Referências Bibliográficas

AFIFI, A.A. e CLARK, V. (1984). Computer – Aided Multivariate Analysis. Lifetime Learning Publications. Belm. California.

ANDERSON, T.W. (1984). An Introduction to Multivariate Statistical Analysis. 2ed.New York : John Wiley & Sons.

BARROSO, L.P. (2003).Tópicos de Análise Multivariada. Universidade de São Paulo.

BURT, C. (1941).The Vector of the Mind: An Introduction to Factor Analysis in Psychology. New York: Mc.Millan.

CAZAR, R.A. (2003). An Exercise on Chemometrics for a Quantitative Analysis Course. Madison. **Journal of Chemical Education**.

CHATFIELD, C. e COLLINS, A.J. (1980). Introduction to Multivariate Analysis. Chapman and Hall. New York.

CLIFF, N. e HAMBURGE, C.D. (1967).The Study of Sampling Errors in Factor Analysis by Means of Artificial Experiments. **Psychological Bulletin** 68: 430-45.

COOLEY, W.W. e LOHNES, P.R. (1971). Multivariate Data Analysis. John Wiley. New York.

COSTA, Giovani Glaucio de O. (2003). Busca de Fatores Associados à Prática de Atos Infracionais por Parte de Adolescentes no Estado do Rio de Janeiro: Um Estudo Preliminar, Estudo Orientado, PUC-RIO.

DAVID, A.A.; KUMAR, V. and GEORGE, S. Day. (1984). Marketing Research.

DILLON, W.R. e GOLDSTEIN, M. (1984). Multivariate Analysis: Methods and Applications. New York: John Wiley & Sons.

EFRON, B. (1979). Bootstrap Methods: Another Look at the Jackknife, **The Annals of Statistic**, 7, 1-26.

EFRON, B. (1980). Computer Intensive Methods in Statistics" in Some Recent Advance in Statistic, Ed. J. Tiago de Oliveira e B. Epstein, Academia das Ciências de Lisboa, Lisboa.

EFRON, B. (1982). The Jackknife, the Bootstrap, and other Resampling Methods, CBNS 38, SIAM-NSF.

EVERITT, B.S. (1978). Graphical Techniques for Multivariate Data. Heinemann Educational Books. London.

FERREIRA, D.F. Análise Multivariada. Minas Gerais: Universidade Federal de Lavras.

GNANADESIKAN, R. (1997). Methods for Statistical Data Analysis of Multivariate Observations. John Wiley. New York.

HAIR, J.F. Jr.; ANDERSON, R.E.; TATHAN, R.L. e BLACK, W.C. (2005). Trad. Sant´Anna,Adonai Schlup ; Neto, Anselmo Chaves. Análise Multivariada de Dados. 5. ed. Porto Alegre : Bookman.

HAIR, J.F. Jr.; ANDERSON, R.E.; TATHAN, R.L. e BLACK, W.C. (1998). Multivariate Data Analysis. 5th ed. Upper Saddle River : Prentice Hall.

HARMAN, H.H. (1967). Modern Factor Analysis . 2 ed. Chicago: University of Chicago.

HAWKINS, D.M. Topics in Multivariate Analysis. Cambridge. University Press: Cambridge.

HOGAN, D.J. (2002). Ordenação Multivariada na Ecologia e seu Uso em Ciências Ambientais. São Paulo: Ambiente e Sociedade.

HOMPSON, B. (1984). Canonical Correlation Analysis: Uses and Interpretation. Beverly Hills: Sage Pub.

JOBSON, J. (1992) D. Applied Multivariate Data Analysis. Vol II: Categorical and Multivariate Methods. Springer Verlag. New York.

JOHNSON, D.E. (1998). Applied Multivariate Methods for Data Analysis. Pacific Grove: Duxbury Press..

JOHNSON, R A. and WICHERN, D.W. (1998). Applied Multivariate Statistical Analysis. 4ed . Upper Saddle River: Prentice Hall.

JOHNSON, R.A. and WICHERN, D.W. (1982). Applied Multivariate Statistical Analysis. Prentice Hall, Inc. Englewood Cliffs, New Jersey.

KAISER, H.F. (1958). The Varimax Criterion for Analytic Rotation in Factor Analysis. **Psychometrika**, 23, 187-200.

KAISER, H.F. (1974). A Second-Generation Little Jiffy. **Psychometrika** 35: 401-15

KAUFMAN, L. and ROUSSEEUW, P.J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. New York . Jonh Wile & Sons.

LEBART, L.; MORINEAU, A. and PIRON, M. (1998). Statistique Exploratoire Multidimensionnelle.França.

LONG, J.S. (1983). Confirmation Factor Analysis. Beverly Hills: Sage Pub.

MARDIA, K.V.; KENT, J.T. and BIBBY, J.M. (1989). Multivariate Analysis. London: Academic Press.

MOITA NETO, J.M. e MOITA, G.C. (1998). Uma Introdução à Análise de Dados Multivaridos. São Paulo: Química Nova.

MORRISON, D. (3rd ed.). (1982). Multivariate Statistical Methods. McGraw-Hill. New York.

PEREIRA, A. (1999). Guia Prático de Utilização do SPSS: Análise de Dados para Ciências Sociais e Psicologia. Lisboa: Edições Silabo.

PRADO, P.I.; LEWWINSOHN, T.M.; CARMO, R.L; PROENÇA, I.M.D. (1988). Estimativas Jackknife e Bootstrap para o Enviesamento e Desvio-padrão do índice de Gini. Doc.Trabalho Nº 67, Cemapre, I.S.E , Lisboa.

REIS, E. (1997). Estatística Multivariada Aplicada. Lisboa: Edições Sílabo.

REYMENT, R. e JORESKOG, K.G. (1996). Applied Factor Analysis in the Natural Science. Cambridge. Cambridge University Press.

ROSA, P.T.M. (2000). Modelos de “Credit Scoring”: Regressão Logística, CHAID e REAL . Dissertação de Mestrado. IME – Universidade de São Paulo.

RUMMEL, R.J. (1970) Applied Factor Analysis. Evanston: Northwestern University Press.

SAS Institute Inc. SAS Procedures Guide, Version 8.Cary, N.C.(1999). SAS Institute Inc.

SAPORTA, G. (1996). Probabilités Analyse Des Données Et Statistique. França.

SEBER, G.A.F. (1984). Multivariate Observations. John Wiley. New York.

SIEGEL, S. (1975). Estatística Não-paramétrica para Ciências do Comportamento. São Paulo: McGraw-Hill Editora

SHARMA, S. (1996). Applied Multivariate Techniques. New York: John Wile & Sons.

SPEARMAN, C. (1940). General Intelligence Objectively Determined and Measured. **American Journal of Psychology**, 15: 201-293.

STEWART, D.W. (1981). The applications and misapplications of factor analysis in marketing research. **Journal of Marketing Research**, 18, p.51-62, Feb.

TABACHNICK, B.G. and FIDEL, L.S. (2001). Using Multivariate Statistics . 4th ed. Allyn 7 Bacon , Boston.

THURSTONE, L.L. (1993). Multiple Factor Analysis. Chicago: Univ. Chicago Press.

THURSTONE, L.L. (1993). The Vector of Mind. Chicago: Univ. Chicago Press.

