**Dylan Wong Xuan Kai** **H00280633**

# F79PA Portfolio Theory and Asset Models Assignment 1

1) Question requires running the provided function to obtain the names of fruits and vegetables.

Line 9 is a list of all required packages for the code below. Line 12-29 installs and loads R packages that have yet to be acquired. Line 32-70 reads the StudentBudgetFn function provided from question which returns 5 random items based on input numbers and line 72 calls the function with function input as my student number (00280633). The items obtained: 'Durian', 'Raspberries', 'Apricot', 'Pears', 'Cellery'. Line 73 is a byte code compiler function which enables reused function above in later questions to be executed more quickly.

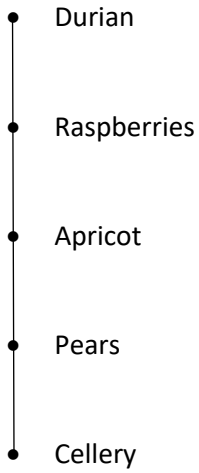
2) Question requires writing a function (named 'MyPreferences') for a matrix which records inputs consisting of 1's and 0's which corresponds to the weak preference relationship of items from Q1 through command line prompt.

Line 80 to 107 reads the 'MyPreferences' function that prompts the user to respond with 0 or 1 based on item preferences to complete the matrix of item preferences where matrix is then stored in a .txt file. Line 111 calls the function with function input as my student number, name of .txt file name of preference matrix, and directory to open or save .txt file.

Preference matrix:

| | Durian | Raspberries | Apricot | Pears | Cellery |
|---|---|---|---|---|---|
| Durian | "1" | "1" | "1" | "1" | "1" |
| Raspberries | "0" | "1" | "1" | "1" | "1" |
| Apricot | "0" | "0" | "1" | "1" | "1" |
| Pears | "0" | "0" | "0" | "1" | "1" |
| Cellery | "0" | "0" | "0" | "0" | "1" |

Plot of weak preference tree diagram:



3) Question requires stating what properties arise from initialization of ‘Preferences’ matrix as diagonal identity matrix produces in my binary relation.

Diagonal identity matrix corresponds to equivalence class and therefore reflexive property, transitive property and symmetry property have been exhibited.

4) Question requires stating properties required for a binary relation to be a weak preference order and determining if ‘Preference’ matrix obtained from Q2 satisfies such properties.

Complete property and (2) transitive property are required for a binary relation of weak preference. Yes ‘Preference’ matrix satisfies both properties since:
(1) Durian is weakly preferred to Apricot but Apricot is not weakly preferred to Durian.
(2)Durian is weakly preferred to Apricot and Apricot is weakly preferred to Pears shows that Durian is also weakly preferred to Pears too.

5) Question requires defining upper and lower contour sets of each item from Q2 ‘Preference’ matrix.

<u>Upper Contour Sets:</u>

Durian : Durian

Raspberries: Durian, Raspberries

Apricot : Durian, Raspberries, Apricot

Pears: Durian, Raspberries, Apricot, Pears

Cellery : Durian, Raspberries, Apricot, Pears, Cellery

<u>Lower Contour Sets:</u>

Durian : Durian, Raspberries, Apricot, Pears, Cellery

Raspberries: Raspberries, Apricot, Pears, Cellery

Apricot : Apricot, Pears, Cellery

Pears: Pears, Cellery

Cellery : Cellery

6) Question requires writing a R code function named ‘ContourSets’ to verify upper and lower contour sets of each item from Q5.

Line 119-165 reads the ’ContourSets’ function which stores the upper contour list and lower contour list in two separate .txt files in device directory based on the values in ‘Preference’ matrix. If ‘1’ then add to lower upper contour list and if ‘0’ then add to lower contour list based on xRy binary relation with x=i (row item), y=j (column item), and x >= y being x

weakly preferred to y and y>=x being y weakly preferred to x, also implying x <= y. Line 169 calls the function with function input as my student number, name of .txt file, and directory to open or save .txt file.

