Matheus Januario Sousa

Aluno de graduação, atualmente fazendo iniciação científica no depto. de Ecologia do IB USP. Pretende investigar se/como a seleção sexual guiou a especiação e extinção dentro do clado dos ruminantes.

exec

Propostas de trabalho final de Matheus

As propostas:

A proposta A possui mais utilidade ao meu trabalho no laboratório, uma vez que envolve a manipulação de objetos da classe “phylo” (um tipo específico de lista no R), enquanto que a proposta B não tem nenhuma aplicação direta em meu trabalho. Por este motivo, eu estou pessoalmente inclinado a preferir a proposta A, apesar de achar a B igualmente divertida de se fazer.

Proposta A:

A *estatística gamma* (Pybus & Harvey, 2000) é uma medida simples para se avaliar mudança na especiação de um clado, sendo que valores negativos de *gamma* menores que -1.645 sugerem que a taxa de especiação foi inicialmente alta, mas depois diminuiu. Apesar das facilidades, a *estatística gamma* apresenta um problema, já que filogenias com diferentes números de teaxons apresentam diferentes valores meaximos de *gamma*, o que pode impedir uma comparação entre clados. Apesar de pelo menos 2 pcotes do R apresentarem funções que calculam a *estatística gamma*, nenhum apresentou uma correção para este problema do número de terminais.

Em minha função no R, pretendo implementar não apenas o método, mas também a correção previamente citada.

O *gamma* pode ser calculado por meio das distâncias entre os nós de uma filogenia, por meio do seguinte cálculo:

Onde *g1, g2 ... gn* é cada distância entre nós de uma filogenia qualquer

A função vai receber um objeto do tipo phylo (um subtipo específico de objeto do tipo lista), calcular cada *g* por meio da função já disponível no pacote “ape” do R, branching.times() e apresentar o *gamma* corrigido por número de terminais na filogenia.

> Oi Matheus... Gostei da sua ideia. Achei que sua função tem

> potencial! Mas você poderia deixar mais claro como vc pretende calcular o gama corrigido? Acabei entendendo o resto da sua função pq a Laura Alencar me explicou hehehe, então tenta deixar mais claro =] —-*Thais Lopes*

Proposta B:

No Brasil, principalmente no ciclo social universitário, *"churrasco"* se refere à todo tipo de evento onde o intuito principal é comer carne assada. Entretanto, essa definição não exclui participantes vegetarianos e veganos, já que também é possível assar legumes. Mesmo assim, é comum acontecer um conflito entre os participantes do churrasco, já que frequentemente vegetarianos exigem pagar um valor menor para participar do evento, pois o gasto *per capita* de um vegetariano (por conta do consumo exclusivo de legumes) é menor que o de um carnívoro. Os adeptos da dieta vegana também enfrentam o mesmo tipo de problema.

Uma forma simples de resolver quanto se deve gastar por evento é criando um função baseada na quantidade de alimento que cada tipo de dieta costuma comer num evento de churrasco, ponderando o gasto por hora e por itens "adicionais" que se deseja ter no evento (sejam estes itens pré derterminados, como pão-de-alho, ou um tipo de item novo, que o usuário terá que descrever para a função).

A função que essa proposta almeja recebe alguns argumentos para saber se o churrasco é da categoria "adulto", "universitario leve" ou "universitario pesado", e quantos participantes de cada tipo de dieta ("carnivoro", "vegetariano" ou "vegano") irão participar do evento. A função também leva em consideração o período que o churrasco é estimado em durar, e retorna a quantidade de cada ingrediente a ser comprada, junto com o valor a ser pago por individuo, levando em consideração a dieta do mesmo.

> Esta proposta também é divertida. Talvez vc vá utilizar as funções aprendidas na aula mais com esta função. Mas se a outra tem mais aplicabilidade pra seu trabalho, investe nela!!! —-*Thais Lopes*

