ACADGILD

SESSION 4:
FOUNDATIONAL R PROGRAMMING-II

Assignment 1

Submitted by: Munmun Ghosal
Login Id: munmun55@gmail.com
(M):+91-8007178659

Table of Contents

1. Problem Statement

1. df1 = data.frame(CustId = c(1:6), Product = c(rep("TV", 3), rep("Radio", 3)))
   df2 = data.frame(CustId = c(2, 4, 6), State = c(rep("Texas", 2), rep("NYC", 1)))
   df1 #left table
   df2 #right table

   For the above given data frames and tables perform the following operations:
   - Return only the rows in which the left table have match.
   - Returns all rows from both tables, join records from the left which have matching keys in the right table.
   - Return all rows from the left table, and any rows with matching keys from the right table.
   - Return all rows from the right table, and any rows with matching keys from the left table.

2. Perform the below operations on above given data frames and tables:
   - Return a long format of the datasets without matching key.
   - Keep only observations in df1 that match in df2.
   - Drop all observations in df1 that match in df2.

2. Solution

1.

The R-script for the given problem is as follows:

```
df1 = data.frame(CustId = c(1:6), Product = c(rep("TV", 3), rep("Radio", 3)))
df2 = data.frame(CustId = c(2, 4, 6), State = c(rep("Texas", 2), rep("NYC", 1)))
df1 #left table
df2 #right table

#Return only the rows in which the left table have match
Left_table <- merge(df1, df2 , by = "CustId")
Left_table

#Return all rows from both tables, join records from the left
#which have matching keys in the right table.
total <- merge(df1, df2, all = TRUE)
total

#Return all rows from the left table, and any rows with matching keys
#from the right table.
row_left_table <- merge(df1, df2, by = "CustId",all.x = TRUE)
row_left_table

#Return all rows from the right table, and any rows with matching keys
#from the left table.
row_right_table <- merge(df1, df2, by = "CustId",all.y = TRUE)
row_right_table
```

The output of the R-Script (from Console window) is given as follows:

```
> df1 = data.frame(CustId = c(1:6), Product = c(rep("TV", 3),
rep("Radio", 3)))
>
> df2 = data.frame(CustId = c(2, 4, 6), State = c(rep("Texas", 2),
rep("NYC", 1)))
>
> df1 #left table
  CustId Product
1      1      TV
2      2      TV
3      3      TV
4      4   Radio
5      5   Radio
6      6   Radio
> df2 #right table
  CustId State
1      2 Texas
2      4 Texas
3      6   NYC
>
> #Return only the rows in which the left table have match
>
> Left_table <- merge(df1, df2 , by = "CustId")
> Left_table
  CustId Product State
1      2      TV Texas
2      4   Radio Texas
3      6   Radio   NYC
>
> #Return all rows from both tables, join records from the left
> #which have matching keys in the right table.
> total <- merge(df1, df2, all = TRUE)
> total
  CustId Product State
1      1      TV  <NA>
2      2      TV Texas
3      3      TV  <NA>
4      4   Radio Texas
5      5   Radio  <NA>
6      6   Radio   NYC
>
> #Return all rows from the left table, and any rows with matching keys
> #from the right table.
> row_left_table <- merge(df1, df2, by = "CustId",all.x = TRUE)
> row_left_table
  CustId Product State
1      1      TV  <NA>
2      2      TV Texas
3      3      TV  <NA>
4      4   Radio Texas
5      5   Radio  <NA>
6      6   Radio   NYC
>
> #Return all rows from the right table, and any rows with matching keys
> #from the left table.
> row_right_table <- merge(df1, df2, by = "CustId",all.y = TRUE)
> row_right_table
  CustId Product State
1      2      TV Texas
2      4   Radio Texas
3      6   Radio   NYC
```

2.

The R-script for the given problem is as follows:

```
library("dplyr")
df1 = data.frame(CustId = c(1:6), Product = c(rep("TV", 3), rep("Radio", 3)))
df2 = data.frame(CustId = c(2, 4, 6), State = c(rep("Texas", 2), rep("NYC", 1)))
df1 #left table
df2 #right table

# Return a long format of the datasets without matching key.
dfj<-merge(x=df1,y=df2,by="CustId",all=FALSE)
dfj

#or
dfx<-merge(df1, df2, by="CustId", all=TRUE)
dfx$CustId <- NULL
dfx

# Keep only observations in df1 that match in df2.
semi_join(df1, df2,by="CustId")

# Drop  all observations in df1 that match in df2.
anti_join(df1,df2,by="CustId")
```

The output of the R-Script (from Console window) is given as follows:

```
> library("dplyr")
> df1 = data.frame(CustId = c(1:6), Product = c(rep("TV", 3), rep("Radio", 3)))
>
> df2 = data.frame(CustId = c(2, 4, 6), State = c(rep("Texas", 2), rep("NYC", 1)))
>
> df1 #left table
  CustId Product
1      1      TV
2      2      TV
3      3      TV
4      4   Radio
5      5   Radio
6      6   Radio
> df2 #right table
  CustId State
1      2 Texas
2      4 Texas
3      6   NYC
>
> # Return a long format of the datasets without matching key.
>
> dfj<-merge(x=df1,y=df2,by="CustId",all=FALSE)
> dfj
  CustId Product State
1      2      TV Texas
2      4   Radio Texas
3      6   Radio   NYC
>
```

```
> #or
>
> dfx<-merge(df1, df2, by="CustId", all=TRUE)
> dfx$CustId <- NULL
> dfx
  Product State
1      TV  <NA>
2      TV Texas
3      TV  <NA>
4   Radio Texas
5   Radio  <NA>
6   Radio   NYC
>
> # Keep only observations in df1 that match in df2.
> semi_join(df1, df2,by="CustId")
  CustId Product
1      2      TV
2      4   Radio
3      6   Radio
>
> # Drop  all observations in df1 that match in df2.
> anti_join(df1,df2,by="CustId")
  CustId Product
1      1      TV
2      3      TV
3      5   Radio
```