Anexo 1

Os Programas *Bootstrap* e Jackknife em R 2.1.1

O programa abaixo referê-se ao usado com a base de dados “Modo de Vida”.

```
#================================================================================
=========
# PROGRAMA FEITO EM 29/SETEMBRO/2005 - ÚLTIMA MODIFICAÇÃO
08/OUTUBRO/2005
#========================================================
================================
# Usando a ajuda
help("factanal")
# Lendo os dados
# Informe o número de variáveis
m      <-  11
# Número de variáveis
x      <-  matrix(scan("c:/Tese
Gcosta/basepaises.txt"),,m,T)
# Análise Fatorial - guardando os dois fatores
cargas <- factanal(x,factor=2)$loadings[,1:2]
#========================================================
================================
# INÍCIO DO BOOTSTRAP
#========================================================
================================
n    <- length(x[,1])                                 #
Tamanho da amostra original (40)
N    <- length(x[,1])                                 #
Tamanho da re-amostra       (40)
M    <- 1000                                          #
Número de replicações       (1000)
#========================================================
================================
# Função auxiliar para gerar números de 1 até o tamanho
da amostra
rp  <- function(N,n){
y  <- rep(NA,N)
```

```
for(i in 1:N){
  aux <- runif(1)
  lb <- 0
  ub <- 1/n
  for(j in 1:n){
    if(lb<aux && aux <= ub ) y[i] <- j
    lb <- lb + 1/n
    ub <- ub + 1/n
  }
}
return(y)
}
#=========================================================
================================
F1  <- matrix(NA,length(x[1,]),M)
F2  <- matrix(NA,length(x[1,]),M)
# Fazendo ...
for(i in 1:M){
 w          <- rp(N,n)
# Gerando valores da reamostra
 z          <- x[w,]
# Alocando a reamostra
 y          <- factanal(z,factor=2)$loading[,1:2]
# Análise Fatorial
 F1[,i]     <- y[,1]
 F2[,i]     <- y[,2]
}
F1.MEAN <- mean(F1[1,])
F2.MEAN <- mean(F2[1,])
F1.VAR  <- var(F1[1,])
F2.VAR  <- var(F2[1,])
alpha   <- 0.05
q1      <- round(M*alpha/2)
q2      <- M - q1 + 1
F1.LI   <- sort(F1[1,])[q1]
F1.LS   <- sort(F1[1,])[q2]
F2.LI   <- sort(F2[1,])[q1]
F2.LS   <- sort(F2[1,])[q2]
for(i in 2:m){
F1.MEAN <- c(F1.MEAN,mean(F1[i,]))
```

```
F2.MEAN <- c(F2.MEAN,mean(F2[i,]))
F1.VAR  <- c(F1.VAR,var(F1[i,]))
F2.VAR  <- c(F2.VAR,var(F2[i,]))
F1.LI   <- c(F1.LI,sort(F1[i,])[q1])
F1.LS   <- c(F1.LS,sort(F1[i,])[q2])
F2.LI   <- c(F2.LI,sort(F2[i,])[q1])
F2.LS   <- c(F2.LS,sort(F2[i,])[q2])
}
F1.VIES <- cargas[,1] - F1.MEAN
F2.VIES <- cargas[,2] - F2.MEAN
F1.EMQ  <- F1.VAR - F1.VIES^2
F2.EMQ  <- F2.VAR - F2.VIES^2
F1.SISIC<- rep(NA,m)
F2.SISIC<- rep(NA,m)
for(i in 1:m){
  if(F1.LI[i]<= 0 && 0 <= F1.LS[i]){
     F1.SISIC[i] <- "NSIG"
  }
  else{
     F1.SISIC[i] <- "SIG"
  }
  if(F2.LI[i]<= 0 && 0 <= F2.LS[i]){
     F2.SISIC[i] <- "NSIG"
  }
  else{
     F2.SISIC[i] <- "SIG"
  }
}
# Salvando os dados num arquivo - distribuição amostral
options(digits=5)
w  <- cbind(t(F1),t(F2))
write(t(w),file="c:/Tese Gcosta/dist-boot-
summary.txt",ncolumn=(2*m))
# Padronização ...
for(i in 1:m){
  F1[i,] <- (F1[i,] - F1.MEAN[i])/sqrt(F1.VAR[i])  #
Padronização
  F2[i,] <- (F2[i,] - F2.MEAN[i])/sqrt(F2.VAR[i])  #
Padronização
}
```

```
# Salvando os dados num arquivo - distribuição amostral
padronizada
options(digits=5)
w  <- cbind(t(F1),t(F2))
write(t(w),file="c:/Tese Gcosta/dist-boot-padro-
summary.txt",ncolumn=(2*m))
# Calculando G
g  <- cargas/sqrt(cbind(F1.VAR,F2.VAR))
p.valor <- matrix(NA,m,2)
for(j in 1:m){
  cont1 <- 0
  cont2 <- 0
  for(i in 1:M){
    # Fator 1
    if(F1[j,i] >= g[j,1] && g[j,1]>= 0 ) cont1 <- cont1 +
1   # Se g >= 0 ele só fará este
    if(F1[j,i] <= g[j,1] && g[j,1] < 0 ) cont1 <- cont1 +
1   # Se g  < 0 ele só fará este
    # Fator 2
    if(F2[j,i] >= g[j,2] && g[j,2]>= 0 ) cont2 <- cont2 +
1   # Se g >= 0 ele só fará este
    if(F2[j,i] <= g[j,2] && g[j,2] < 0 ) cont2 <- cont2 +
1   # Se g <  0 ele só fará este
  }
  p.valor[j,1] <- cont1/M
  p.valor[j,2] <- cont2/M
}
F1.SIGVP <- rep(NA,m)
F2.SIGVP <- rep(NA,m)
for(i in 1:m){
  if( p.valor[i,1] > 0.05 ){
     F1.SIGVP[i] <- "NSIG"
  }else{
     F1.SIGVP[i] <- "SIG"
  }
  if( p.valor[i,2] > 0.05 ){
     F2.SIGVP[i] <- "NSIG"
  }else{
     F2.SIGVP[i] <- "SIG"
  }
}
p.valor
F1.VP <- p.valor[,1]
F2.VP <- p.valor[,2]
F1.C  <- cargas[,1]
F2.C  <- cargas[,2]
F1.summary <-
data.frame(cbind(F1.C,F1.MEAN,F1.VAR,F1.LI,F1.LS,F1.VIES,
F1.EMQ,F1.SISIC,F1.VP,F1.SIGVP))
F2.summary <-
data.frame(cbind(F2.C,F2.MEAN,F2.VAR,F2.LI,F2.LS,F2.VIES,
F2.EMQ,F2.SISIC,F2.VP,F2.SIGVP))
options(digits=5)
F1.summary        # RESUMO DO FATOR 1
F2.summary        # RESUMO DO FATOR 2
# Salvando os dados num arquivo - resumo estatístico
options(digits=5)
F.summary <-
rbind(cbind(F1.C,F1.MEAN,F1.VAR,F1.LI,F1.LS,F1.VIES,F1.EM
