

LAST MODIFED: 061619

AXTRIA INSTITUTE

R102 TRAINING ASSIGNMENT

Copyright © 2019 Axtria Inc. All Rights Reserved.

The information contained herein is subject to change without notice.

# NOTES

* Please place your output for Questions 1 and 2 in different files (CSV/Excel/Text) and submit both your output files and R scripts together.
* Read each sub-question completely before working on it.
* Ideally no for-loops should be used for this assignment. Please take advantage of R's vector-based functions.

# subsetting dataframes

**HINT:**

When subsetting a dataframe you always subset in the form DataFrame[row , column] or DataFrame[row , ] or DataFrame[ , column]

**QUESTIONS:**

* 1. Import training\_data1 and name the dataframe “ims\_trx”.

setwd("D:/A5286")

getwd()

ims\_trx=read.csv("training\_data1\_2207.csv")

View(ims\_trx)

* 1. Create a new column called “TRX\_Round” that is the TRX\_Qty value rounded to the nearest number (integer).

TRX\_Round=round(ims\_trx$TRX\_Qty,0)

ims\_trx=cbind(ims\_trx,TRX\_Round)

View(ims\_trx)

* 1. Create another new column called “target\_flag” and from the trgt\_flg column change “Y” into Yes and “N” into No.

str(ims\_trx)

ims\_trx$trgt\_flg=as.character(ims\_trx$trgt\_flg)

target\_flag=replace(ims\_trx$trgt\_flg,ims\_trx$trgt\_flg==c("N","Y"),c("NO","YES"))

View(target\_flag)

ims\_trx=cbind(ims\_trx,target\_flag)

View(ims\_trx)

* 1. Save the strings “ims\_id”, “TRX\_Round”, and “target\_flag” into a vector named “subsetCols”. Use the subsetCols vector to select these columns from the ims\_trx dataframe.

colnames(ims\_trx)

subsetCols=ims\_trx[,c("ims\_id","TRX\_Round", "target\_flag")]

View(subsetCols)

* 1. Find all values in the TRX\_Round column greater than 100 and save this to a vector named “subsetRows”. Use the subsetRows vector to select these rows from the ims\_trx dataframe.

subsetRows= ims\_trx[ims\_trx$TRX\_Round>100,"TRX\_Round"]

View(subsetRows)

* 1. Use subsetCols and subsetRows to create a final subset of ims\_trx and save it as “trx\_over\_100”.

trx\_over\_100=ims\_trx[ims\_trx$TRX\_Round>100,c( "ims\_id","TRX\_Round", "target\_flag")]

View(trx\_over\_100)

# JOINING AND SUMMATIONS

**QUESTIONS:**

* 1. Import the three datasets. Name training\_data1 as “monthlyTRX”, training\_data2 as “territory”, and training\_data3 as “specialtyGroup”. In territory and specialtyGroup replace spaces or periods in column titles with underscores.

monthlyTRX=read.csv("training\_data1\_2207.csv")

View(monthlyTRX)

territory=read.csv("training\_data2\_2207.csv")

View(territory)

colnames(territory)[2]="Territory\_ID"

View(territory)

specialtyGroup=read.csv("training\_data3\_2207.csv")

colnames(specailtyGroup)=c("IMS\_Specialty","Specialty\_Group")

View(specailtyGroup)

* 1. Left join territory with specialtyGroup on the Spec and IMS Specialty columns and name this dataframe “terr2”. Keep only the ims\_id, Territory\_ID, and Specialty\_Group columns. For ims\_id's that do not have a specialtyGroup i.e. they are “NA”, set the Specialty\_Group to “OTHER”.

terr2=merge(territory,specialtyGroup,by.x = "Spec", by.y = "IMS\_Specialty",all.x=T)

View(terr2)

colnames(terr2)

terr2=terr2[-1]

View(terr2)

terr2[is.na(terr2$Specialty\_Group),]="OTHER"

View(terr2)

* 1. Using the terr2 dataframe create a vector or dataframe named “SpecialtyCount” that is the count of ims\_ids in each specialtyGroup.

SpecialtyCount=aggregate(cbind(Count=ims\_id)~Specialty\_Group,data = terr2,FUN = function(x){NROW(x)})

View(SpecialtyCount)

* 1. Now inner join the terr2 dataframe with the monthlyTRX dataframe on their ims\_ids and save it into a dataframe called “ims\_data”.

ims\_data=merge(terr2,monthlyTRX,by.x = "ims\_id", by.y = "ims\_id")

View(ims\_data)

* 1. Using the ims\_data dataframe sum the TRX\_qty column by ims\_id and save it into a dataframe called trx\_sum\_2012. Order this dataframe from greatest to least.

trx\_sum\_2012=aggregate(TRX\_Qty~ims\_id, data = ims\_data, FUN = sum)

View(trx\_sum\_2012)

trx\_sum\_2012=trx\_sum\_2012[order(trx\_sum\_2012$TRX\_Qty,decreasing = TRUE),]

View(trx\_sum\_2012)

* 1. Using the ims\_data dataframe remove all rows where trgt\_flg is N from the dataset and save as targeted\_physicians

targeted\_physicians= ims\_data[ims\_data$trgt\_flg=="Y",]

View(targeted\_physicians)

* 1. Create a function named topTerritory that takes the targeted\_physicians dataframe and a month as input and outputs the Territory ID with the highest TRX\_Qty in the month.