Amanda

Meu nome é Amanda Rusiska Piovezani, sou bióloga pela Universidade Estadual do Norte do Paraná (UENP), mestre em bioinformática pelo Universidade de São Paulo (BIOINFO-USP) e doutoranda em bioinformática também pela Universidade de São Paulo (BIOINFO-USP).

O meu projeto de doutorado está intitulado como “DESENVOLVIMENTO DE UMA FERRAMENTA DE BIOINFORMÁTICA PARA INTEGRAÇÃO DAS ESCALAS DE TRANSCRIÇÃO GÊNICA, PROTEÔMICA E FISIOLOGIA UTILIZANDO SORGO E CANA-DE-AÇÚCAR COMO SISTEMAS MODELOS”. Para o seu desenvolvimento faz-se necessária uma intensa utilização de métodos estatísticos, os quais são eficientemente executados através do ambiente R, e portanto esta é a principal motivação para esta disciplina.

Meus Exercícios

exec

Propostas de Trabalho Final

Proposta A

Nome da Função: lplt()

Descrição: Uma função de Planejamento de Vida a Longo Prazo (Life Planning Long Term - “lplt”). Não se trata do cálculo da aposentadoria pública ou privada, ao invés disso, essa função mostra como deve ser feito um planejamento financeiro quando se deseja acumular uma renda extra a ser recebida no momento da aposentadoria.

Input: São considerados os valores de idade ao iniciar o planejamento, idade para se aposentar, valor mensal máximo que o usuário pode guardar, o valor mensal desejado para recebimento ao final e se disponível, o valor já acumulado até o momento deste cálculo.

Output: Uma lista contendo: a descrição do valor mensal a ser guardado, juntamente com o período de tempo necessário para que seja possível aposentar e receber mensalmente a quantia desejada. Se o usuário passou algum valor já acumulado, a função faz as devidas correções de valores considerando esta quantia.

Proposta B

Nome da Função: photo()

Descrição: Uma função a ser utilizada no estudo do processo fotossintético. Ela irá implementar a função bioquímica para o cálculo da velocidade máxima de carboxilação da enzima fosfoenolpiruvato

carboxilase (Vpmax) e da velocidade de regeneração do substrato dessa enzima (Vpr).

Input: A função deve receber um identificador da amostra (parâmetro 'ID'), dados de assimilação de CO2 (parâmetro 'A' ou 'photo') e quantidade de CO2 interno na célula (parâmetro 'ci').

Output: São os valores de Vpmax e Vpr.

Fundamentação teórica: Livro: Biochemical Models of Leaf Photosynthesis. Author: S. Von Caemmerer. Coleção: Techniques in Plant Sciences, nº2.

> As duas propostas parecem viáveis, mas eu achei a proposta B mais interessante porque lida com um problema biológico. Fique a vontade pra decidir qual realizar. —- *Leonardo*

Resolução Proposta A

Código da Função

```
#=========================================================================
# Discipline: Uso da Linguagem R para Análise de Dados em Ecologia - 2014
# R Function: Life Planning at Long Term. Date: May/2014
# Author: Amanda Rusiska Piovezani
#=========================================================================
# Description: Code of the function to Life Planning at Long Term (lplt).
# This is not the calculation of public or private retirement, instead,
# this function shows how could be done a financial planning when you
# wish to receive a monthly extra income at the moment of the retirement.

# For other information, please see the lplt.rd and README files.

lplt=function(a=0,b,c,d=0,max=0){ #receive the arguments
  value_received_by_month <- a
  #a = the value in which the user wish to receive monthly at the moment
  #of the retirement.
  age_moment <- b
  #b = age (years) at the moment of function calculation.
  age_retirement <- c #c = age (years) at the moment of the retirement.
  value_already_saved <- d #d = saved value before this calculation.
  #Default value equal zero.
  #max = maximum value which user can save monthly.
  #Calculation of time (in months) which user should save money in order
  #to receive the wished amount at the moment of retirement.
  years_saving_money <- age_retirement-age_moment
  time_to_save_money <- years_saving_money*12 #time in months
```

```
#Calculation of time (in months) to be received considering world life
#expectancy approximately equal to 70 years [1].
time_to_receive_money <- (70-age_retirement)*12 #time in months.
if(b != 0 & c != 0){ #arguments 'b' and 'c' should be passed on
  #beginning with value different from zero.
  ## First way to function execution: considers the maximum value that
  #user can save montly. ##
  if(max != 0){
    #Calculation of the total value that will be saved until retirement.
    total_value_using_maxValue <- max*time_to_save_money
    #Calculation of value that will be received monthly at the moment
    #of the retirement.
    value_received_monthly <- round(total_value_using_maxValue/
                                      time_to_receive_money, 2)
    #print function output.
    return (cat (
        "============================================================\n
         Plan your financial future, it helps to have a quiet old age!\n
         ============================================================\n"
         "Your Life Planning at Long Term includes:\n",
         "- Age at beginning planing:", age_moment ,"\n",
         "- Age at retirement:", age_retirement , "\n",
         "- Value that should be saved monthly (in your
            currency):", max, "for a period of", years_saving_money,
            "years\n",
         "- Value that will be received monthly (in your currency)
            on retirement:", value_received_monthly,"\n"))
  }
  else{
    ## Second way to function execution: considers the final value to be
    #received on retirement (argument 'a'), to it, the max argument
    #should has value equal zero. ##
    if(a != 0){
      #Calculation of the total value that should be saved until
      #retirement.
      total_value_to_save <- value_received_by_month*
        time_to_receive_money
      #Calculation of the value that should be saved monthly
      value_to_save_montly <- round((total_value_to_save/
                                       time_to_save_money), 2)
      #If some value was already saved before, it should be subtracted
      #from the total value to be save (total_value_to_save)
      #and after divided by the total number of months that user have
      #to save.
      value_to_save_montly <- round((
        total_value_to_save - value_already_saved)/time_to_save_money, 2)
      #print function output.
      return (cat (
        "============================================================\n
         Plan your financial future, it helps to have a quiet old age!\n
         ============================================================\n"
```

```
            "Your Life Planning at Long Term includes:\n",
            "- Age at beginning planing:", age_moment ,"\n",
            "- Age at retirement:", age_retirement , "\n",
            "- Value that should be saved monthly (in your
               currency):", value_to_save_montly, "for a period of",
               years_saving_money, "years\n",
            "- Value that will be received monthly (in your currency)
               on retirement:", value_received_by_month, "\n"))
      }
      else{
        stop("argument 'a' is mandatory when maximum value is zero, for
             help please see function documentation.\n")
      }
    }
  }
}
```

