R Project DEMO

\_Cammands:

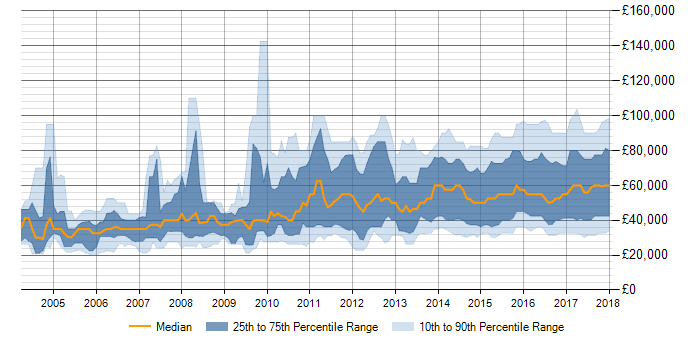
Presentation



pRESENTATION OF |R

Open source, trend ( ML, IA, DS)





Lists of Functions:

X <- c(1,2,3,4,6,0,-3)

x[x<3 & x>10] this is equivalent to the elaborated filter

Plotting:

Hist, linear, DENSITY ,

\_

Idex:

R cheatSheet

Workshop Plan

|  |  |
| --- | --- |
| Activities |  |
| Presnetation of R & R IDE ( studio) | Simple presnetation   * History * Domain of usage ( medical , industry ,retail , , ) * Trend (hist) |
| Data: vector, lists, matrix, dataframe | Basic Operation  a<- 3+2  Variable : v<- 1:10  v <- seq(from=2, to 13, by=2)  Vectors: v<- c(1, 3, 5, 8)    class(variable)  v<- c(1, 3, 5, 8**:**12)  v[c(1:2, 6:8)]  **Processing vectors value:**  Displaying values of an array: v[2], v[length(v)]  Filtering Values : v[v>3]  **Compar with advanced filter on a set of values ( name, age 0 with a filer o na criteria ( age>25)**  ength(v)  mean(v)  sum(v)/length(v)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Functions:  Max, min , mean, sum , length |
| How to use help  Example : ??seq  Compare : excel hidden formula with R programing instructions |  |
| To search: | v[2:4 && 6:9] |
| Function : | gtThan <- function(a){  v[v>a]  }    gtThan(5) |
| Data Frame | * Load csv file, file.choose,, set working directory * Display , simpler, filter. length, sum , mean, * Order, Aggregation   Exmaple:  planets\_df[planets\_df$rings ==T, 1:3]  planets\_df[planets\_df$rings ==T, ]  planets\_df[planets\_df$rings ==T, c(“name”, “type”)]  planets\_df[planets\_df$diameter>1.0 , ] |

<https://www.datacamp.com/community/tutorials/15-easy-solutions-data-frame-problems-r>

https://campus.datacamp.com/courses/free-introduction-to-r/chapter-5-data-frames?ex=10

|  |  |
| --- | --- |
| Excel/ Access/ Programming | R |
| filter | filter(df, f condition )  Ex:  Df <- c(1,2,4,7,9,0)  F <- filter (Df, capital !=””) |
| sort | Sort function  Ex : sort(x, decreasing = FALSE, ...) |
| graph | Have different ways to graph data  Line chart Ex: cars <- c(1, 3, 6, 4, 9)  plot(cars)  **lines(trucks, type="o", pch=22, lty=2, col="red")**  Second way :  Bar chart ex :  barplot(cars)  Third way :  Histograms :  hist(suvs)  Fourth way:  Pie chart :  pie(cars)  Fifth way :  Dot charts:  dotchart(t(autos\_data)) |
| Algorithmic structure |  |
| IF statement | Ex:  if(hours > 100) net.price <- net.price \* 0.9 |
| Loop Statement | Ex :  for (year in c(2010,2011,2012,2013,2014,2015)){  print(paste("The year is", year))  }  Second way – using next :  for (i in 1:10) {  if (!i %% 2){  next  }  print(i)  } |
| Variable | Ex :  A < -- 10 |
| Tables | Vector, matrix, Data frame, list, Factor, |
|  |  |

<https://campus.datacamp.com/courses/data-visualization-with-ggplot2-1/chapter-1-introduction-f5aef4fe-3bbb-4e18-b29a-8fb1361d7835?ex=8>

|  |  |
| --- | --- |
| Visualization |R |  |
| ggplot(mtcars, aes(x = wt, y = mpg, shape = disp)) + geom\_point() |  |
| ggplot( iris, aes(x = special.Length, y=special.Width))+ geo\_jitter(alpha =0.6) +  facet\_grid(. ~ Species) +  stat\_smooth(method =”lm” , se=F, col=”red”) |  |

keybor ard shorcut:

Change directory ⌘ O

**LINKS:**

<https://campus.datacamp.com/courses/cleaning-data-in-r/chapter-1-introduction-and-exploring-raw-data?ex=5>