Q,F1.SISIC,F1.VP,F1.SIGVP),

cbind(F2.C,F2.MEAN,F2.VAR,F2.LI,F2.LS,F2.VIES,F2.EMQ,F2.S
ISIC,F2.VP,F2.SIGVP))
write(t(F.summary),file="c:/Tese Gcosta/boot-
summary.txt",ncolumn=10)
#========================================================
================================
# FIM DO BOOTSTRAP
#========================================================
================================
#========================================================
================================
# Para aqui
#========================================================
================================
#========================================================
================================
# INÍCIO DO JACKNIFE
#========================================================
================================
n    <- length(x[,1])  # Tamanho da amostra original (40)
N    <- length(x[,1])  # Tamanho da re-amostra       (40)
M    <- length(x[,1])  # Número de replicações       (40)
#========================================================
================================
F1  <- matrix(NA,length(x[1,]),M)
F2  <- matrix(NA,length(x[1,]),M)
# Fazendo ...
for(i in 1:M){
 if(i==1) w <- seq(2,N,1)
# Gerando valores da reamostra
 if(i==N) w <- seq(1,N-1,1)
# Gerando valores da reamostra
 if(i!=1 && i!=N) w <- c(seq(1,(i-1),1),seq((i+1),N,1))
# Gerando valores da reamostra
 z          <- x[w,]
# Alocando a reamostra
 y          <- factanal(z,factor=2)$loading[,1:2]
# Análise Fatorial
 F1[,i]     <- y[,1]
 F2[,i]     <- y[,2]
}
F1.MEAN <- mean(F1[1,])
F2.MEAN <- mean(F2[1,])
F1.VAR  <- var(F1[1,])
F2.VAR  <- var(F2[1,])
alpha   <- 0.05
q1      <- round(M*alpha/2)
q2      <- M - q1 + 1
F1.LI   <- sort(F1[1,])[q1]
F1.LS   <- sort(F1[1,])[q2]
F2.LI   <- sort(F2[1,])[q1]
F2.LS   <- sort(F2[1,])[q2]
for(i in 2:m){
F1.MEAN <- c(F1.MEAN,mean(F1[i,]))
F2.MEAN <- c(F2.MEAN,mean(F2[i,]))
F1.VAR  <- c(F1.VAR,var(F1[i,]))
F2.VAR  <- c(F2.VAR,var(F2[i,]))
F1.LI   <- c(F1.LI,sort(F1[i,])[q1])
F1.LS   <- c(F1.LS,sort(F1[i,])[q2])
F2.LI   <- c(F2.LI,sort(F2[i,])[q1])
F2.LS   <- c(F2.LS,sort(F2[i,])[q2])
}
F1.VIES <- cargas[,1] - F1.MEAN
F2.VIES <- cargas[,2] - F2.MEAN
F1.EMQ  <- F1.VAR - F1.VIES^2
F2.EMQ  <- F2.VAR - F2.VIES^2
F1.SISIC<- rep(NA,m)
F2.SISIC<- rep(NA,m)
for(i in 1:m){
  if(F1.LI[i]<= 0 && 0 <= F1.LS[i]){
     F1.SISIC[i] <- "NSIG"
  }
  else{
     F1.SISIC[i] <- "SIG"
  }
  if(F2.LI[i]<= 0 && 0 <= F2.LS[i]){
     F2.SISIC[i] <- "NSIG"
  }
  else{
     F2.SISIC[i] <- "SIG"
  }
}
# Salvando os dados num arquivo - distribuição amostral
options(digits=5)
w  <- cbind(t(F1),t(F2))
write(t(w),file="c:/Tese Gcosta/dist-jack-
summary.txt",ncolumn=(2*m))
# Padronização ...
for(i in 1:m){
  F1[i,] <- (F1[i,] - F1.MEAN[i])/sqrt(F1.VAR[i])  #
Padronização
  F2[i,] <- (F2[i,] - F2.MEAN[i])/sqrt(F2.VAR[i])  #
Padronização
}
# Salvando os dados num arquivo - distribuição amostral
padronizada
options(digits=5)
w  <- cbind(t(F1),t(F2))
```

```
write(t(w),file="c:/Tese Gcosta/dist-jack-padro-
summary.txt",ncolumn=(2*m))
# Calculando G
g  <- cargas/sqrt(cbind(F1.VAR,F2.VAR))
p.valor <- matrix(NA,m,2)
for(j in 1:m){
  cont1 <- 0
  cont2 <- 0
  for(i in 1:M){
    # Fator 1
    if(F1[j,i] >= g[j,1] && g[j,1]>= 0 ) cont1 <- cont1 +
1   # Se g >= 0 ele só fará este
    if(F1[j,i] <= g[j,1] && g[j,1] < 0 ) cont1 <- cont1 +
1   # Se g  < 0 ele só fará este
    # Fator 2
    if(F2[j,i] >= g[j,2] && g[j,2]>= 0 ) cont2 <- cont2 +
1   # Se g >= 0 ele só fará este
    if(F2[j,i] <= g[j,2] && g[j,2] < 0 ) cont2 <- cont2 +
1   # Se g <  0 ele só fará este
  }
  p.valor[j,1] <- cont1/M
  p.valor[j,2] <- cont2/M
}
F1.SIGVP <- rep(NA,m)
F2.SIGVP <- rep(NA,m)
for(i in 1:m){
  if( p.valor[i,1] > 0.05 ){
     F1.SIGVP[i] <- "NSIG"
  }else{
     F1.SIGVP[i] <- "SIG"
  }
  if( p.valor[i,2] > 0.05 ){
     F2.SIGVP[i] <- "NSIG"
  }else{
     F2.SIGVP[i] <- "SIG"
  }
}
p.valor
F1.VP <- p.valor[,1]
F2.VP <- p.valor[,2]
```

```
F1.C  <- cargas[,1]
F2.C  <- cargas[,2]
F1.summary <-
data.frame(cbind(F1.C,F1.MEAN,F1.VAR,F1.LI,F1.LS,F1.VIES,
F1.EMQ,F1.SISIC,F1.VP,F1.SIGVP))
F2.summary <-
data.frame(cbind(F2.C,F2.MEAN,F2.VAR,F2.LI,F2.LS,F2.VIES,
F2.EMQ,F2.SISIC,F2.VP,F2.SIGVP))
options(digits=5)
F1.summary        # RESUMO DO FATOR 1
F2.summary        # RESUMO DO FATOR 2
# Salvando os dados num arquivo - resumo estatístico
options(digits=5)
F.summary <-
rbind(cbind(F1.C,F1.MEAN,F1.VAR,F1.LI,F1.LS,F1.VIES,F1.EM
Q,F1.SISIC,F1.VP,F1.SIGVP),

cbind(F2.C,F2.MEAN,F2.VAR,F2.LI,F2.LS,F2.VIES,F2.EMQ,F2.S
ISIC,F2.VP,F2.SIGVP))
write(t(F.summary),file="c:/Tese Gcosta/jack-
summary.txt",ncolumn=10)
#========================================================
================================
# FIM DO JACKNIFE
#========================================================
================================
```