Trabalho final de Matheus

Página de Ajuda / Help

```
churrascada                package:nenhum
R Documentation
```

```
Planejar um churrasco

Description

churrascada cria um data frame, informando que itens devem ser comprados num
churrasco, quanto sera gasto com cada item, a quantidade a ser comprada de
cada item e o quanto cada pessoa deve pagar se desejar cobrir a sua parte da
conta.
Tudo isso leva em consideracao a quantidade relativa de pessoas que seguem
cada tipo de dieta, o contexto social do churrasco e a duracao do mesmo.

Usage

churrascada = function (categoria, carnivoros, vegetarianos, veganos,
duracao, itens_removidos = rep(FALSE, times=21), precos_por_item = NULL)

Arguments:

categoria       um vetor de "character" contendo o nome por extenso da
categoria do churrasco (ou seja: o contexto social em que o churrasco
acontecera)
carnivoros      um valor do tipo "numeric", que diz respeito ao numero de
carnivoros que vao comparecer ao churrasco
vegetarianos    um valor do tipo "numeric", que diz respeito ao numero de
vegetarianos que vao comparecer ao churrasco
veganos         um valor do tipo "numeric", que diz respeito ao numero de
veganos que vao comparecer ao churrasco
duracao         um valor do tipo "numeric", que diz respeito ao numero de
horas que o churrasco avi durar, em horas
preco_por_item  um vetor do tipo "numeric" contendo 21 elementos
itens_removidos um vetor do tipo "logic", contendo 21 elementos, apenas TRUE
e FALSE

Details:

O argumento "categoria" aceita apenas uma dentre as opcoes "universitario
leve", "universitario pesado", "customizado" e "adulto".

A indexacao correspondente entre ""itens_removidos" e cada componente da
lista de compras que pode ser removido é:

1. Carne_de_Vaca(Kg)
2. carne_de_Frango(Kg)
3. Carne_de_porco(Kg)
4. Cerveja(lata)
5. Gelatina(pacote)
6. Queijo_coalho
7. Pao_de_alho
8. Cebola
9. Batata
10. Tomate
```

```
11. Berinjela
12. Abobrinha
13. Pimentao
14. Refrigerante
15. Pao
16. Plasticos
17. Carvao
18. Vinagre
19. Destilados
20. Azeite
21. Arroz

Se for usada a opcao "customizado" do argumento "categoria", deve
NECESSARIAMENTE apresentar o conteudo de"itens_removidos".

O conteudo de "itens_removidos" so precisa ser apresentado se for
selecioanda a opcao "customizado" do argumento "categoria"
Se os valores de "carnivoros", "vegetarianos" e "veganos" forem todos iguais
a zero. A funcao retorna uma mensagem de erro (nao existem churrasco sem
ninguem)

O argumento "duracao" ano pode ter valores menores ou iguais a zero (pois e
um intervalo de tempo)

Note:

A quantidade de cada item nao esta arredondada!! Isso permite que o
comprador tenha uma atitude maleavel quanto a quantidades (promocoes de
packs de cerveja, pr exemplo), ou leve em consideracao o peso medio de
vegetais (ja que em muitos supermercados o vegetal e pago por peso, nao por
unidade)
Na hora de cobrar os amigos pelo custo do churrasco, nao se esqueca de
refazer as contas!!!

Author:

Matheus Januario Sousa
mjlopessousa@gmail.com

Examples:


#### Uma tipica festa de aniversario de um universitario de biologia:
churrascada(categoria = "universitario leve", carnivoros = 25, vegetarianos
= 2, veganos = 1, duracao = 8, precos_por_item = c(25, 10, 10, 2.25, 1.5,
15.5, 8, 3.8, 3.8, 5, 3.8, 3.8, 3.8, 2.5, 0.5, 20, 6, 7, 20, 11, 3))



### Um churrasco de familia, porem com cerveja
churrascada(categoria = "adulto", carnivoros = 25, vegetarianos = 0, veganos
```

```
= 0, duracao = 8, precos_por_item = c(25, 10, 10, 2.25, 1.5, 15.5, 8, 3.8,
3.8, 5, 3.8, 3.8, 3.8, 2.5, 0.5, 20, 6, 7, 20, 11, 3))

### Um churrasco censura 12 anos:
churrascada(categoria = "customizado", carnivoros = 25, vegetarianos = 2,
veganos = 0, duracao = 8, precos_por_item = c(25, 10, 10, 2.25, 1.5, 15.5,
8, 3.8, 3.8, 5, 3.8, 3.8, 3.8, 2.5, 0.5, 20, 6, 7, 20, 11, 3),
itens_removidos = c(F,F,F,T,rep(F, times=14),T,F,F ))
```

