81c818b_4") #Carga los paquetes de la instalacion
#py_install("nombre del paquete a instalar") # ]Instalar un paquete faltante
os = import("os") # Acceder a las libreria sde python en R
```

```
## Warning: Python 'C:/Users/anton/.conda/pkgs/python-3.7.7-h81c818b_4/
## python.exe' was requested but 'C:/Users/anton/AppData/Local/r-miniconda/envs/r-
## reticulate/python.exe' was loaded instead (see reticulate::py_config() for more
## information)
```

```
os$listdir(".") # Lista el contenido del directorio de trabajo
```

```
## [1] "cuadrado.py" "Reticulate_r_Phyton.pdf"
## [3] "Reticulate_r_Phyton.Rmd" "tema1"
## [5] "tema10" "tema11"
## [7] "tema12" "tema2"
## [9] "tema3" "tema4"
## [11] "tema5" "tema6"
## [13] "tema8" "tema9"
```

#Ejecutar una funcion de python en R con la funcion source_python

```
source_python("cuadrado.py")
multiplicacion(3,3)
```

```
## [1] 9
```

conversion de objetos de un lenguaje a otro

```
np = import("numpy", convert =FALSE) #Para trabajar objetos nativos de python
x= np$array(c(1:4))
sum = x$cumsum()
```

```
#imprimir la lista de numeros nativa python creada en R
print(sum)
```

```
## [ 1  3  6 10]
```

```
#Conversion explicita de R a Python  
py_to_r(sum)
```

```
## [1]  1  3  6 10
```

Opcion Ayuda

```
#Para consultar la documentacion como R o Python las conversiones Reticulate son en automaticas por que  
py_help(os$chdir)  
help(py_to_r)
```

```
## starting httpd help server ... done
```

Array

```
#Declarando un array  
#Creamos un array objeto python , la opcion order=C (utiliza opcion de guardado en en memoria de "C" en  
a <- np_array(c(1:10), dtype="float16", order="C")  
a
```

```
## [ 1.  2.  3.  4.  5.  6.  7.  8.  9. 10.]
```

```
datos <- iris  
head(datos)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 1          5.1          3.5          1.4          0.2  setosa  
## 2          4.9          3.0          1.4          0.2  setosa  
## 3          4.7          3.2          1.3          0.2  setosa  
## 4          4.6          3.1          1.5          0.2  setosa  
## 5          5.0          3.6          1.4          0.2  setosa  
## 6          5.4          3.9          1.7          0.4  setosa
```

```
#Convertimos el objeto datos a Python para manipularlo en codigo Python  
datos_py <- r_to_py (datos) #Del data frame de R pasamos a Python con r_to_py
```