

```
#Script dissertação Ariane
#Data 29/05/2017
```

```
#1 - agrupamento dos alimentos do QFA;
#2 - análise fatorial para gerar os padrões alimentares;
#3 - categorizar cada padrão alimentar em relação ao grau de adesão (baixo ou alto) de acordo com os escores de consumo individual, que foram divididos em tercís;
#4 - regressão logística bruta entre cada padrão e as variáveis sociais - as associações com p-valores <0,20 foram conduzidas a ajustada;
#5 - regressão ajustada;
```

```
library(epiDisplay)
library(sjPlot)
library(psych)
library(ggplot2)
library(reshape2)
library(plyr)
library(GPArotation)
```

```
setwd("E:/PROSAUDE16/Prosaude/Replicações alunos/Ariane")
```

```
banco <- read.spss("ariane_padrões2.sav", to.data.frame=TRUE)
attach(banco)
```

```
# Agrupamento dos alimentos segundo características nutricionais
```

```
banco$Arroz <- as.integer(ArrozF)
banco$massas <- rowSums(cbind(MacarraoF, LasanhaF), na.rm=T)
```

```
banco$paes <- rowSums(cbind(PaoF, BiscoitoSalgadoF), na.rm=T)
```

```
banco$frutas <- rowSums(cbind(LaranjaF, BananaF, MamaoF, MacaF, MelanciaF,
                             AbacaxiF, MangaF, LimaoF, MaracujaF, UvaF), na.rm=T)
```

```
banco$hortalicas <- rowSums(cbind(alfaceF, CouveF, RepolhoF, CouveFlorF, TomateF,
                                  PepinoF, ChuchuF, AbobrinhaF, AboboraF, CenouraF,
                                  BeterrabaF, QuiaboF, VagemF, PimentaoF, AlhoF,
                                  CebolaF), na.rm=T)
```

```
banco$vegetais_conserva <- as.integer(EnlatadosF)
banco$bovina_miudos <- rowSums(cbind(CarneBoiF, BuchoF, ChurrascoF), na.rm=T)
banco$suina <- as.integer(CarnePorcoF)
banco$carnes_processadas <- rowSums(cbind(SardinhaF, BaconF, CarneConservaF), na.rm=T)
banco$frios <- as.integer(FriosF)
```

```

banco$aves_pescados <- rowSums(cbind(FrangoF,PeixeFrescoF), na.rm=T)
banco$leite_derivados <- rowSums(cbind(QueijoF,RequeijaoF,LeiteF,IogurteF), na.rm=T)
banco$Ovos <- as.integer(OvosF)
banco$feijao <- as.integer(FeijaoF)

banco$gorduras <- rowSums(cbind(ManteigaF, MaioneseF), na.rm=T)

banco$doces <- rowSums(cbind(BoloF, BiscoitoRecheadoF,BiscoitoDoceF,SorveteF,BalasF,ChocolatePoF,
                             ChocolateBarraF,DoceBaseLeiteF,DoceBaseFrutaF,AcucarF), na.rm=T)
banco$salgados <- rowSums(cbind(PizzaF, SalgadosRisoliF), na.rm=T)
banco$cafe <- rowSums(cbind(CafeF, ChaF), na.rm=T)
banco$refrigerantes <- rowSums(cbind(RefrigeranteColaF, OutrosRefrigerantesF), na.rm=T)
banco$sucos <- as.integer(SucoFrutaF)
banco$bebidas_alcool <- rowSums(cbind(VinhoF, CervejaF, OutrasBebidasF), na.rm=T)
banco$petiscos <- rowSums(cbind(BatataFritaF,SalgadinhosCheetosF,PipocaF,AmendoimF), na.rm=T)
banco$leguminosas <- as.integer(LentilhaF)
banco$cereais <- as.integer(PolentaF)
banco$Tuberculos <- rowSums(cbind(BatataF, MandiocaF, FarinhaF), na.rm=T)
banco$carnes_ultra <- rowSums(cbind(HamburguerF, SalsichaF, LinguicaF), na.rm=T)

#Exibe o resultado da análise de componentes principais
sjt.pca(banco[,c("Arroz","massas","paes","frutas","hortalicas","vegetais_conserva","bovina_miudos",