A funcao "churrascada"

```
#Funcao churrascada

churrascada = function (categoria, carnivoros, vegetarianos, veganos,
duracao, itens_removidos = rep(FALSE, times=21), precos_por_item = NULL)
#cria uma funcao chamada churrascada, com os argumentos: "categoria",
"carnivoros", "vegetarianos", "veganos", "duracao", "itens_removidos", e
"precos_por_item"
{
#################################################################
##testando se as entradas estao certas para a funcao comecar##
#################################################################

  if(!inherits(categoria,"character"))   #se a classe de "categoria" nao
for"numeric":
    {
      stop("\"categoria\" foi especificado incorretamente. Por favor, leia o
help da funcao")   #para o programa, e imprime a mensagem de erro entre
parenteses
    }
  categoria=match.arg(categoria, c("universitario leve", "universitario
pesado", "adulto", "customizado"), several.ok=TRUE)   #"categoria" nao
aceitara conteudo diferente das possibilidades ("universitario leve",
"universitario pesado", "customizado", ou "adulto") aceitas pela funcao
  if(!inherits(carnivoros, "numeric") || carnivoros < 0)   #se a classe de
"carnivoros" nao for "numeric", ou se "carnivoros" conter valores menores
que zero:
    {
      stop("\"carnivoros\" foi especificado incorretamente. Por favor, leia
o help da funcao")   #para o programa, e imprime a mensagem de erro entre
parenteses
    }
  if(!inherits(vegetarianos, "numeric") || vegetarianos < 0)   #se a classe
de "vegetarianos" nao for "numeric", ou se "vegetarianos" conter valores
menores que zero:
    {
      stop("\"vegetarianos\" foi especificado incorretamente. Por favor,
leia o help da funcao")   #para o programa, e imprime a mensagem de erro
```

```
entre parenteses
    }
  if(!inherits(veganos, "numeric") || veganos < 0)   #se a classe de
"veganos" nao for "numeric", ou se "veganos" conter valores menores que
zero:
    {
      stop("\"veganos\" foi especificado incorretamente. Por favor, leia o
help da funcao")   #para o programa, e imprime a mensagem de erro entre
parenteses
    }
  if(carnivoros == 0 && vegetarianos == 0 && veganos == 0)   #Se os valores
de "carnivoros" E de "vegetarianos" E de "veganos" forem iguais a zero:
    {
      stop("A dieta dos participantes do churrasco nao foi especificada
corretamente. Por favor, leia o help da funcao")   #para o programa, e
imprime a mensagem de erro entre parenteses
    }
  if(!inherits(duracao, "numeric") || duracao <= 0)   #se a classe de
"duracao" nao for "numeric", ou se "duracao" conter valores iguais ou
menores que zero:
    {
      stop("\"duracao\" foi especificado incorretamente. Por favor, leia o
help da funcao")   #para o programa, e imprime a mensagem de erro entre
parenteses
    }
  if(!inherits(itens_removidos, "logical") || length(itens_removidos) != 21)
#se a classe de "itens_removidos" nao for"logical", ou se o seu comprimento
nao for exatamente igual a 21:
    {
      stop("\"itens_removidos\" foi especificado incorretamente. Por favor,
leia o help da funcao")   #para o programa, e imprime a mensagem de erro
entre parenteses
    }
  if(!inherits(precos_por_item, c("numeric", "NULL")) || precos_por_item <=
0)   #se a classe de "precos_por_item" nao for "numeric" (ou "NULL"), ou se
"precos_por_item" conter valores iguais ou menores que zero:
    {
      stop("\"precos_por_item\" foi especificado incorretamente. Por favor,
leia o help da funcao")   #para o programa, e imprime a mensagem de erro
entre parenteses
    }
  if(is.null(precos_por_item))   #se nao houver valor em "precos_por_item"
(o que e o default da funcao), faca:
    {
      stop("\"precos_por_item\" nao foi especificado. Por favor, leia o help
da funcao")   #para o programa, e imprime a mensagem de erro entre
parenteses
    }
  if(class(precos_por_item) == "numeric")   #se a classe de
"precos_por_item" for "numeric", faca:
```

```
    {
    dados = data.frame(V1 = precos_por_item, V2 = itens)   #cria um data
frame chamado "dados", contendo os objetos "precos_por_item" e "itens"
    }

##objetos internos da funcao:
  itens = c("Carne_de_Vaca(Kg)", "carne_de_Frango(Kg)",
"Carne_de_porco(Kg)", "Cerveja(lata)",
