**Homework 2**

**STAT 6494 Fall 2017**

**Haim Bar**

**Due Date: September 17, 2017**

**(by 23:59, via HuskyCT)**

1. **SAS**: you want to create a permanent data set that uses a DATALINES statement to read in values for X and Y. In the DATA step, you want to create a new variable, Z, equal to 100 + 50 X + 2X^2 – 25Y - 3Y^2. Use the following (X, Y) data pairs: (1, 2), (3, 6), (5, 9), (-2, -3) and (9, 11). View this data set using PROC CONTENTS and PROC PRINT

libname hw2 'P:\STAT-6494-Data Management in SAS and R\data';

**Data** hw2.p1;

Input x y;

z=**100**+**50**\*x+**2**\*x\*\***2**-**25**\*y-**3**\*y\*\***2**;

Datalines;

1 2

3 6

5 9

-2 -3

9 11

;

**Run**;

\*The Descriptor Portion of Data Set p1;

**Proc** **contents** data=hw2.p1;

**Run**;

\*The Data Portion of Data Set p1;

**Proc** **print** data=hw2.p1;

**Run**;



1. **R**: Repeat question 1 in R.
2. Create a matrix to save (X, Y) values, calculate Z, and save the matrix which contains the columns X,Y,Z into an Rdata file.
3. Create a data frame to save (X, Y) values, calculate Z, save the data frame which contains the columns X,Y,Z into an Rdata file.
4. Create one list to save (X, Y) values, and calculate Z, and save the workspace.
5. Load the Rdata you saved in part b, and view this data frame using str().

# 2a

data=c(1, 2,3, 6,5, 9,-2, -3,9, 11)  
cnames=c("X","Y")  
rnames=c(1:5)  
m=matrix(data,nrow=5,ncol=2,byrow=T,dimnames=list(rnames,cnames))  
Z=100+50\*m[,1]+2\*m[,1]\*\*2-25\*m[,2]-3\*m[,2]\*\*2  
m=cbind(m,Z)  
m

## X Y Z  
## 1 1 2 90  
## 2 3 6 10  
## 3 5 9 -68  
## 4 -2 -3 56  
## 5 9 11 74

str(m)

## num [1:5, 1:3] 1 3 5 -2 9 2 6 9 -3 11 ...  
## - attr(\*, "dimnames")=List of 2  
## ..$ : chr [1:5] "1" "2" "3" "4" ...  
## ..$ : chr [1:3] "X" "Y" "Z"

save(m, file="P:/STAT-6494-Data Management in SAS and R/xyzmatrix.RData")

# 2b

X=c(1, 3,5,-2,9)  
Y=c(2,6,9,-3,11)  
Z=100+50\*X+2\*X\*\*2-25\*Y-3\*Y\*\*2  
p1<-data.frame(X,Y,Z)  
p1

## X Y Z  
## 1 1 2 90  
## 2 3 6 10  
## 3 5 9 -68  
## 4 -2 -3 56  
## 5 9 11 74

str(p1)

## 'data.frame': 5 obs. of 3 variables:  
## $ X: num 1 3 5 -2 9  
## $ Y: num 2 6 9 -3 11  
## $ Z: num 90 10 -68 56 74

save(p1, file="P:/STAT-6494-Data Management in SAS and R/xyzdataframe.RData")

# 2C

X=c(1, 3,5,-2,9)  
Y=c(2,6,9,-3,11)  
L=list(X=X,Y=Y)  
Z=100+50\*X+2\*X\*\*2-25\*Y-3\*Y\*\*2  
L=list(X=X,Y=Y,Z=Z)  
L

## $X  
## [1] 1 3 5 -2 9  
##   
## $Y  
## [1] 2 6 9 -3 11  
##   
## $Z  
## [1] 90 10 -68 56 74

str(L)

## List of 3  
## $ X: num [1:5] 1 3 5 -2 9  
## $ Y: num [1:5] 2 6 9 -3 11  
## $ Z: num [1:5] 90 10 -68 56 74

save.image(file="P:/STAT-6494-Data Management in SAS and R/xyzworkspace.RData")

# 2d

load("P:/STAT-6494-Data Management in SAS and R/xyzdataframe.RData")  
str(p1)

