Introduction to R

Copyright 2014 RESTRICTED CIRCULATION

Fundamentals of R

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES

What is R?

- R is open source and comes with ever growing functionalities based on modules being added to its libraries continuously.
- There have been few attempts at developing point and click R platforms [rattle] but programming platforms [R studio] are much more complete, flexible and reliable.
- We'll focus on Rstudio

Copyright 2014 RESTRICTED CIRCULATION

Object Assignment

- R Object Creation
- Object Naming
- Good Coding Practices

Copyright 2014 RESTRICTED CIRCULATION

“Basic” Data types & Creation

- Numeric
- Character
- Logical

Copyright 2014 RESTRICTED CIRCULATION

Sneak Peak to R Functions

- Few Functions: class , as.* , is.*
- Help/Documentation
- Searching with approximate Names

Copyright 2014 RESTRICTED CIRCULATION

Basic Data Operations: Numeric

- Algebraic Operations
- Operations with functions

Copyright 2014 RESTRICTED CIRCULATION

Basic Data Operation : Strings

- All string operations are achieved with functions

- Concatenating : paste
- Substitution : sub, gsub
- Part Extraction : substr

Copyright 2014 RESTRICTED CIRCULATION

Logical Operations

- Single condition with operators:
  - ==, != , >, < , <=, >= etc
- Combining Multiple Conditions:
  - & and |

Copyright 2014 RESTRICTED CIRCULATION

Collections of “basic” data types

- Vectors
- List
- Data frame

Copyright 2014 RESTRICTED CIRCULATION

Vectors Creation and Access

- Vectors:
  - Collection of same “basic” data types
  - Created with function c()
- Vector Access:
  - Extracting Individual Element
  - Extracting multiple elements
  - Conditional access/extraction

Copyright 2014 RESTRICTED CIRCULATION

Other Handy Ways to Create Vectors

- a:b
- seq
- rep

Copyright 2014 RESTRICTED CIRCULATION

Vector Operations

- Combine
  - The same way you create a vector, you can combine them.
- Operations on vectors
  - Operations happen between corresponding members of vectors
  - When two vectors are of unequal length values of the shorter one are recycled
  - This is true for vectors being input to functions as well. Output is a vector with that function being applied on all elements

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES

Some Utility Functions and Operators

- match
- %in%
- which
- %%
- sort
- rev
- sample
- unique

Copyright 2014 RESTRICTED CIRCULATION

Lists

- Lists are generic vectors
  - They can contain multiple objects not necessarily of the same type
  - It can contain higher level data types as well. For example you can create a list of vectors
- Members of a list can be accessed with their indices as similar to vectors, although syntax is slightly different

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES

Data frames

- These are the one higher data types which we'll be using most of the times
- Data types and functionalities associated with them which we have studied till now will be used to manipulate these data frames and variables/columns/rows inside them
- R has many inbuilt datasets.

Copyright 2014 RESTRICTED CIRCULATION

Exploring data frames

- View
- names, rowans
- head
- str
- dim
- nrow, ncol

Copyright 2014 RESTRICTED CIRCULATION

Accessing values in a data frame

- General format of access is
  - df[row index vector, column index vector]
- One column
  - df[, 4]
- One row
  - df [3,]
- Multiple columns and rows
  - df [c(1,3,4) , c(2,4,6)]
- Negative sign here for indices means exclusion of that/those particular index/indices
- Instead of column and row numbers, you can use their names as well.

Copyright 2014 RESTRICTED CIRCULATION

Conditional Access to a data frame

- Target is to get gas mileage [mpg] data for vehicles which have automatic transmission [am=0]
- We'll start with creating a logical vector for automatic transmission
  - L=mtcars$am==0
- mtcars[L,]
  - This will give me data for vehicles which have automatic transmission
- mtcars[L,]$mpg
  - This will me data for gas mileage of those vehicles