"Gelatina(pacote)","Queijo_coalho(pacote)","Pao_de_alho(pacote)","Cebola(uni
d.)", "Batata(unid.)", "Tomate(unid.)", "Berinjela(unid.)",
"Abobrinha(unid.)", "Pimentao(unid.)",
"Refrigerante(L)","Pao(unid.)","Plasticos
diversos","Carvao(Kg)","Vinagre(unid.)","Destilados(L)","Azeite(unid.)",
"Arroz(Kg)") #cria um objeto chamado "itens", contendo os nomes de todos os
itens que podem ser incluidos no churrasco
  fim = c(1)   #cria um objeto de comprimento 1, chamado "fim" (so sera
usado no final da funcao)
  fim2 = c(1)   #cria um objeto de comprimento 1, chamado "fim2" (so sera
usado no final da funcao)
  resultado = data.frame(quantidade = rep(NA, times=length(itens)),
valor_gasto_por_item = c(rep(NA)))   #cria um data frame chamado
"resultado", preenchido totalmente por NA, com o comprimento do objeto
"itens" e com duas colunas: "quantidade" e "valor_gasto_por_item"
  row.names(resultado) = itens   #faz com que os nomes das linhas de
"resultado" seja igual ao conteudo de "itens"
  carn = c(rep(1, times=5), rep(0.5, times=2), 0, 0, 0.5, 0, 0, 0, rep(1,
times=8))   #cria um objeto com a proporcao relativa (as outras dietas) com
que a dieta de uma pessoa carnivora come cada um dos itens de um churrasco;
armazena essa proporcao no objeto "carn". Reflete a opiniao pessoal de quem
escreveu o codigo
  vege = c(rep(0, times=3), rep(1, times=18))   #cria um objeto com a
proporcao relativa (as outras dietas) com que a dieta de uma pessoa
vegetariana come cada um dos itens de um churrasco; armazena essa proporcao
no objeto "vege". Reflete a opiniao pessoal de quem escreveu o codigo
  vega = c(rep(0, times=6), rep(1, times=15))   #cria um objeto com a
proporcao relativa (as outras dietas) com que a dieta de uma pessoa vegana
come cada um dos itens de um churrasco; armazena essa proporcao no objeto
"vega". Reflete a opiniao pessoal de quem escreveu o codigo
  consumo = c((carnivoros*carn)+(vegetarianos*vege)+(veganos*vega))
#calcula a proporcao relativa de cada item que sera consumido no churrasco,
levando em consideracao as proporcoes dentro de "carn", "vege", e "vega";
armazena esse calculo no objeto "consumo"
  porcoes = c(0.12, 0.12, 0.14, 7.2, 0.4, 0.2, 0.2, 0.5, 0.5, 1.5, 0.5, 0.5,
0.5, 1.2, 1, 0.025, (((0.24*duracao)/8)/(carnivoros+vegetarianos+veganos)),
1/(carnivoros+vegetarianos+veganos), 0.12,
(1/(carnivoros+vegetarianos+veganos)), 0.08)   # cria um bjeto chamado
"porcoes", contendo a porcao media que uma pessoa normalmente come de cada
um dos itens em 8 horas. Reflete a opiniao pessoal de quem escreveu o codigo

###########################################################
##separando pela categoria de churrasco que sera feito##
```

```
#########################################################

  #categoria adulto:
  if (categoria == "adulto") #se o objeto categoria conter a string
"adulto", faca:
    {
      n = 19   #armazena o valor das posicoes dos itens que nao serao
comprados nesta categoria de churrasco. Reflete a opiniao pessoal de quem
escreveu o codigo
      porcoes[4] = porcoes[4]/2   #multiplica por 1/2 a quantidade de
cerveja a ser consumida nesta categoria de churrasco. Reflete a opiniao
pessoal de quem escreveu o codigo
      resultado = resultado[-n,]   #retira o conteudo de "resultado", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      carn = carn[-n]   #retira o conteudo do vetor "carn", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vege = vege[-n]   #retira o conteudo do vetor "vege", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vega = vega[-n]   #retira o conteudo do vetor "vega", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      dados = dados[-n,]   #retira o conteudo de "dados, nas mesmas posicoes
dos itens que nao serao comprados nesta categoria de churrasco.
      consumo = consumo[-n]   #retira o conteudo do vetor "consumo", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      porcoes = porcoes[-n]   #retira o conteudo do vetor "porcoes", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      resultado$quantidade = consumo * porcoes * (duracao/8)   #multiplica o
consumo total de  porcoes pela quantidade de porcoes e pela duracao do
churrasco. Armazena esse resultado em "resultado$quantidade"