Lower Contour Set of items (x >= y):

```
               V1       V2            V3        V4       V5
Durian >=      "Durian" "Raspberries" "Apricot" "Pears"  "Cellery"
Raspberries >= ""       "Raspberries" "Apricot" "Pears"  "Cellery"
Apricot >=     ""       ""            "Apricot" "Pears"  "Cellery"
Pears >=       ""       ""            ""        "Pears"  "Cellery"
Cellery >=     ""       ""            ""        ""       "Cellery"
```

Upper Contour Set of items (y>=x, also implying x <= y):

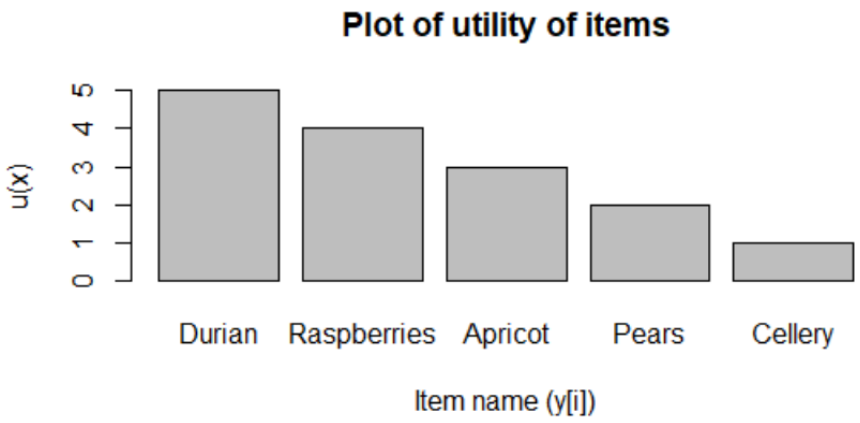
```
               V1       V2            V3        V4       V5
Durian <=      "Durian" ""            ""        ""       ""
Raspberries <= "Durian" "Raspberries" ""        ""       ""
Apricot <=     "Durian" "Raspberries" "Apricot" ""       ""
Pears <=       "Durian" "Raspberries" "Apricot" "Pears"  ""
Cellery <=     "Durian" "Raspberries" "Apricot" "Pears"  "Cellery"
```

7) Question requires creating a function called 'MyInducedUtility' which plots utility function and returns utility values based on lower contour set of items from Q6.

Line 179 to 222 reads the 'MyInducedUtility' function which plots the utility of each item based on the lower contour list of the items, returns the utility of each item as vectors and stores the plot diagram as .png file in the directory. The utility of each item (by row) is basically the total number of items in each row of lower contour list from Q6. Line 226 calls the function with function input as my student number, name of .txt file of both upper and l ower contour list , and directory to open or save .txt file.

Return values and plot of utility of items:

```
> MyInducedUtility(00280633,'MyUpperContourList.txt','MyLowerContourList.
txt',"C:\\Users\\UX330\\Documents\\F79PAProject\\H00280633\\")
[1] 5 4 3 2 1
```




Comment:

Utility comparison between items u(item name):

u(Durian) >= u(Raspberries) >=u( Apricot) >= u(Pears) >= u(Cellery)