                "suina","carnes_processadas","frios","aves_pescados","leite_derivados","Ovos",
                "feijao","gorduras","doces","salgados","cafe","refrigerantes","sucos",
                "bebidas_alcool","petiscos","leguminosas","cereais","Tuberculos",
                "carnes_ultra")], rotation="varimax", nmbf.fctr=4)

#extraí os componentes principais
#comando do pacote psych
res_pca <- principal(banco[,c("Arroz","massas","paes","frutas","hortalicas","vegetais_conserva","bovina_miudos",
                              "suina","carnes_processadas","frios","aves_pescados","leite_derivados","Ovos",
                              "feijao","gorduras","doces","salgados","cafe","refrigerantes","sucos",
                              "bebidas_alcool","petiscos","leguminosas","cereais","Tuberculos",
                              "carnes_ultra")], rotate="varimax", nfactors=4, method="regression")

banco <- cbind(banco, res_pca$scores)

#Definição de adesão através do escore de consumo individual dividido em tercís
quantile(banco$RC1, probs=c(0, 0.33, 0.66, 1), na.rm=T)
quantile(banco$RC2, probs=c(0, 0.33, 0.66, 1), na.rm=T)
quantile(banco$RC3, probs=c(0, 0.33, 0.66, 1), na.rm=T)
quantile(banco$RC4, probs=c(0, 0.33, 0.66, 1), na.rm=T)

```

```

#considerou-se como ponto de corte o percentil 66%
#0=baixa adesão; 1=alta adesão

#Padrão Ultraprocessados
banco$Ultraprocessados <- ifelse(banco$RC1 < 0.4679494, 0, 1)

#Padrão Saudável
banco$Saudavel <- ifelse(banco$RC2 < 0.3810827, 0, 1)

#Padrão Carnes
banco$Carnes <- ifelse(banco$RC4 < 0.4268895, 0, 1)

#Padrão Tradicional
banco$Tradicional <- ifelse(banco$RC3 < 0.3572539, 0, 1)

# CATEGORIZAÇÃO DAS VARIÁVEIS SÓCIODEMOGRÁFICAS
#Sexo
banco$Sexo <- relevel(banco$p4g14, 2)
tab1(Sexo, graph=F)

#Idade
banco$Idade <- cut(p4idade, breaks=c(10,39,49,59,90), labels=c("< 40","40 a 49","50 a 59", "> 59"))
tab1(Idade, graph=F)

#Escolaridade
banco$Escolaridade <- as.integer(p4g9)
banco$Escolaridade <- ifelse(banco$Escolaridade <= 3, 1,
                             ifelse(banco$Escolaridade >= 4 & banco$Escolaridade <= 5, 2,
                                     ifelse(banco$Escolaridade >= 6 & banco$Escolaridade <= 7, 3, NA)))
banco$Escolaridade <- factor(banco$Escolaridade, levels=c(3,2,1),
                             labels=c("Ensino Superior ou mais", "Ensino Médio completo", "Até ensino Fundamental"))
tab1(banco$Escolaridade, graph=F)

#Cor/raça
banco$Raca <- as.integer(p4g17)
banco$Raca <- ifelse(banco$Raca == 3, 1,
                     ifelse(banco$Raca >= 1 & banco$Raca <= 2, 2,
                             ifelse(banco$Raca >= 4 & banco$Raca <= 5, 2, NA)))
banco$Raca <- factor(banco$Raca, levels=c(1,2), labels=c("Branços", "Não brancos"))
tab1(banco$Raca, graph=F)

#Estado civil
banco$Estado_civil <- as.integer(p4g16)

```

```

banco$Estado_civil <- ifelse(banco$Estado_civil == 4, 1,
                             ifelse(banco$Estado_civil == 1, 2,
                                     ifelse(banco$Estado_civil >= 2 & banco$Estado_civil <= 3, 3, NA)))