      resultado$valor_gasto_por_item = (dados$V1)*(resultado$quantidade)
#multiplica a quantidade de cada produto que deve ser comprada pelo preco de
cada produto. Armazena isso em "resultado$valor_gasto_por_item"
      valor_total = (sum(resultado$quantidade*dados$V1))   #soma o valor de
cada item a ser comprado, guarda esse valor em "valor_total"
      fim[1] = c("O valor total a ser pago por todos os ingredientes e:")
#coloca a frase dentro dos parenteses no objeto "fim"
      fim2[1] = valor_total   #colocar o valor de "valor_total" no objeto
"fim2"
        #calculando os precos individuais por dieta
          #carnivoros
          if(carnivoros != "0")   #se houver carnivoros indo pro churrasco,
faca:
            {
              preco_carnivoro = porcoes*dados$V1*carn   #calcula o preco que
cada carnivoro vai pagar por item que vai consumir, e armazena no objeto
"preco_carnivoro"
              preco_carnivoro = sum(preco_carnivoro)   #soma o valor de
```

```
todos os itens que um carnivoro vai consumir, armazena no objeto
"preco_carnivoro"
              fim[length(fim)+1] = c("O valor a ser pago por cada carnivoro
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_carnivoro   #coloca o preco que
cada carnivoro individual deve pagar, se for contribuir com a sua parte do
churrasco tambem no objeto "fim"
            }
          #vegetarianos
          if(vegetarianos != "0")   #se houver vegetarianos indo pro
churrasco, faca:
            {
              preco_vegetariano = porcoes*dados$V1*vege   #calcula o preco
que cada vegetariano vai pagar por item que vai consumir, e armazena no
objeto "preco_vegetariano"
              preco_vegetariano = sum(preco_vegetariano)   #soma o valor de
todos os itens que um vegetariano vai consumir, armazena no objeto
"preco_vegetariano"
              fim[length(fim)+1] = c("O valor a ser pago por cada
vegetariano e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = (preco_vegetariano)   #coloca o preco
que cada vegetariano individual deve pagar, se for contribuir com a sua
parte do churrasco tambem no objeto "fim"
            }
          #veganos
          if(veganos != "0")   #se houver veganos indo pro churrasco, faca:
            {
              preco_vegano = porcoes*dados$V1*vega#calcula o preco que cada
vegano vai pagar por item que vai consumir, e armazena no objeto
"preco_vegano"
              preco_vegano = sum(preco_vegano)   #soma o valor de todos os
itens que um vegano vai consumir, armazena no objeto "preco_vegano"
              fim[length(fim)+1] = c("O valor a ser pago por cada vegano
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_vegano   #coloca o preco que cada
vegano individual deve pagar, se for contribuir com a sua parte do churrasco
tambem no objeto "fim"
            }
      final = data.frame(fim, fim2)   #cria um data frame com a resposta
final do programa, usando o conteudo dos objetos "fim" e "fim2"
      colnames(final) = c("condicao", "valor(R$)")   #renomeia  as colunas
do data frame "final"
      cat("\n")   #pula uma linha
      cat("\n")   #pula uma linha
      cat("A lista de compras e:\n")   #imprime a mensagem dentro dos
parenteses no terminal
      cat("\n")   #pula uma linha
      return(list(resultado, final)) #imprime a lista de compras e o data
frame "final" e encerra o programa
  }
  ##########################
```

```
  #### Fim do programa! ####
  ##########################
  #categoria "universitario pesado"
  if (categoria == "universitario pesado")
    {
      n = c(1, 5, 6, 7)   #armazena o valor das posicoes dos itens que nao
serao comprados nesta categoria de churrasco. reflete a opiniao pessoal de
quem escreveu o codigo
      porcoes[4] = (10) #muda a porcao esperada de cerveja que cada pessoa
deve beber
      resultado = resultado[-n,]    #retira o conteudo de "resultado", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      carn = carn[-n]   #retira o conteudo do vetor "carn", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vege = vege[-n]   #retira o conteudo do vetor "vege", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vega = vega[-n]   #retira o conteudo do vetor "vega", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      dados = dados[-n,]   #retira o conteudo de "dados, nas mesmas posicoes