Anexo 2

O Programa Bootstrap em SAS V.8

O programa abaixo referê-se ao usado com a base de dados “Creme Dental”.

```
*libname in 'C:\Tese Giovani';
libname in2 'C:\Tese Giovani\Terceira Fase';
options nodate ls=max ps=max;

PROC IMPORT OUT= IN2.basecremedental
            DATAFILE= "C:\Tese Giovani\Terceira
Fase\basecremedental.xls"
            DBMS=EXCEL2000 REPLACE;
     GETNAMES=YES;
RUN;

data base(drop=F7);
 set in2.basecremedental;
run;

%macro sel(i,j);
data base&i;
 set base;
  if _n_=&i then delete;
run;

proc factor data=base&i
            method=principal
             rotate=Varimax
            outstat=perfil&i
            nfactors=2 noprint;
 var V1-V6;
run;

data perfil_f1_&i(keep=i f1_v01-f1_v06)
     perfil_f2_&i(keep=i f2_v01-f2_v06);
 set perfil&i;
 if _type_='UNROTATE' and _name_ in ('Factor1'
'Factor2');

 if _name_='Factor1' then do;
    f1_v01= v1 ;   f1_v02= v2 ;   f1_v03= v3 ;   f1_v04=
v4 ;   f1_v05= v5 ;   f1_v06= v6 ;
   end;
 else if _name_='Factor2' then do;
    f2_v01= v1 ;   f2_v02= v2 ;   f2_v03= v3 ;   f2_v04=
v4 ;   f2_v05= v5 ;   f2_v06= v6 ;
 end;
    i=1;

     if _name_='Factor1' then output perfil_f1_&i;
else if _name_='Factor2' then output perfil_f2_&i;
run;

data amostra&i(drop=i);
 merge perfil_f1_&i perfil_f2_&i;
  by i;
  amostra=&i;
run;

proc delete data=perfil&i;
proc delete data=perfil_f1_&i;
proc delete data=perfil_f2_&i;
proc delete data=base&i;
run;
%mend sel;
%sel(   1    ,    87575    ); %sel  (   2    ,    42631    );
%sel(   3    ,    98398    ); %sel  (   4    ,    82401    );
%sel(   5    ,    75217    ); %sel  (   6    ,    23436    );
%sel(   7    ,    78850    ); %sel  (   8    ,    99497    );
%sel(   9    ,    75267    ); %sel  (   10   ,    12243    );
%sel(   11   ,    55216    ); %sel  (   12   ,    90000    );
%sel(   13   ,    10855    ); %sel  (   14   ,    20001    );
%sel(   15   ,    44804    ); %sel  (   16   ,    31726    );
%sel(   17   ,    93948    ); %sel  (   18   ,    74881    );
%sel(   19   ,    20262    ); %sel  (   20   ,    13439    );
%sel(   21   ,    3817); %sel  (   22   ,    6551);
```

```
%sel(   23   ,    54403    ); %sel  (   24   ,    11361    );
%sel(   25   ,    97085    ); %sel  (   26   ,    83919    );
%sel(   27   ,    85594    ); %sel  (   28   ,    35212    );
%sel(   29   ,    32651    );%sel   (   30   ,    84167    );
%sel(   31   ,    23953    ); %sel  (   32   ,    56974    );
%sel(   33   ,    59788    ); %sel  (   34   ,    91566    );
%sel(   35   ,    84992    ); %sel  (   36   ,    20942    );
%sel(   37   ,    84190    ); %sel  (   38   ,    50512    );
%sel(   39   ,    98086    ); %sel  (   40   ,    23271    );
%sel(   41   ,    59386    ); %sel  (   42   ,    49980    );
%sel(   43   ,    71145    ); %sel  (   44   ,    29931    );
%sel(   45   ,    64912    ); %sel  (   46   ,    12436    );
%sel(   47   ,    49793    ); %sel  (   48   ,    33565    );
%sel(   49   ,    57161    ); %sel  (   50   ,    94119    );
%sel(   51   ,    63066    ); %sel  (   52   ,    88110    );
%sel(   53   ,    44405    ); %sel  (   54   ,    13065    );
%sel(   55   ,    25214    ); %sel  (   56   ,    47196    );
%sel(   57   ,    29784    ); %sel  (   58   ,    22568    );
%sel(   59   ,    60846    ); %sel  (   60   ,    83724    );
%sel(   61   ,    72102    ); %sel  (   62   ,    27420    );
%sel(   63   ,    5148); %sel  (   64   ,    77389    );
%sel(   65   ,    48909    ); %sel  (   66   ,    52879    );
%sel(   67   ,    39849    ); %sel  (   68   ,    72711    );
%sel(   69   ,    14754    ); %sel  (   70   ,    16441    );
%sel(   71   ,    81171    ); %sel  (   72   ,    91869    );
%sel(   73   ,    64291    ); %sel  (   74   ,    16099    );
%sel(   75   ,    28189    ); %sel  (   76   ,    58487    );
%sel(   77   ,    16035    ); %sel  (   78   ,    93280    );
%sel(   79   ,    23064    ); %sel  (   80   ,    28573    );
%sel(   81   ,    32756    ); %sel  (   82   ,    12601    );
%sel(   83   ,    71664    ); %sel  (   84   ,    31319    );
%sel(   85   ,    3285); %sel  (   86   ,    73965    );
%sel(   87   ,    30113    ); %sel  (   88   ,    56699    );
%sel(   89   ,    59394    ); %sel  (   90   ,    24559    );
%sel(   91   ,    8943); %sel  (   92   ,    49173    );
%sel(   93   ,    43540    ); %sel  (   94   ,    22720    );
%sel(   95   ,    49001    ); %sel  (   96   ,    51072    );
%sel(   97   ,    91797    ); %sel  (   98   ,    54853    );
%sel(   99   ,    31344    ); %sel  (   100  ,    44768    );
%sel(   101  ,    80496    ); %sel  (   102  ,    71547    );
```

```
%sel(    103 ,     83316     ); %sel  (    104 ,     64796     );
%sel(    105 ,     79490     ); %sel  (    106 ,     88229     );
%sel(    107 ,     15576     ); %sel  (    108 ,     83319     );
%sel(    109 ,     90206     ); %sel  (    110 ,     25221     );
%sel(    111 ,     11345     ); %sel  (    112 ,     35133     );
%sel(    113 ,     57730     ); %sel  (    114 ,     86329     );
%sel(    115 ,     57344     ); %sel  (    116 ,     87914     );
%sel(    117 ,     21095     ); %sel  (    118 ,     91279     );
%sel(    119 ,     69902     ); %sel  (    120 ,     21540     );
%sel(    121 ,     63059     ); %sel  (    122 ,     22079     );
%sel(    123 ,     51355     ); %sel  (    124 ,     75010     );
%sel(    125 ,     85215     ); %sel  (    126 ,     10590     );
%sel(    127 ,     18821     ); %sel  (    128 ,     60801     );
%sel(    129 ,     45243     ); %sel  (    130 ,     50163     );
%sel(    131 ,     71707     ); %sel  (    132 ,     46959     );
%sel(    133 ,     99526     ); %sel  (    134 ,     60669     );
%sel(    135 ,     43433     ); %sel  (    136 ,     69226     );
%sel(    137 ,     83372     ); %sel  (    138 ,     15105     );
%sel(    139 ,     66160     ); %sel  (    140 ,     68847     );
%sel(    141 ,     61005     ); %sel  (    142 ,     45044     );
%sel(    143 ,     92248     ); %sel  (    144 ,     86235     );
%sel(    145 ,     28617     ); %sel  (    146 ,     68294     );