banco$Estado_civil <- factor(banco$Estado_civil, levels=c(1,2,3),
                             labels=c("Solteiro", "Casado ou em união", "Separado ou viúvo"))
tab1(banco$Estado_civil, graph=F)

#Renda
banco$Renda <- as.integer(p4g5)
banco$Renda <- ifelse(banco$Renda <= 6, 1, ifelse(banco$Renda >= 7, 2, NA))
banco$Renda <- factor(banco$Renda, levels=c(2,1), labels=c("até 6 salários mínimos", "> 6 salários mínimos"))
tab1(banco$Renda, graph=F)

detach(banco)
attach(banco)

#REGRESSÃO LOGÍSTICA BRUTA - PADRÃO ULTRAPROCESSADO x VARIÁVEIS SOCIODEMOGRÁFICAS
logistic.display(glm(Ultraprocessados ~ Sexo, family=binomial))
logistic.display(glm(Ultraprocessados ~ Idade, family=binomial))
logistic.display(glm(Ultraprocessados ~ Escolaridade, family=binomial))
logistic.display(glm(Ultraprocessados ~ Raca, family=binomial))
logistic.display(glm(Ultraprocessados ~ Estado_civil, family=binomial))
logistic.display(glm(Ultraprocessados ~ Renda, family=binomial))

#REGRESSÃO LOGÍSTICA AJUSTADA - PADRÃO ULTRAPROCESSADO x SEXO, IDADE E RENDA
logistic.display(glm(Ultraprocessados ~ Sexo + Idade + Renda, family=binomial))

#####
#REGRESSÃO LOGÍSTICA BRUTA - PADRÃO SAUDÁVEL x VARIÁVEIS SOCIODEMOGRÁFICAS
logistic.display(glm(Saudavel ~ Sexo, family=binomial))
logistic.display(glm(Saudavel ~ Idade, family=binomial))
logistic.display(glm(Saudavel ~ Escolaridade, family=binomial))
logistic.display(glm(Saudavel ~ Raca, family=binomial))
logistic.display(glm(Saudavel ~ Estado_civil, family=binomial))
logistic.display(glm(Saudavel ~ Renda, family=binomial))

#REGRESSÃO LOGÍSTICA AJUSTADA - PADRÃO SAUDÁVEL x SEXO, IDADE, ESTADO CIVIL, ESCOLARIDADE E RENDA
logistic.display(glm(Saudavel ~ Sexo+Idade+Escolaridade+Estado_civil+Renda, family=binomial))

#####
#REGRESSÃO LOGÍSTICA BRUTA - PADRÃO CARNES x VARIÁVEIS SOCIODEMOGRÁFICAS
logistic.display(glm(Carnes ~ Sexo, family=binomial))
logistic.display(glm(Carnes ~ Idade, family=binomial))

```

```
logistic.display(glm(Carnes ~ Escolaridade, family=binomial))
logistic.display(glm(Carnes ~ Raca, family=binomial))
logistic.display(glm(Carnes ~ Estado_civil, family=binomial))
logistic.display(glm(Carnes ~ Renda, family=binomial))
```

```
#REGRESSÃO LOGÍSTICA AJUSTADA - PADRÃO CARNES x SEXO, IDADE, ESTADO CIVIL E RENDA
logistic.display(glm(Carnes ~ Renda+Sexo+Idade+Estado_civil, family=binomial))
```

```
#####
```

```
#REGRESSÃO LOGÍSTICA BRUTA - PADRÃO TRADICIONAL x VARIÁVEIS SOCIODEMOGRÁFICAS
```

```
logistic.display(glm(Tradicional ~ Sexo, family=binomial))
logistic.display(glm(Tradicional ~ Idade, family=binomial))
logistic.display(glm(Tradicional ~ Escolaridade, family=binomial))
logistic.display(glm(Tradicional ~ Raca, family=binomial))
logistic.display(glm(Tradicional ~ Estado_civil, family=binomial))
logistic.display(glm(Tradicional ~ Renda, family=binomial))
```

```
#REGRESSÃO LOGÍSTICA AJUSTADA - PADRÃO TRADICIONAL x SEXO, IDADE, RAÇA, ESTADO CIVIL, ESCOLARIDADE e RENDA
logistic.display(glm(Tradicional ~ Sexo+Idade+Escolaridade+Raca+Estado_civil+Renda, family=binomial))
```