(for xRy binary relation being x>=y being u(x)>=u(y), with >= being weakly preferred to)

```
# Portfolio Thoery and Asset model
# H00280633
# Assignment 1 R code
# Note: Plese ensure latest version of Rtools to prevent error dueing package installation

## Question 1 ##
# To prevent code error from not having the required package for all code below
# Name of pacakges required
packages_required<-c("base", "datasets", "graphics", "grDevices", "methods", "stats", "com
piler", "utils")

# Check if packages exist in device, or else install package
for(i in packages_required){
  if(is.element(i,installed.packages()[,1])==TRUE){
    next
  }else{
    install.packages(i)
  }
}

# Basic packages
library(base)
library(datasets)
library(graphics)
library(methods)
library(stats)

library(compiler) # Byte compiler package to run repeated code more speedily (cmpfun())
# Pls do not detach package 'utils', will lead to error warning but code still works with output
library(utils)    # if error remains then close RStudio tab and reopen again, error will be gone

# Function to Create Student Budget Set
StudentBudgetSetFn <- function(StudentNum){

  #Set seed
  set.seed(StudentNum)

  # Consider the following items -- lablelled 1 to 20
  # 1. Apple
  # 2. Apricot
  # 3. Banana
  # 4. Blueberries
  # 5. Blackberries
  # 6. cabbage
  # 7. Carrot
  # 8. Cellery
  # 9. Durian
  # 10. Egg plant
  # 11. Fig
  # 12.Fennel
  # 13.Grape fruit
```

```
 # 14.Grapes
 # 15.Green Beans
 # 16.Honey Dew Mellon
 # 17.Oranges
 # 18.Pears
 # 19.Raspberries
 # 20.Strawberries

 #Array of items
 TotalItems=c('Apple', 'Apricot', 'Banana', 'Blueberries', 'Blackberries', 'cabbage', 'Carrot', 'Cellery',
          'Durian', 'Egg plant', 'Fig', 'Fennel', 'Grape fruit', 'Grapes', 'Green Beans', 'Honey Dew Mellon',
          'Oranges', 'Pears', 'Raspberries', 'Strawberries')


 # Use the command sample and as follows to select an index of items in the list to consider in your preference elicitation survey.
 StudentBudgetSetIndx = sample(1:20, 5, replace=F)
 StudentBudgetSet = TotalItems[StudentBudgetSetIndx]

 return(StudentBudgetSet)
}
# My student number: H00280633
StudentBudgetSetFn(StudentNum=00280633) # To obtain names of fruits and vegetables
cmp_items=cmpfun(StudentBudgetSetFn)    # Byte code compiler of R for function 'StudentBudgetSetFn'

## Question 2 ##
# Function to create matrix of items preference based on manual input of user
# Write student number to extract list of items in sequence
# Write "filename" with extension .txt or .csv
# Write "directory" to open from or save file in desired directory
MyPreferences <- function(StudentNum,filename,directory){
 setwd(directory)  # Ensure working directory is directory of interest
 # Store array of position of items in variable y
 y <- cmp_items(StudentNum)
 # y[1]=Durian, y[2]=Raspberries, y[3]=Apricot, y[4]=Pears, y[5]=Cellery

 x <- diag(5)     # Generating 5*5 diagonal identity matrix **Notation: x[row,column]
 rownames(x)<-y   # Rewrite matrix row names with corresponding item names
 colnames(x)<-y   # Rewrite matrix column names with corresponding item names
 n<-5             # Referring to the total 5 items retrieved from Question 1
 z<-numeric(n)

 # For loop to fill weak prefernce matrix of items based on prompt and reply of user
 for(i in 1:n){
  for(j in 1:n){
   if(i==j){   # If items compared are the same, automatically = '1'
     next
```

```
    }else{     # If items compared are different, prompt with question and fill matrix with user
 respond
      z[j]<-paste('Do you weakly prefer', y[i] ,'vs', y[j], '?', sep=" ")
      print(z[j])
      x[i,j]<-readline(prompt="\n(enter 1 Yes, 0 otherwise)\n")
    }
   }
  }
  # Prints matrix to a .txt file and copy file to desired directory
  write.table(x,file=filename, row.names = TRUE, col.names = TRUE)
  file.copy(filename,directory)
}
# Tweak desired input 1 for student number, input 2 for file name, input 3 for directory
# *** Must use "\\" instead of "\" after copy and paste from directory to prevent Hex digits c
haracter string error
# My Example
MyPreferences(00280633, "MyPreferencesResults.txt", "C:\\Users\\UX330\\Documents\\F7
9PAProject\\H00280633\\")

## Question 6 ##
# Function to produce .txt file of upper and lower contour set of itmes from preference matr
ix
# Write student number to extract list of items in sequence
# Write "filename" to open file where Preference matrix is saved at
# Write "directory" to open from or save file in desired directory
# Use "\\" instead of "\" for directory to prevent Hex digits character string error
ContourSets<-function(StudentNum, filename, directory){
  setwd(directory)   # Ensure working directory is directory of interest
  # Store array of position of items in variable y
  y <- cmp_items(StudentNum)
  # y[1]=Durian, y[2]=Raspberries, y[3]=Apricot, y[4]=Pears, y[5]=Cellery

  n<-5  # Referring to the total 5 items retrieved from Question 1