<https://medium.com/@GalarnykMichael/accessing-data-from-twitter-api-using-r-part1-b387a1c7d3e>

<https://www.mockaroo.com>

<http://databasetestdata.com>

<http://www.lemonde.fr/connaitre-et-comprendre-les-metiers-de-demain/article/2018/01/19/la-big-data-intelligence-prochaine-etape-pour-les-datamasters_5244160_5240459.html>

Scenario:

\Presentsaiton

Processing Data 50 records

Go through fundamental function

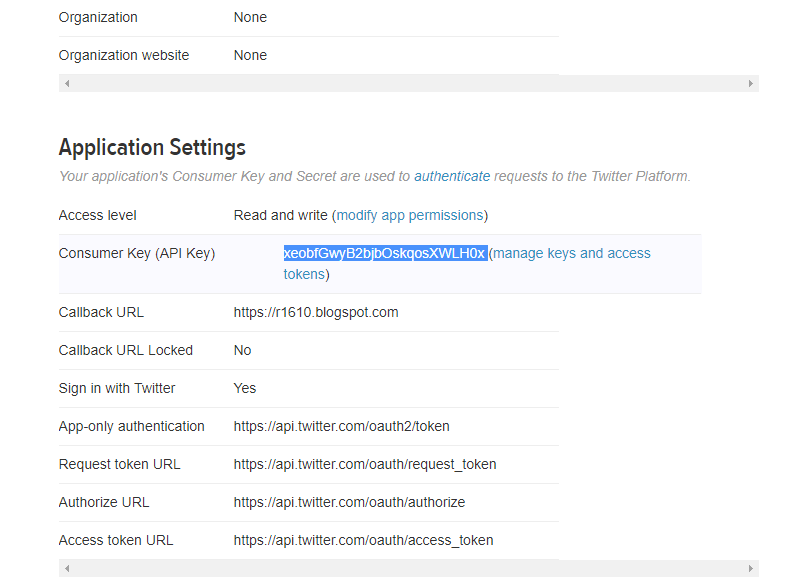
Pprocessing half million =records related to Traffic flights, , in onw simgle contatenated command

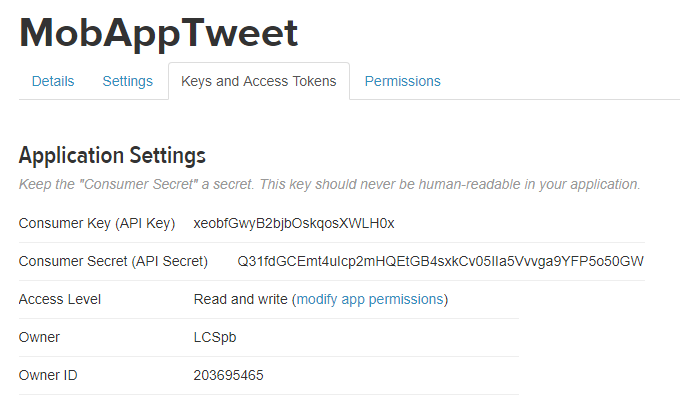
\_\_\_\_\_\_\_\_\_\_

ENuogt time: DEMO processing DAta out of tweeter

Hotest topics, stactistic related to a subject

We are Digital Migrants, Kids are Digitally Native





## Application Settings

*Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.*

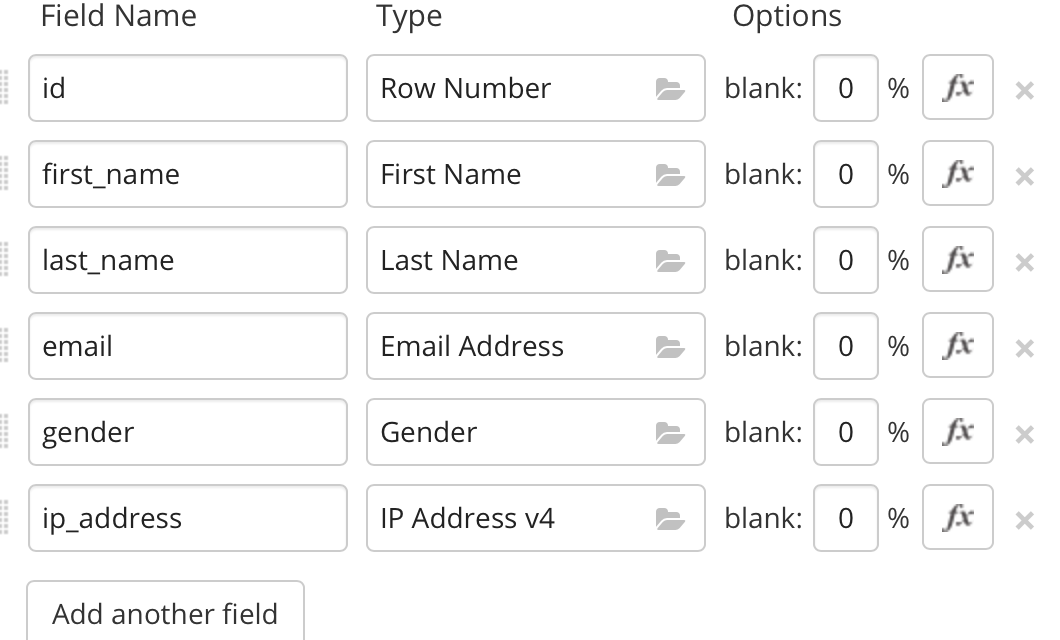
|  |  |
| --- | --- |
| **Consumer Key (API Key)** | xeobfGwyB2bjbOskqosXWLH0x |
| **Consumer Secret (API Secret)** | Q31fdGCEmt4uIcp2mHQEtGB4sxkCv05IIa5Vvvga9YFP5o50GW |
| **Access Token** | 203695465-4WpOzbmQueythd8hKrd4wiorUelEXQyG5DxfsGfz |
| **Access Token Secret** | p5souw4Od1OqAtYgdxrelV8BUzGcHcr7EtON1R1yBIfYk |
| Access LevelRead and write ([modify app permissions](https://apps.twitter.com/app/14751592/permissions)) |  |
| Owner LCSpb | Owner ID 203695465 |