## 'data.frame': 5 obs. of 3 variables:  
## $ X: num 1 3 5 -2 9  
## $ Y: num 2 6 9 -3 11  
## $ Z: num 90 10 -68 56 74

1. **SAS**: You have a text file called geocaching.txt with data values arranged as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Description** | **Starting Column** | **Ending Column** | **Data Type** |
| Name | Cache name | 1 | 20 | Char |
| LongDeg | Longitude degrees | 21 | 22 | Num |
| LongMin | Longitude minutes | 23 | 28 | Num |
| LatDeg | Latitude degrees | 29 | 30 | Num |
| LatMin | Latitude minutes | 31 | 36 | Num |
|  |  |  |  |  |

Here’s a listing of the file:

Higgensville Hike 4030.2937446.539

Really Roaring 4027.4047442.147

Cushetunk Climb 4037.0247448.014

Uplands Trek 4030.9907452.794

Create a temporary SAS data set called Cache using this data file. Use column input to read the data values.

**data** Cache;

infile 'P:\STAT-6494-Data Management in SAS and R\data\geocaching.txt';

input name $ **1**-**20**

longdeg **21**-**22**

longmin **23**-**28**

latdeg **29**-**30**

latmin **31**-**36**;

label name='Cache name'

longdeg='Longitude degrees'

longmin='Longitude minutes'

latdeg='Latitude degrees'

latmin='Latitude minutes';

**run**;

1. **R**: Read the geocaching.txt file into a data frame.

Cache<-read.table("P:/STAT-6494-Data Management in SAS and R/data/geocaching.txt ", widths=c(20,2,6,2,6), header=F)

colnames(Cache)<-c("Name","LongDeg","LongMin","LatDeg","LatMin")

1. **SAS**: You are given a text file called stockprices.txt containing information on the purchase and sale of stocks. The data layout is as follows.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Description** | **Starting Column** | **Length** | **Data Type** |
| Stock | Stock symbol | 1 | 4 | Char |
| PurDate | Purchase date | 5 | 10 | mm/dd/yyyy |
| PurPrice | Purchase price | 15 | 6 | Dollar signs and commas |
| Number | Number of shares | 21 | 4 | Num |
| SellDate | Selling date | 25 | 10 | mm/dd/yyyy |
| SellPrice | Selling price | 35 | 6 | Dollar signs and commas |

A listing of the data file is:

IBM 5/21/2006 $80.0 10007/20/2006 $88.5

CSCO04/05/2005 $17.5 20009/21/2005 $23.6

MOT 03/01/2004 $14.7 50010/10/2006 $19.9

XMSR04/15/2006 $28.4 20004/15/2006 $12.7

BBY 02/15/2005 $45.2 10009/09/2006 $56.8

1. Create a SAS data set (call it Stocks) by reading the data from the file. Use a FILENAME statement to create a fileref instead of using the file name on the INFILE statements. Use formatted input.

filename fileref 'P:\STAT-6494-Data Management in SAS and R\data\stockprices.txt';

**data** Stocks;

infile fileref;

input @**1** Stock $4.

@**5** PurDate mmddyy10.

@**15** PurPrice dollar6.

@**21** Number **4.**

@**25** SellDate mmddyy10.

@**35** SellPrice dollar6.1;

format PurDate mmddyy10. PurPrice dollar6.1 SellDate mmddyy10. SellPrice dollar6.1;

**run**;

1. Repeat part (a) with List Input using INFORMAT.

filename fileref 'P:\STAT-6494-Data Management in SAS and R\data\stockprices.txt';

**data** Stocks;

informat Stock $4.

PurDate mmddyy10.

PurPrice dollar6.

Number **4.**

SellDate mmddyy10.