dos itens que nao serao comprados nesta categoria de churrasco.
      consumo = consumo[-n]    #retira o conteudo do vetor "consumo", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      porcoes = porcoes[-n]   #retira o conteudo do vetor "porcoes", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      resultado$quantidade = consumo * porcoes * (duracao/8)   #multiplica o
consumo total de  porcoes pela quantidade de porcoes e pela duracao do
churrasco. Armazena esse resultado em "resultado$quantidade"
      resultado$valor_gasto_por_item = (dados$V1)*(resultado$quantidade)
#multiplica a quantidade de cada produto que deve ser comprada pelo preco de
cada produto. Armazena isso em "resultado$valor_gasto_por_item"
      valor_total = (sum(resultado$quantidade*dados$V1))
      fim[1] = c("O valor total a ser pago por todos os ingredientes e:")
#coloca a frase dentro dos parenteses no objeto "fim"
      fim2[1] = valor_total   #colocar o valor de "valor_total" no objeto
"fim2"
        #calculando os precos individuais por dieta
          #carnivoros
          if(carnivoros != "0")   #se houver carnivoros indo pro churrasco,
faca:
            {
              preco_carnivoro = porcoes*dados$V1*carn   #calcula o preco que
cada carnivoro vai pagar por item que vai consumir, e armazena no objeto
"preco_carnivoro"
              preco_carnivoro = sum(preco_carnivoro)   #soma o valor de
todos os itens que um carnivoro vai consumir, armazena no objeto
"preco_carnivoro"
              fim[length(fim)+1] = c("O valor a ser pago por cada carnivoro
```

```
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_carnivoro   #coloca o preco que
cada carnivoro individual deve pagar, se for contribuir com a sua parte do
churrasco tambem no objeto "fim"
            }
          #vegetarianos
          if(vegetarianos != "0")   #se houver vegetarianos indo pro
churrasco, faca:
            {
              preco_vegetariano = porcoes*dados$V1*vege   #calcula o preco
que cada vegetariano vai pagar por item que vai consumir, e armazena no
objeto "preco_vegetariano"
              preco_vegetariano = sum(preco_vegetariano)   #soma o valor de
todos os itens que um vegetariano vai consumir, armazena no objeto
"preco_vegetariano"
              fim[length(fim)+1] = c("O valor a ser pago por cada
vegetariano e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = (preco_vegetariano)   #coloca o preco
que cada vegetariano individual deve pagar, se for contribuir com a sua
parte do churrasco tambem no objeto "fim"
            }
          #veganos
          if(veganos != "0")   #se houver veganos indo pro churrasco, faca:
            {
              preco_vegano = porcoes*dados$V1*vega#calcula o preco que cada
vegano vai pagar por item que vai consumir, e armazena no objeto
"preco_vegano"
              preco_vegano = sum(preco_vegano)   #soma o valor de todos os
itens que um vegano vai consumir, armazena no objeto "preco_vegano"
              fim[length(fim)+1] = c("O valor a ser pago por cada vegano
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_vegano   #coloca o preco que cada
vegano individual deve pagar, se for contribuir com a sua parte do churrasco
tambem no objeto "fim"
            }
      final = data.frame(fim, fim2)   #cria um data frame com a resposta
final do programa, usando o conteudo dos objetos "fim" e "fim2"
      colnames(final) = c("condicao", "valor(R$)")   #renomeia  as colunas
do data frame "final"
      cat("\n")   #pula uma linha
      cat("\n")   #pula uma linha
      cat("A lista de compras e:\n")   #imprime a mensagem dentro dos
parenteses no terminal
      cat("\n")   #pula uma linha
      return(list(resultado, final)) #imprime a lista de compras e o data
frame "final" e encerra o programa
      cat("\n")   #pula uma linha
      cat("Bon apetit!")   #imprime a mensagem dentro dos parenteses
  }
  ############################
  #### Fim do programa! ####
```

```
  ############################

  #categoria "customizado"
  if (categoria == "customizado")
    {
      n = which(itens_removidos==TRUE)   #armazena o valor das posicoes dos