%sel(    147 ,     35647     ); %sel  (    148 ,     2813);
%sel(    149 ,     39138     ); %sel  (    150 ,     93455     );
%sel(    151 ,     40786     ); %sel  (    152 ,     73002     );
%sel(    153 ,     71325     ); %sel  (    154 ,     97954     );
%sel(    155 ,     20338     ); %sel  (    156 ,     57102     );
%sel(    157 ,     10793     ); %sel  (    158 ,     12959     );
%sel(    159 ,     86666     ); %sel  (    160 ,     61205     );
%sel(    161 ,     13028     ); %sel  (    162 ,     63430     );
%sel(    163 ,     59274     ); %sel  (    164 ,     44090     );
%sel(    165 ,     28600     ); %sel  (    166 ,     95663     );
%sel(    167 ,     22556     ); %sel  (    168 ,     40501     );
%sel(    169 ,     72080     ); %sel  (    170 ,     11798     );
%sel(    171 ,     83621     ); %sel  (    172 ,     62901     );
%sel(    173 ,     70967     ); %sel  (    174 ,     80505     );
%sel(    175 ,     56130     ); %sel  (    176 ,     66572     );
%sel(    177 ,     20738     ); %sel  (    178 ,     83718     );
%sel(    179 ,     14546     ); %sel  (    180 ,     75444     );
%sel(    181 ,     48545     ); %sel  (    182 ,     74441     );
```

```
%sel(   183 ,   2142); %sel (   184 ,   51781   );
%sel(   185 ,   54372   ); %sel (   186 ,   6390);
%sel(   187 ,   73222   ); %sel (   188 ,   28451   );
%sel(   189 ,   31665   ); %sel (   190 ,   93921   );
%sel(   191 ,   13335   ); %sel (   192 ,   75270   );
%sel(   193 ,   39017   ); %sel (   194 ,   97305   );
%sel(   195 ,   46917   ); %sel (   196 ,   86436   );
%sel(   197 ,   4297); %sel (   198 ,   13145   );
%sel(   199 ,   11559   ); %sel (   200 ,   40844   );

data in2.boot;
 set
amostra1 amostra101
amostra2 amostra102
amostra3 amostra103
amostra4 amostra104
amostra5 amostra105
amostra6 amostra106
amostra7 amostra107
amostra8 amostra108
amostra9 amostra109
amostra10    amostra110
amostra11    amostra111
amostra12    amostra112
amostra13    amostra113
amostra14    amostra114
amostra15    amostra115
amostra16    amostra116
amostra17    amostra117
amostra18    amostra118
amostra19    amostra119
amostra20    amostra120
amostra21    amostra121
amostra22    amostra122
amostra23    amostra123
amostra24    amostra124
amostra25    amostra125
amostra26    amostra126
amostra27    amostra127
amostra28    amostra128
```

```
amostra29   amostra129
amostra30   amostra130
amostra31   amostra131
amostra32   amostra132
amostra33   amostra133
amostra34   amostra134
amostra35   amostra135
amostra36   amostra136
amostra37   amostra137
amostra38   amostra138
amostra39   amostra139
amostra40   amostra140
amostra41   amostra141
amostra42   amostra142
amostra43   amostra143
amostra44   amostra144
amostra45   amostra145
amostra46   amostra146
amostra47   amostra147
amostra48   amostra148
amostra49   amostra149
amostra50   amostra150
amostra51   amostra151
amostra52   amostra152
amostra53   amostra153
amostra54   amostra154
amostra55   amostra155
amostra56   amostra156
amostra57   amostra157
amostra58   amostra158
amostra59   amostra159
amostra60   amostra160
amostra61   amostra161
amostra62   amostra162
amostra63   amostra163
amostra64   amostra164
amostra65   amostra165
amostra66   amostra166
amostra67   amostra167
amostra68   amostra168
```

```
amostra69    amostra169
amostra70    amostra170
amostra71    amostra171
amostra72    amostra172
amostra73    amostra173
amostra74    amostra174
amostra75    amostra175
amostra76    amostra176
amostra77    amostra177
amostra78    amostra178
amostra79    amostra179
amostra80    amostra180
amostra81    amostra181
amostra82    amostra182
amostra83    amostra183
amostra84    amostra184
amostra85    amostra185
amostra86    amostra186
amostra87    amostra187
amostra88    amostra188
amostra89    amostra189
amostra90    amostra190
amostra91    amostra191
amostra92    amostra192
amostra93    amostra193
amostra94    amostra194
amostra95    amostra195
amostra96    amostra196
amostra97    amostra197
amostra98    amostra198
amostra99    amostra199
amostra100   amostra200;
run;

proc delete data=amostra1;
proc delete data=amostra2;
proc delete data=amostra3;
proc delete data=amostra4;
proc delete data=amostra5;
proc delete data=amostra6;
proc delete data=amostra7;
proc delete data=amostra8;
proc delete data=amostra9;
proc delete data=amostra10;
proc delete data=amostra11;
proc delete data=amostra12;
proc delete data=amostra13;
proc delete data=amostra14;
proc delete data=amostra15;
proc delete data=amostra16;
proc delete data=amostra17;
proc delete data=amostra18;
proc delete data=amostra19;
proc delete data=amostra20;
proc delete data=amostra21;
proc delete data=amostra22;
proc delete data=amostra23;
proc delete data=amostra24;
proc delete data=amostra25;
proc delete data=amostra26;
proc delete data=amostra27;
proc delete data=amostra28;
proc delete data=amostra29;
proc delete data=amostra30;
proc delete data=amostra31;
proc delete data=amostra32;
proc delete data=amostra33;
proc delete data=amostra34;
proc delete data=amostra35;
proc delete data=amostra36;
proc delete data=amostra37;
proc delete data=amostra38;
proc delete data=amostra39;
proc delete data=amostra40;
proc delete data=amostra41;
proc delete data=amostra42;
proc delete data=amostra43;
proc delete data=amostra44;
proc delete data=amostra45;
proc delete data=amostra46;
proc delete data=amostra47;
proc delete data=amostra48;
proc delete data=amostra49;
proc delete data=amostra50;
proc delete data=amostra51;
proc delete data=amostra52;
proc delete data=amostra53;
proc delete data=amostra54;
proc delete data=amostra55;
proc delete data=amostra56;
proc delete data=amostra57;
proc delete data=amostra58;
proc delete data=amostra59;
proc delete data=amostra60;
proc delete data=amostra61;
proc delete data=amostra62;
proc delete data=amostra63;
proc delete data=amostra64;
proc delete data=amostra65;
proc delete data=amostra66;
proc delete data=amostra67;
proc delete data=amostra68;
proc delete data=amostra69;
proc delete data=amostra70;
proc delete data=amostra71;
proc delete data=amostra72;
proc delete data=amostra73;
proc delete data=amostra74;