SellPrice dollar6.;

infile fileref;

input Stock @**5** PurDate @**15** PurPrice @**21** Number **4.** SellDate SellPrice;

format PurDate mmddyy10. PurPrice dollar6.1 SellDate mmddyy10. SellPrice dollar6.1;

**run**;

1. Compute several new variables as follows:

|  |  |  |
| --- | --- | --- |
| **Variable** | **Description** | **Computation** |
| TotalPur | Total purchase price | Number times PurPrice |
| TotalSell | Total selling price | Number times Sell Price |
| Profit | Profit | TotalSell minus TotalPur |
|  |  |  |

**data** Stocks;

set stocks;

TotalPur= Number\*PurPrice;

TotalSell= Number\*SellPrice;

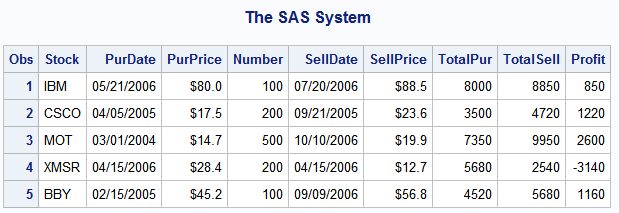
Profit= TotalSell-TotalPur;

**run**;

1. Print out the contents of this data set using PROC PRINT.

**proc** **print** data=Stocks;

**run**;



1. **R**: You are given a text file called stockprices\_r.txt containing information on the purchase and sale of stocks. The data layout is as follows.

|  |  |
| --- | --- |
| **Variable** | **Description** |
| Stock | Stock symbol |
| PurDate | Purchase date |
| PurPrice | Purchase price |
| Number | Number of shares |
| SellDate | Selling date |
| SellPrice | Selling price |

A listing of the data file is:

IBM 5/21/2006 8000.0 100 07/20/2006 88.5

CSCO 04/05/2005 1700.5 200 09/21/2005 23.6

MOT 03/01/2004 1400.7 500 10/10/2006 19.9

XMSR 04/15/2006 2800.4 200 04/15/2006 12.7

BBY 02/15/2005 4500.2 100 09/09/2006 56.8

1. Create a data frame by reading the data from the file. Choose your own column names.

stocks<-read.csv("P:/STAT-6494-Data Management in SAS and R/data/stockprices\_r.txt", sep="", header=F)

colnames(stocks)<-c("Stock","PurDate","PurPrice","Number","SellDate","SellPrice")

1. Repeat question 5c, calculate all 3 variables and save them into a list.

attach(stocks)

TotalPur= Number\*PurPrice

TotalSell= Number\*SellPrice

Profit= TotalSell-TotalPur

L=list(TotalPur=TotalPur,TotalSell=TotalSell,Profit=Profit)

1. Print out a sentence like the following: (use the paste function)

IBM is bought at $8,000 on 2006-05-21.

paste(stocks[1,1],"is bought at",paste("$", prettyNum(stocks[1,3],big.mark=","),sep=""), "on", paste(as.Date(stocks[1,2],"%m/%d/%Y"),".",sep=""))

1. **R**: You are given a text file called bankdata\_r.txt. The data layout is as follows.

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Starting Column** | **Length** | **Data Type** |
| Name | 1 | 14 | Char |
| Gender | 15 | 1 | Factor |
| ID | 16 | 5 | Char |
| Balance | 21 | 8 | Num |
| Credit History | 29 | 4 | Ordinal |

Here’s a listing of file:

Philip Jones MV123414322.32GOOD

Nathan PhilipsFV139915202.45GOOD

Shu Lu FW889251233.45GOOD

Betty Boop MV767750002.78MED

1. Create a data frame by reading the data from the file. Skip the first person, choose your own column names. Save the data as the type listed in the above table.

bank<-read.fwf("P:/STAT-6494-Data Management in SAS and R/data/bankdata\_r.txt", widths=c(14,1,5,8,4), n=3, header=F, skip=1)

colnames(bank)<-c("Name","Gender","ID","Balance","Credit\_History")

bank$Name<-as.character(bank$Name)

bank$Gender<-factor(bank$Gender)

bank$ID=as.character(bank$ID)

bank$Credit\_History<-factor(Credit\_History,levels=c("MED","GOOD"))

1. View this data set using str().

str(bank)

'data.frame': 3 obs. of 5 variables:

$ Name : chr "Nathan Philips" "Shu Lu " "Betty Boop "

$ Gender : Factor w/ 2 levels "F","M": 1 1 2

$ ID : chr "V1399" "W8892" "V7677"

$ Balance : num 15202 51233 50003

$ Credit\_History: Ord.factor w/ 2 levels "MED"<"GOOD": NA NA NA