itens que nao serao comprados nesta categoria de churrasco (neste caso, e
dado pelo vetor booleano do argumento "itens_removidos" desta funcao!)
      porcoes[4] = (0.3*12)
      resultado = resultado[-n,]    #retira o conteudo de "resultado", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      carn = carn[-n]   #retira o conteudo do vetor "carn", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vege = vege[-n]   #retira o conteudo do vetor "vege", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      vega = vega[-n]   #retira o conteudo do vetor "vega", nas mesmas
posicoes dos itens que nao serao comprados nesta categoria de churrasco.
      dados = dados[-n,]   #retira o conteudo de "dados, nas mesmas posicoes
dos itens que nao serao comprados nesta categoria de churrasco.
      consumo = consumo[-n]    #retira o conteudo do vetor "consumo", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      porcoes = porcoes[-n]   #retira o conteudo do vetor "porcoes", nas
mesmas posicoes dos itens que nao serao comprados nesta categoria de
churrasco.
      resultado$quantidade = consumo * porcoes * (duracao/8)   #multiplica o
consumo total de  porcoes pela quantidade de porcoes e pela duracao do
churrasco. Armazena esse resultado em "resultado$quantidade"
      resultado$valor_gasto_por_item = (dados$V1)*(resultado$quantidade)
#multiplica a quantidade de cada produto que deve ser comprada pelo preco de
cada produto. Armazena isso em "resultado$valor_gasto_por_item"
      valor_total = (sum(resultado$quantidade*dados$V1))   #soma o valor de
cada item a ser comprado, guarda esse valor em "valor_total"
      fim[1] = c("O valor total a ser pago por todos os ingredientes e:")
#coloca a frase dentro dos parenteses no objeto "fim"
      fim2[1] = valor_total   #colocar o valor de "valor_total" no objeto
"fim2"
        #calculando os precos individuais por dieta
          #carnivoros
          if(carnivoros != "0")   #se houver carnivoros indo pro churrasco,
faca:
            {
              preco_carnivoro = porcoes*dados$V1*carn   #calcula o preco que
cada carnivoro vai pagar por item que vai consumir, e armazena no objeto
"preco_carnivoro"
              preco_carnivoro = sum(preco_carnivoro)   #soma o valor de
todos os itens que um carnivoro vai consumir, armazena no objeto
"preco_carnivoro"
              fim[length(fim)+1] = c("O valor a ser pago por cada carnivoro
```

```
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_carnivoro   #coloca o preco que
cada carnivoro individual deve pagar, se for contribuir com a sua parte do
churrasco tambem no objeto "fim"
            }
          #vegetarianos
          if(vegetarianos != "0")   #se houver vegetarianos indo pro
churrasco, faca:
            {
              preco_vegetariano = porcoes*dados$V1*vege   #calcula o preco
que cada vegetariano vai pagar por item que vai consumir, e armazena no
objeto "preco_vegetariano"
              preco_vegetariano = sum(preco_vegetariano)   #soma o valor de
todos os itens que um vegetariano vai consumir, armazena no objeto
"preco_vegetariano"
              fim[length(fim)+1] = c("O valor a ser pago por cada
vegetariano e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = (preco_vegetariano)   #coloca o preco
que cada vegetariano individual deve pagar, se for contribuir com a sua
parte do churrasco tambem no objeto "fim"
            }
          #veganos
          if(veganos != "0")   #se houver veganos indo pro churrasco, faca:
            {
              preco_vegano = porcoes*dados$V1*vega#calcula o preco que cada
vegano vai pagar por item que vai consumir, e armazena no objeto
"preco_vegano"
              preco_vegano = sum(preco_vegano)   #soma o valor de todos os
itens que um vegano vai consumir, armazena no objeto "preco_vegano"
              fim[length(fim)+1] = c("O valor a ser pago por cada vegano
e:") #coloca a frase dentro dos parenteses no objeto "fim"
              fim2[length(fim2)+1] = preco_vegano   #coloca o preco que cada
vegano individual deve pagar, se for contribuir com a sua parte do churrasco
tambem no objeto "fim"
            }
      final = data.frame(fim, fim2)   #cria um data frame com a resposta
final do programa, usando o conteudo dos objetos "fim" e "fim2"