proc delete data=amostra75;
proc delete data=amostra76;
proc delete data=amostra77;
proc delete data=amostra78;
proc delete data=amostra79;
proc delete data=amostra80;
proc delete data=amostra81;
proc delete data=amostra82;
proc delete data=amostra83;
proc delete data=amostra84;
proc delete data=amostra85;
proc delete data=amostra86;
proc delete data=amostra87;
proc delete data=amostra88;
proc delete data=amostra89;
proc delete data=amostra90;
proc delete data=amostra91;
proc delete data=amostra92;
proc delete data=amostra93;
proc delete data=amostra94;
proc delete data=amostra95;
proc delete data=amostra96;
proc delete data=amostra97;
proc delete data=amostra98;
proc delete data=amostra99;
proc delete data=amostra100;
proc delete data=amostra101;
proc delete data=amostra102;
proc delete data=amostra103;
proc delete data=amostra104;
proc delete data=amostra105;
proc delete data=amostra106;
proc delete data=amostra107;
proc delete data=amostra108;
proc delete data=amostra109;
proc delete data=amostra110;
proc delete data=amostra111;
proc delete data=amostra112;
proc delete data=amostra113;
proc delete data=amostra114;
proc delete data=amostra115;
proc delete data=amostra116;
proc delete data=amostra117;
proc delete data=amostra118;
proc delete data=amostra119;
proc delete data=amostra120;
proc delete data=amostra121;
proc delete data=amostra122;
proc delete data=amostra123;
proc delete data=amostra124;
proc delete data=amostra125;
proc delete data=amostra126;
proc delete data=amostra127;
proc delete data=amostra128;
proc delete data=amostra129;
proc delete data=amostra130;
proc delete data=amostra131;
proc delete data=amostra132;
proc delete data=amostra133;
proc delete data=amostra134;
proc delete data=amostra135;
proc delete data=amostra136;
proc delete data=amostra137;
proc delete data=amostra138;
proc delete data=amostra139;
proc delete data=amostra140;
proc delete data=amostra141;
proc delete data=amostra142;
proc delete data=amostra143;
proc delete data=amostra144;
proc delete data=amostra145;
proc delete data=amostra146;
proc delete data=amostra147;
proc delete data=amostra148;
proc delete data=amostra149;
proc delete data=amostra150;
proc delete data=amostra151;
proc delete data=amostra152;
proc delete data=amostra153;
proc delete data=amostra154;
proc delete data=amostra155;
proc delete data=amostra156;
proc delete data=amostra157;
proc delete data=amostra158;
proc delete data=amostra159;
proc delete data=amostra160;
proc delete data=amostra161;
proc delete data=amostra162;
proc delete data=amostra163;
proc delete data=amostra164;
proc delete data=amostra165;
proc delete data=amostra166;
proc delete data=amostra167;
proc delete data=amostra168;
proc delete data=amostra169;
proc delete data=amostra170;
proc delete data=amostra171;
proc delete data=amostra172;
proc delete data=amostra173;
proc delete data=amostra174;
proc delete data=amostra175;
proc delete data=amostra176;
proc delete data=amostra177;
proc delete data=amostra178;
proc delete data=amostra179;
proc delete data=amostra180;
proc delete data=amostra181;
proc delete data=amostra182;
proc delete data=amostra183;
proc delete data=amostra184;
proc delete data=amostra185;
proc delete data=amostra186;
proc delete data=amostra187;
proc delete data=amostra188;
proc delete data=amostra189;
proc delete data=amostra190;
proc delete data=amostra191;
proc delete data=amostra192;
proc delete data=amostra193;
proc delete data=amostra194;
proc delete data=amostra195;
proc delete data=amostra196;
proc delete data=amostra197;
proc delete data=amostra198;
proc delete data=amostra199;
proc delete data=amostra200;

proc export data= in2.boot
            outfile= "boot"
            dbms=excel2000 replace;
run;
```

```
proc sort data=in2.boot;
 by amostra;
run;

proc means data=in2.boot mean var max noprint;
 var amostra
 f1_v01  f1_v02  f1_v03  f1_v04  f1_v05
 f1_v06
 f2_v01  f2_v02  f2_v03  f2_v04  f2_v05
 f2_v06
;
  output out=saida(drop=nada1-nada2)
         mean=nada1
 m_f1_v1  m_f1_v2  m_f1_v3  m_f1_v4  m_f1_v5
 m_f1_v6
 m_f2_v1  m_f2_v2  m_f2_v3  m_f2_v4  m_f2_v5
 m_f2_v6
         var =nada2
 v_f1_v1  v_f1_v2  v_f1_v3  v_f1_v4  v_f1_v5
 v_f1_v6
 v_f2_v1  v_f2_v2  v_f2_v3  v_f2_v4  v_f2_v5
 v_f2_v6
         max =B;
 title'Media, Variancia e Tamanho da Amostra - Topicos 2
e 4';
run;

%macro alfa(alfa);
data saidaK(keep=LI LS _type_);
 set saida;
  LI=ROUND(B*&alfa/2); *limite inferior do intervalo
Boostrap*;
  LS=ROUND((B-LI)+1);  *limite superior do intervalo
Boostrap*;
run;
%mend alfa;
%alfa(0.05); *confiança*;

%macro valor_limite(k,j);
proc sort data=in2.boot;
```

```
 by f&k._&j;

data boot&k(keep=n _type_ f&k._&j);
 set in2.boot;
  _type_=0;
   n=_n_;
run;

data limite;
 merge saidaK boot&k;
  by _type_;
   if LI=n then LI_Boot_F&k._&j=f&k._&j;
      else      LI_Boot_F&k._&j=.;
   if LS=n then LS_Boot_F&k._&j=f&k._&j;
      else      LS_Boot_F&k._&j=.;
run;

proc means data=limite max noprint;
var LI_Boot_F&k._&j LS_Boot_F&k._&j;
 output out=limites_f&k._&j
        max=LI_Boot LS_Boot;
run;
proc delete data=limite;
proc delete data=boot&k;
run;
%mend valor_limite;
%valor_limite(1,v01 );
%valor_limite(1,v02 );
%valor_limite(1,v03 );
%valor_limite(1,v04 );
%valor_limite(1,v05 );
%valor_limite(1,v06 );
%valor_limite(2,v01 );
%valor_limite(2,v02 );
%valor_limite(2,v03 );
%valor_limite(2,v04 );
%valor_limite(2,v05 );
%valor_limite(2,v06 );

data limites;
```

```
 set
 limites_f1_v01  limites_f1_v02  limites_f1_v03
limites_f1_v04  limites_f1_v05
 limites_f1_v06
 limites_f2_v01  limites_f2_v02  limites_f2_v03
limites_f2_v04  limites_f2_v05
 limites_f2_v06;

     if _n_=1  then _name_='f1_v01';
else if _n_=2  then _name_='f1_v02';
else if _n_=3  then _name_='f1_v03';
else if _n_=4  then _name_='f1_v04';
else if _n_=5  then _name_='f1_v05';
else if _n_=6  then _name_='f1_v06';
else if _n_=7  then _name_='f2_v01';
else if _n_=8  then _name_='f2_v02';
else if _n_=9  then _name_='f2_v03';
else if _n_=10 then _name_='f2_v04';
else if _n_=11 then _name_='f2_v05';
else if _n_=12 then _name_='f2_v06';