<https://medium.com/@GalarnykMichael/accessing-data-from-twitter-api-using-r-part1-b387a1c7d3e>

Auto Generate DATA for test,

https://www.mockaroo.com

http://databasetestdata.com



Need some mock data to test your app? Mockaroo lets you generate up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| R progrmming:  # Poker and roulette winnings from Monday to Friday:  poker\_vector <- c(140, -50, 20, -120, 240)  roulette\_vector <- c(-24, -50, 100, -350, 10)  days\_vector <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")  names(poker\_vector) <- days\_vector  names(roulette\_vector) <- days\_vector  # Which days did you make money on poker?  selection\_vector <- poker\_vector[poker\_vector>0]  # OR selection\_vector <- poker\_vector>0    # Print out selection\_vector  selection\_vector | Videos |
|  | https://youtu.be/JoArGkOpeU0 |
|  |  |
|  |  |

<http://www.lemonde.fr/connaitre-et-comprendre-les-metiers-de-demain/article/2018/01/19/la-big-data-intelligence-prochaine-etape-pour-les-datamasters_5244160_5240459.html>

**Activities**

**Activity 1:**

|  |  |
| --- | --- |
| **Script** | **Keywords:** |
| cols <- c('price', 'bore', 'stroke', 'horspower','peak.rmp')  # convert the character "?" to an NA  dat[, cols] = lapply(dat[, cols], function(x) ifelse(x=='?', NA, x))  # remove rows with NAS  dat = dat[complete.cases(dat), ]  ## convert character columns to numeric  dat[, cols] = lapply(dat[, cols], as.numeric) | *Lapply*  *complete.cases* |
| lapply returns a list of the same length as X, each element of which is the result of applying FUN to the corresponding element of X. | |
| > matrix(1:9,ncol=3)  [,1] [,2] [,3]  [1,] 1 4 7  [2,] 2 5 8  [3,] 3 6 9 | *array(1:18, dim=c(3,3,2))*  *, , 1*  *[,1] [,2] [,3]*  *[1,] 1 4 7*  *[2,] 2 5 8*  *[3,] 3 6 9*  *, , 2*  *[,1] [,2] [,3]*  *[1,] 10 13 16*  *[2,] 11 14 17*  *[3,] 12 15 18* |
| Add a new colum to the data frame then rearrange the columns such that the last column “Day” appears first  Compute the mean for all rows excluding the first columns values  **Same thing as apply(Duckweed.df[,-1], 1, mean)**  **Ingeneral we can exclude columns by :  apply(Duckweed,df[, -c(1,3,5)], 1, mean )** | |
| # Creating a Graph  attach(mtcars)  plot(wt, mpg)  abline(lm(mpg~wt))  title("Regression of MPG on Weight") | |
| mydata <- read.xlsx("c:/myexcel.xlsx", sheetName = "mysheet") | |
| age <- c(25, 30, 56)  gender <- c("male", "female", "male")  weight <- c(160, 110, 220)  mydata <- data.frame(age,gender,weight) | |
| is.na(x) # returns TRUE of x is missing  y <- c(1,2,3,NA)  is.na(y) # returns a vector (F F F T) | |
| x <- c(1,2,NA,3)  mean(x) # returns NA  mean(x, na.rm=TRUE) # returns 2 | |
|  | |

Data Set repository

<https://vincentarelbundock.github.io/Rdatasets/datasets.html>

<https://docs.microsoft.com/en-us/azure/machine-learning/studio/use-sample-datasets>

<https://www.rdocumentation.org/packages/ggplot2/versions/2.2.1/topics/aes>

https://youtu.be/5E1miS8JrvQ