Página de Ajuda

R Documentation

```
lplt           R Documentation

Life Planning at Long Term

Description

A function of Life Planning at Long Term (lplt). This is not the
 calculation of public or private retirement, instead, this function
 shows how could be done a financial planning when you wish to receive
 a monthlly extra income at the moment of the retirement.


Usage

lplt(a = 0, b, c)

lplt(b, c, max=0)


Arguments

a   Value that user wish to receive monthly at the moment of the
        retirement (integer). Default value equal zero.

b   Age (years) at the moment of function calculation (integer).
        Mandatory argument.
```

```
c   Age (years) at the moment of the retirement (integer).
        Mandatory argument.

d   Value possibly saved before lplt calculation. Default value
        equal zero (integer).

max Maximum value that user can save monthly (integer).
        Default value equal zero.


Details

There are two ways of run this function: the first considers the
 amount of money which user would like receive monthly on the
 moment of retirement (argument 'a'), to then calculate how much
 money should be saved before this period. Different from the
 first, the second way considers a maximum value which user can
 save by month. From this value is calculated the total value
 that will be saved on period until retirement and the amount
 that will be received by month after it.

On function is used the world life expectancy value (integer)
 equal to 70 years, according to World Health Organization
 (2014)[1].


Value
Will print on the screen a text containing the values involved on function
calculation, that are:
- Age at beginning planing
- Age at retirement
- Value that should be saved monthly
- Value that will be received monthly


Author(s)

Amanda Rusiska Piovezani <arpiovezani@usp.br>


References

[1] World Health Organization. Department of Health Statistics
 and Information Systems (2014, March). WHO methods for life
 expectancy and healthy life expectancy. Retrieved from
 http://www.who.int/healthinfo/statistics/LT_method.pdf?ua=1.


Examples

example 1:
```

```
lplt(a=1000,b=27,c=65)

example 2:
lplt(b=31,c=60, max=600)

{ ~plan }
{ ~life }
```

README

```
#============================== FPLT README ==============================#
# Discipline: Uso da Linguagem R para Analise de Dados em Ecologia - 2014
# R Function: Life Planning at Long Term. Date: May/2014
# Author: Amanda Rusiska Piovezani  <arpiovezani@usp.br>
#=========================================================================

#==============
# Description
#==============
A function of Life Planning at Long Term (lplt). This is not the calcula
 tion of public or private retirement, instead, this function shows how
 could be done a financial planning when you wish to receive a monthlly
 extra income at the moment of the retirement.

#=========
# Input
#=========
The values of: age (years) which should be considered as the beginning of
 planning (argument 'b'), retirement age (argument 'c'), maximum monthly
 amount that you can save (argument 'max'), the required monthly amount
 for receiving the end (argument 'a') and if available, the user can pass
 the value already saved before this function calculation (argument 'd').

#=========
# Output
#=========
Will print on the screen a text containing the values involved on function
 calculation, that are:
- Age at beginning planing
- Age at retirement
- Value that should be saved monthly (in your currency) and the needed
  period for this deposit.
- Value that will be received monthly

#========================
# Content of lplt package
#========================
The 'README' file describes the object, input, output, how to execute the
```

```
 function and one example.
The 'lplt.r' file contains the function code.
The 'lplt.rd' file is the R Documentation

#===========
# Execution
#===========
Note: Commands that should be run from the UNIX shell (e.g., bash or csh)
 are prefixed with a '$' character. Commands that should be run from
 either R script or at the R interactive shell are prefixed with a '>'
 character.

1) Inside lplt directory, start a R session:
$ R

2) Make load of the function:
> source("lplt.r")

3) Execute the function passing your arguments:
example 1:
> lplt(a=3000,b=27,c=65)

4) Output from example 1:
Your Life Planning at Long Term includes:
 - Age at beginning planing: 27
 - Age at retirement: 65
 - Value that should be saved monthly (in your currency): 394.74  for a
   period of 38 years
 - Value that will be received monthly (in your currency) on
   retirement: 3000

#====================== README END ======================#
```

Baixar todos os arquivos da função

lplt.zip

From:
http://ecor.ib.usp.br/ - **ecoR**

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2014:alunos:trabalho_final:arpiovezani:start

Last update: **2020/08/12 06:04**