run;

proc delete data=limites_f1_v01;
proc delete data=limites_f1_v02;
proc delete data=limites_f1_v03;
proc delete data=limites_f1_v04;
proc delete data=limites_f1_v05;
proc delete data=limites_f1_v06;
proc delete data= limites_f2_v01;
proc delete data=limites_f2_v02;
proc delete data=limites_f2_v03;
proc delete data=limites_f2_v04;
proc delete data=limites_f2_v05;
proc delete data= limites_f2_v06;
run;

proc factor data=base
            method=principal
             rotate=Varimax
            outstat=perfil
            nfactors=2 noprint;
 var V1-V6;
run;

data perfil_f1(keep=i f1_v01-f1_v06)
     perfil_f2(keep=i f2_v01-f2_v06);
 set perfil;
 if _type_='UNROTATE' and _name_ in ('Factor1'
'Factor2');

if _name_='Factor1' then do;
    f1_v01= v1 ;   f1_v02= v2 ;   f1_v03= v3 ;   f1_v04=
v4 ;   f1_v05= v5 ;
    f1_v06= v6 ;
   end;
 else if _name_='Factor2' then do;
    f2_v01= v1 ;   f2_v02= v2 ;   f2_v03= v3 ;   f2_v04=
v4 ;   f2_v05= v5 ;
    f2_v06= v6 ;
 end;
    i=1;

     if _name_='Factor1' then output perfil_f1;
else if _name_='Factor2' then output perfil_f2;
run;

data Total(drop=i);
 merge perfil_f1 perfil_f2;
  by i;
run;

proc delete data=perfil;
proc delete data=perfil_f1;
proc delete data=perfil_f2;
run;

proc transpose data=total out=fator;
 var f1_v01-f1_v06 f2_v01-f2_v06;
run;
```

```
data fator;
 set fator;
  rename col1=cargas;
run;

proc transpose data=saida out=medias;
 var m_f1_v1-m_f1_v6 m_f2_v1-m_f2_v6;
 id _type_;
run;

proc transpose data=saida out=variancias;
 var v_f1_v1-v_f1_v6 v_f2_v1-v_f2_v6;
 id _type_;
run;

data medias;
 set medias;
  rename _0=media;

  if _name_='m_f1_v1' then _name_='f1_v01';
  if _name_='m_f1_v2' then _name_='f1_v02';
  if _name_='m_f1_v3' then _name_='f1_v03';
  if _name_='m_f1_v4' then _name_='f1_v04';
  if _name_='m_f1_v5' then _name_='f1_v05';
  if _name_='m_f1_v6' then _name_='f1_v06';
 if _name_='m_f2_v1 ' then _name_='f2_v01';
  if _name_='m_f2_v2 ' then _name_='f2_v02';
  if _name_='m_f2_v3 ' then _name_='f2_v03';
  if _name_='m_f2_v4 ' then _name_='f2_v04';
  if _name_='m_f2_v5 ' then _name_='f2_v05';
  if _name_='m_f2_v6 ' then _name_='f2_v06';
run;

data variancias;
 set variancias;
  rename _0=variancia;
  if _name_='v_f1_v1' then _name_='f1_v01';
  if _name_='v_f1_v2' then _name_='f1_v02';
  if _name_='v_f1_v3' then _name_='f1_v03';
  if _name_='v_f1_v4' then _name_='f1_v04';
  if _name_='v_f1_v5' then _name_='f1_v05';
  if _name_='v_f1_v6' then _name_='f1_v06';
  if _name_='v_f2_v1 ' then _name_='f2_v01';
  if _name_='v_f2_v2 ' then _name_='f2_v02';
  if _name_='v_f2_v3 ' then _name_='f2_v03';
  if _name_='v_f2_v4 ' then _name_='f2_v04';
  if _name_='v_f2_v5 ' then _name_='f2_v05';
  if _name_='v_f2_v6 ' then _name_='f2_v06';
run;
proc sort data=limites;
 by _name_;
proc sort data=fator;
 by _name_;
proc sort data=medias;
 by _name_;
proc sort data=variancias;
 by _name_;

data total_5_6(keep=fator_variavel cargas media variancia
                    LI_Boot LS_Boot sig_ic vies
erro_medio g);
 merge fator medias variancias limites;
  by _name_;
   rename _name_=fator_variavel;
   vies=cargas-media;
   erro_medio=variancia+vies**2;
   zero=0;
  if LI_Boot<=zero<=LS_Boot then sig_ic='Nao_Signif';
     else                        sig_ic='Signif    ';
  g=cargas/(variancia**(1/2));

run;

proc transpose data=total_5_6
out=g_total_5_6(drop=_name_);
 var g;
 id fator_variavel;
run;
```

```
data g_total_5_6;
 set g_total_5_6;
 _type_=0;
  rename f1_v01=g_f1_v01;
  rename f1_v02=g_f1_v02;
  rename f1_v03=g_f1_v03;
  rename f1_v04=g_f1_v04;
  rename f1_v05=g_f1_v05;
  rename f1_v06=g_f1_v06;
  rename f2_v01=g_f2_v01;
  rename f2_v02=g_f2_v02;
  rename f2_v03=g_f2_v03;
  rename f2_v04=g_f2_v04;
  rename f2_v05=g_f2_v05;
  rename f2_v06=g_f2_v06;
run;

data boot;
 set in2.boot;
  _type_=0;
run;

data boot_norm(keep=norm_f1_v01-norm_f1_v06
                    norm_f2_v01-norm_f2_v06
                    g_f1_v01-g_f1_v06
                    g_f2_v01-g_f2_v06);
 merge boot saida g_total_5_6;
  by _type_;

 norm_f1_v01=(f1_v01-m_f1_v1)/(v_f1_v1**(1/2));
 norm_f1_v02=(f1_v02-m_f1_v2)/(v_f1_v2**(1/2));
 norm_f1_v03=(f1_v03-m_f1_v3)/(v_f1_v3**(1/2));
 norm_f1_v04=(f1_v04-m_f1_v4)/(v_f1_v4**(1/2));
 norm_f1_v05=(f1_v05-m_f1_v5)/(v_f1_v5**(1/2));
 norm_f1_v06=(f1_v06-m_f1_v6)/(v_f1_v6**(1/2));
 norm_f2_v01=(f2_v01-m_f2_v1)/(v_f2_v1**(1/2));
 norm_f2_v02=(f2_v02-m_f2_v2)/(v_f2_v2**(1/2));
 norm_f2_v03=(f2_v03-m_f2_v3)/(v_f2_v3**(1/2));
 norm_f2_v04=(f2_v04-m_f2_v4)/(v_f2_v4**(1/2));
 norm_f2_v05=(f2_v05-m_f2_v5)/(v_f2_v5**(1/2));
 norm_f2_v06=(f2_v06-m_f2_v6)/(v_f2_v6**(1/2));

run;

%macro pvalor(p);

data s_&p(keep=s_&p);
 set boot_norm;
 if (g_&p>0 and norm_&p>g_&p) or
    (g_&p<0 and norm_&p<g_&p) then s_&p=1;
  else                             s_&p=0;
run;

proc means data=s_&p sum noprint;
 var s_&p;
  output out=saida_&p
         sum=;
run;

proc delete data=s_&p;

data saida_&p(keep=fator_variavel p_valor);
 set saida_&p;
  fator_variavel="&p";
  p_valor=s_&p/_freq_;
run;

%mend pvalor;
%pvalor(f1_v01);
%pvalor(f1_v02);
%pvalor(f1_v03);
%pvalor(f1_v04);
%pvalor(f1_v05);
%pvalor(f1_v06);
%pvalor(f2_v01);
%pvalor(f2_v02);
%pvalor(f2_v03);
%pvalor(f2_v04);
%pvalor(f2_v05);
%pvalor(f2_v06);
```

```
data p_valor;
 set saida_f1_v01 saida_f1_v02 saida_f1_v03 saida_f1_v04
saida_f1_v05
     saida_f1_v06 saida_f2_v01 saida_f2_v02 saida_f2_v03
saida_f2_v04
     saida_f2_v05 saida_f2_v06;
  if p_valor>0.05  then sig_vp='Nao_Signif';
     else               sig_vp='Signif    ';

run;

proc delete data=saida_f1_v01;
proc delete data=saida_f1_v02;
proc delete data=saida_f1_v03;
proc delete data=saida_f1_v04;
proc delete data=saida_f1_v05;
proc delete data=saida_f1_v06;
proc delete data=saida_f2_v01;
proc delete data=saida_f2_v02;
proc delete data=saida_f2_v03;
proc delete data=saida_f2_v04;
proc delete data=saida_f2_v05;
proc delete data=saida_f2_v06;
run;

data total_5_6(drop=g);
 merge total_5_6 p_valor;
  by fator_variavel;
run;

proc print data=total_5_6;
 title'Analise Fatorial - Bootstrap';
run;
```