      colnames(final) = c("condicao", "valor(R$)")   #renomeia  as colunas
do data frame "final"
      cat("\n")   #pula uma linha
      cat("\n")   #pula uma linha
      cat("A lista de compras e:\n")   #imprime a mensagem dentro dos
parenteses no terminal
      cat("\n")   #pula uma linha
      return(list(resultado, final)) #imprime a lista de compras e o data
frame "final" e encerra o programa
  }
  ############################
  #### Fim do programa! ####
  ############################
```

```
  #categoria "universitario leve"
  resultado$quantidade = consumo * porcoes * (duracao/8)
  #--------e aqui!!!!
  resultado$valor_gasto_por_item = (dados$V1)*(resultado$quantidade)
  valor_total = (sum(resultado$quantidade*dados$V1))
  fim[1] = c("O valor total a ser pago por todos os ingredientes e:")
#coloca a frase dentro dos parenteses no objeto "fim"
  fim2[1] = valor_total   #colocar o valor de "valor_total" no objeto "fim2"
    #calculando os precos individuais por dieta
      #carnivoros
      if(carnivoros != "0")   #se houver carnivoros indo pro churrasco,
faca:
        {
          preco_carnivoro = porcoes*dados$V1*carn   #calcula o preco que
cada carnivoro vai pagar por item que vai consumir, e armazena no objeto
"preco_carnivoro"
          preco_carnivoro = sum(preco_carnivoro)   #soma o valor de todos os
itens que um carnivoro vai consumir, armazena no objeto "preco_carnivoro"
          fim[length(fim)+1] = c("O valor a ser pago por cada carnivoro e:")
#coloca a frase dentro dos parenteses no objeto "fim"
          fim2[length(fim2)+1] = preco_carnivoro   #coloca o preco que cada
carnivoro individual deve pagar, se for contribuir com a sua parte do
churrasco tambem no objeto "fim"
        }
    #vegetarianos
      if(vegetarianos != "0")   #se houver vegetarianos indo pro churrasco,
faca:
        {
          preco_vegetariano = porcoes*dados$V1*vege   #calcula o preco que
cada vegetariano vai pagar por item que vai consumir, e armazena no objeto
"preco_vegetariano"
          preco_vegetariano = sum(preco_vegetariano)   #soma o valor de
todos os itens que um vegetariano vai consumir, armazena no objeto
"preco_vegetariano"
          fim[length(fim)+1] = c("O valor a ser pago por cada vegetariano
e:") #coloca a frase dentro dos parenteses no objeto "fim"
          fim2[length(fim2)+1] = (preco_vegetariano)   #coloca o preco que
cada vegetariano individual deve pagar, se for contribuir com a sua parte do
churrasco tambem no objeto "fim"
        }
    #veganos
      if(veganos != "0")   #se houver veganos indo pro churrasco, faca:
        {
          preco_vegano = porcoes*dados$V1*vega#calcula o preco que cada
vegano vai pagar por item que vai consumir, e armazena no objeto
"preco_vegano"
          preco_vegano = sum(preco_vegano)   #soma o valor de todos os itens
que um vegano vai consumir, armazena no objeto "preco_vegano"
          fim[length(fim)+1] = c("O valor a ser pago por cada vegano e:")
#coloca a frase dentro dos parenteses no objeto "fim"
```

```
          fim2[length(fim2)+1] = preco_vegano   #coloca o preco que cada
vegano individual deve pagar, se for contribuir com a sua parte do churrasco
tambem no objeto "fim"
        }
  final = data.frame(fim, fim2)   #cria um data frame com a resposta final
do programa, usando o conteudo dos objetos "fim" e "fim2"
  colnames(final) = c("condicao", "valor(R$)")   #renomeia  as colunas do
data frame "final"
  cat("\n")   #pula uma linha
  cat("A lista de compras e:")   #imprime a mensagem dentro dos parenteses
no terminal
  cat("\n")   #pula uma linha
  return(list(resultado, final)) #imprime a lista de compras e o data frame
"final" e encerra o programa
################################################
####fim do programa! - dessa vez de verdade!####
################################################

#Bom churrasco, e obrigado pelo suporte na disciplina!

                                                    #Matheus Januario Sousa
}
```

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

Permanent link:
http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2015:alunos:trabalho_final:matheussousa:start

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