  # Creating lists with 5 vectors
  UpperContoursList<-as.list(vector("list",5))
  LowerContoursList<-as.list(vector("list",5))

  for(a in 1:n){  # Converting each vector to empty arrays to fill upper and lower contour list
    UpperContoursList[[a]]<-array(data='',dim = c(1,5))
    LowerContoursList[[a]]<-array(data='',dim = c(1,5))
  }
  PreferencesII<-as.matrix(read.table(filename))  # Converting .txt files to matrix

  # For loops to fill upper and lower contour list of each item with compared item based on p
reference matrix values
  for(i in 1:n){
    for(j in 1:n){
      if(i==j){ # if items compared are the same, basically included in both upper and lower con
tour list of items
        LowerContoursList[[i]][j]<-y[j]
        UpperContoursList[[i]][j]<-y[j]
```

```
    }else{
      # If items compared are different, add item names to upper and lower contour list base
d on corresponding values on preference matrix
      # If '1' add to upper contour list of the item[i], if '0' add to lower contour list of the item
[i]
      if(PreferencesII[i,j]==1){
        LowerContoursList[[i]][j]<-y[j]
      }else{
        UpperContoursList[[i]][j]<-y[j]
      }
    }
  }
 }
 # Row names for upper and lower contour list
 items_upper<-c('Durian <=','Raspberries <=','Apricot <=','Pears <=','Cellery <=')
 items_lower<-c('Durian >=','Raspberries >=','Apricot >=','Pears >=','Cellery >=')
 UpperContoursList<-do.call(rbind,UpperContoursList)
 LowerContoursList<-do.call(rbind,LowerContoursList)
 rownames(UpperContoursList)<-items_upper
 rownames(LowerContoursList)<-items_lower
 # Prints upper and lower contour list to a .txt file and copy file to desired directory
 write.table(UpperContoursList,file='MyUpperContourList.txt', row.names = TRUE, col.name
s = TRUE)
 file.copy('MyUpperContourList.txt',directory)
 write.table(LowerContoursList,file='MyLowerContourList.txt', row.names = TRUE, col.name
s = TRUE)
 file.copy('MyLowerContourList.txt',directory)
}
# Tweak desired input 1 for student number, input 2 for file name, input 3 for directory
# *** Must use "\\" instead of "\" after copy and paste from directory to prevent Hex digits c
haracter string error
# My Example
ContourSets(00280633,"MyPreferencesResults.txt", "C:\\Users\\UX330\\Documents\\F79P
AProject\\H00280633\\")



## Question 7 ##
# Function to produce plot for utility of each item based on lower contour list of items
# Write student number to extract list of items in sequence
# Write "filename" to open file where upper and lower contour list is saved at
# Write "directory" to open from or save file in desired directory
# Use "\\" instead of "\" for directory to prevent Hex digits character string error
MyInducedUtility<-function(StudentNum, filename_upper, filename_lower, directory){
 setwd(directory)   # Ensure working directory is directory of interest
 # Store array of position of items in variable y
 y <- cmp_items(StudentNum)
 # y[1]=Durian, y[2]=Raspberries, y[3]=Apricot, y[4]=Pears, y[5]=Cellery

 n<-5  # Referring to the total 5 items retrieved from Question 1
 MyUtility<-numeric(n)
```

```
 # Converting .txt files of upper and lower contour list to matrix
 MyUpperContourListII<-as.matrix(read.table(filename_upper))
 MyLowerContourListII<-as.matrix(read.table(filename_lower))
 # Creating lists with 5 vectors
 upper_contour<-as.list(vector("list",5))
 lower_contour<-as.list(vector("list",5))

 # Alternate method of converting each vector to empty arrays to fill upper and lower conto
ur list
 for(i in 1:n){
  upper_contour[[i]]<-MyUpperContourListII[i,1:5]
  lower_contour[[i]]<-MyLowerContourListII[i,1:5]
 }

 MyUpperContourListII<-upper_contour
 MyLowerContourListII<-lower_contour

 # For loop to compute number of items for each corresponding ites of upper and lower con
tour list
 for(i in 1:5){
  for(j in 1:5){
   # For each item row[i] of lower contour list calculate number of items in row
   if(MyLowerContourListII[[i]][j]!=''){
    MyUtility[i]<-MyUtility[i]+1
   }else{
    next
   }
  }
 }
 # Total items in each item row[i] of lower contour list are then divivded by total items (n<-
5) to reflecet on item utility
 # Plot the utilty of item, return the vector values of each item and save plot to directory loc
ation
 png(filename="MyUtility.png", width=600, height=350)
 MyUtility<-MyUtility
 barplot(MyUtility,main="Plot of utility of items",ylab="u(x)",xlab="Item name (y[i])",names.
arg =y )
 dev.off()
 barplot(MyUtility,main="Plot of utility of items",ylab="u(x)",xlab="Item name (y[i])",names.
arg =y )
 MyUtility
}
# Tweak desired input 1 for student number, input 2 for upper contour list file name, input 3
 for lower contour list file name, and input 4 for directory
# *** Must use "\\" instead of "\" after copy and paste from directory to prevent Hex digits c
haracter string error
# My Example
MyInducedUtility(00280633,'MyUpperContourList.txt','MyLowerContourList.txt',"C:\\Users\
\UX330\\Documents\\F79PAProject\\H00280633\\")
```