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES

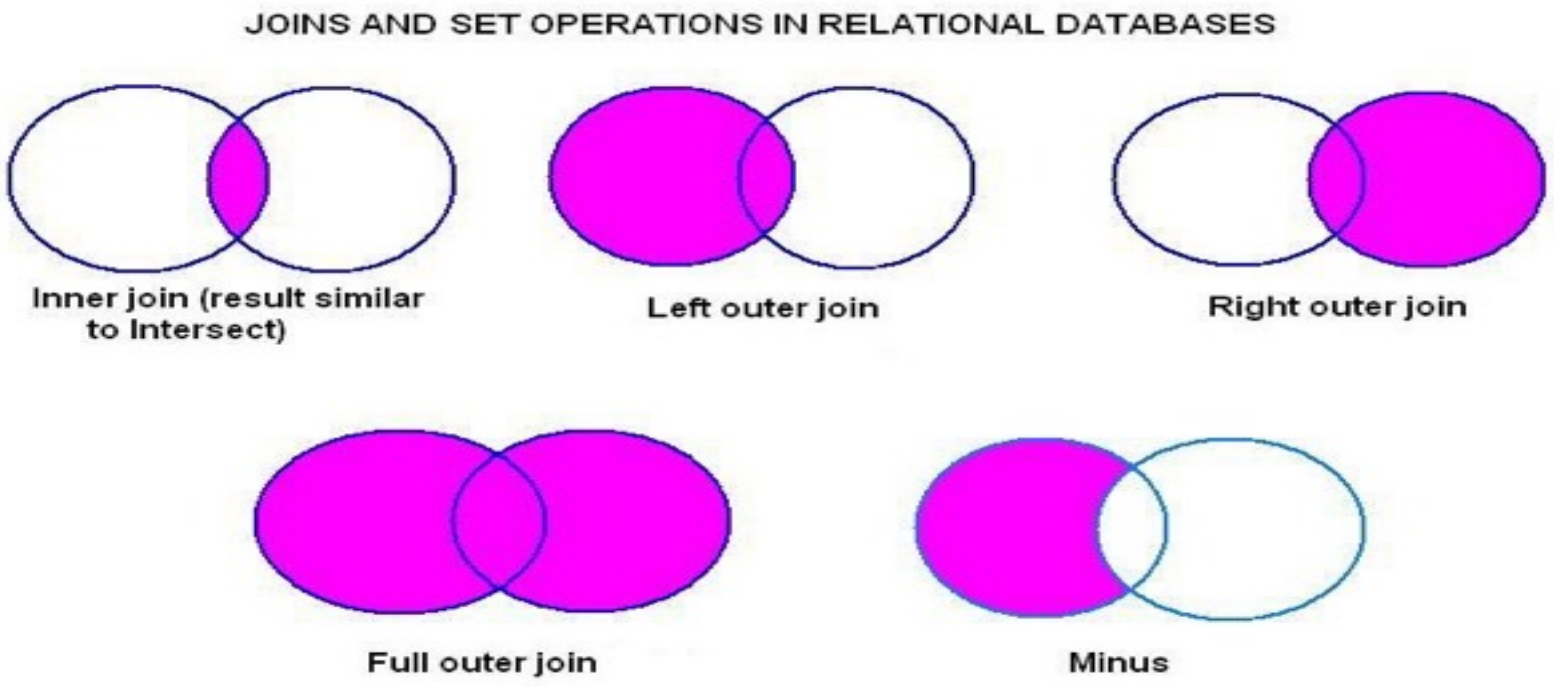
Sorting data sets

- You can simply use function order() with columns which you want to sort. You can use multiple keys/columns to sort
- newdata=mtcars[order(am),]
  - This would return the dataset sorted by variable "am" in ascending order
  - Mtcars[order(-am),] : this would result in descending order sort by variable "am"
- Newdata=mtcars[order(am,-mpg),]
  - What kind of sorting would this result in?

Copyright 2014 RESTRICTED CIRCULATION

Merging two datasets

- Merging two dataset is combining two datasets based on one or more common column. We'll consider one common column first.
- Lets call this common column values id. Following picture depicts what kind of combinations/joins/merges are possible



Copyright 2014 RESTRICTED CIRCULATION

Merging Dataset

- Package Used : dplyr

- types of join :
  - inner
  - full
  - left
  - right
  - anti
  - semi

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES

Iterative Operations in R

- For loops
- Does a parametrised task repetitively for a predefined vector of indices
- Indices dont have to be contiguous numbers
- For loops in R should be avoided with vectorisation for efficiency

Copyright 2014 RESTRICTED CIRCULATION

User Defined Functions

- Three components of a function:
  - Name : should not be same as other pre existing functions
  - Input : Input should be well defined and function should contain a input sanity check
  - Output: Result of function application which is given as output
- None of these components are essential to write a function [although they can not be all missing at once]

Copyright 2014 RESTRICTED CIRCULATION

EDVANCER EDUVENTURES