Anexo 3

O Programa Jackknife em SAS V.8

O programa abaixo referê-se ao usado com a base de dados “Creme Dental”.

```
libname in2 'C:\Tese Giovani\Terceira Fase';
options nodate ls=max ps=max;
proc import out= in2.basecremedental
            datafile=
"C:\Tese Giovani\Terceira Fase\basecremedental.xls"
            dbms=excel2000 replace;
     getnames=yes;
run;

data base;
 set in2.basecremedental;
run;
%macro sel(i);
data base&i;
 set base;
  if _n_=&i then delete;
run;

proc factor data=base&i
            method=principal
             rotate=Varimax
            outstat=perfil&i
            nfactors=2 noprint;
 var V1-V6;
run;

data perfil_f1_&i(keep=i f1_v01-f1_v06)
     perfil_f2_&i(keep=i f2_v01-f2_v06);
 set perfil&i;
 if _type_='UNROTATE' and _name_ in ('Factor1'
'Factor2');

 if _name_='Factor1' then do;
```

```
    f1_v01= v1 ;   f1_v02= v2 ;   f1_v03= v3 ;   f1_v04=
v4 ;   f1_v05= v5 ;   f1_v06= v6 ;
   end;
 else if _name_='Factor2' then do;
    f2_v01= v1 ;   f2_v02= v2 ;   f2_v03= v3 ;   f2_v04=
v4 ;   f2_v05= v5 ;   f2_v06= v6 ;
 end;
    i=1;

     if _name_='Factor1' then output perfil_f1_&i;
else if _name_='Factor2' then output perfil_f2_&i;
run;

data amostra&i(drop=i);
 merge perfil_f1_&i perfil_f2_&i;
  by i;
  amostra=&i;
run;

proc delete data=perfil&i;
proc delete data=perfil_f1_&i;
proc delete data=perfil_f2_&i;
proc delete data=base&i;
run;

%mend sel;
%sel(1  ); %sel (2  );
%sel(3  ); %sel (4  );
%sel(5  ); %sel (6  );
%sel(7  ); %sel (8  );
%sel(9  ); %sel (10 );
%sel(11 ); %sel (12 );
%sel(13 ); %sel (14 );
%sel(15 ); %sel (16 );
%sel(17 ); %sel (18 );
%sel(19 ); %sel (20 );
%sel(21 ); %sel (22 );
%sel(23 ); %sel (24 );
%sel(25 ); %sel (26 );
%sel(27 ); %sel (28 );
```

```
%sel(29 );%sel  (30 );
%sel(31 ); %sel (32 );
%sel(33 ); %sel (34 );
%sel(35 ); %sel (36 );
%sel(37 ); %sel (38 );
%sel(39 ); %sel (40 );

data in2.jack;
 set
amostra1
amostra2
amostra3
amostra4
amostra5
amostra6
amostra7
amostra8
amostra9
amostra10
amostra11
amostra12
amostra13
amostra14
amostra15
amostra16
amostra17
amostra18
amostra19
amostra20
amostra21
amostra22
amostra23
amostra24
amostra25
amostra26
amostra27
amostra28
amostra29
amostra30
amostra31
```

```
amostra32
amostra33
amostra34
amostra35
amostra36
amostra37
amostra38
amostra39
amostra40;
run;

proc delete data=amostra1;
proc delete data=amostra2;
proc delete data=amostra3;
proc delete data=amostra4;
proc delete data=amostra5;
proc delete data=amostra6;
proc delete data=amostra7;
proc delete data=amostra8;
proc delete data=amostra9;
proc delete data=amostra10;
proc delete data=amostra11;
proc delete data=amostra12;
proc delete data=amostra13;
proc delete data=amostra14;
proc delete data=amostra15;
proc delete data=amostra16;
proc delete data=amostra17;
proc delete data=amostra18;
proc delete data=amostra19;
proc delete data=amostra20;
proc delete data=amostra21;
proc delete data=amostra22;
proc delete data=amostra23;
proc delete data=amostra24;
proc delete data=amostra25;
proc delete data=amostra26;
proc delete data=amostra27;
proc delete data=amostra28;
proc delete data=amostra29;
proc delete data=amostra30;
proc delete data=amostra31;
proc delete data=amostra32;
proc delete data=amostra33;
proc delete data=amostra34;
proc delete data=amostra35;
proc delete data=amostra36;
proc delete data=amostra37;
proc delete data=amostra38;
proc delete data=amostra39;
proc delete data=amostra40;

proc export data= in2.jack
            outfile= "jack"
            dbms=excel2000 replace;
run;

proc sort data=in2.jack;
 by amostra;
run;

proc means data=in2.jack mean var max noprint;
 var amostra
 f1_v01  f1_v02  f1_v03  f1_v04  f1_v05  f1_v06
 f2_v01  f2_v02  f2_v03  f2_v04  f2_v05  f2_v06;
  output out=saida(drop=nada1-nada2)
         mean=nada1
 m_f1_v1  m_f1_v2  m_f1_v3  m_f1_v4  m_f1_v5  m_f1_v6
 m_f2_v1  m_f2_v2  m_f2_v3  m_f2_v4  m_f2_v5  m_f2_v6

         var =nada2
 v_f1_v1  v_f1_v2  v_f1_v3  v_f1_v4  v_f1_v5   v_f1_v6
 v_f2_v1  v_f2_v2  v_f2_v3  v_f2_v4  v_f2_v5   v_f2_v6
         max =B;
 title'Media, Variancia e Tamanho da Amostra - Topicos 2
e 4';
run;

%macro alfa(alfa);
data saidaK(keep=LI LS _type_);
```

```
 set saida;
  LI=ROUND(B*&alfa/2); *limite inferior do intervalo
Boostrap*;
  LS=ROUND((B-LI)+1);  *limite superior do intervalo
Boostrap*;
run;
%mend alfa;
%alfa(0.05); *confiança*;

%macro valor_limite(k,j);
proc sort data=in2.jack;
 by f&k._&j;

data boot&k(keep=n _type_ f&k._&j);
 set in2.jack;
  _type_=0;
   n=_n_;
run;

data limite;
 merge saidaK boot&k;
  by _type_;
   if LI=n then LI_Jack_F&k._&j=f&k._&j;
      else      LI_Jack_F&k._&j=.;
   if LS=n then LS_Jack_F&k._&j=f&k._&j;
      else      LS_Jack_F&k._&j=.;
run;

proc means data=limite max noprint;
var LI_Jack_F&k._&j LS_Jack_F&k._&j;
 output out=limites_f&k._&j
        max=LI_Jack LS_Jack;
run;
proc delete data=limite;
proc delete data=boot&k;
run;
%mend valor_limite;
%valor_limite(1,v01 );
%valor_limite(1,v02 );
%valor_limite(1,v03 );
```

```
%valor_limite(1,v04 );
%valor_limite(1,v05 );
%valor_limite(1,v06 );
%valor_limite(2,v01 );
%valor_limite(2,v02 );
%valor_limite(2,v03 );
%valor_limite(2,v04 );
%valor_limite(2,v05 );
%valor_limite(2,v06 );

data limites;
 set
 limites_f1_v01  limites_f1_v02  limites_f1_v03
limites_f1_v04  limites_f1_v05
 limites_f1_v06   limites_f2_v01  limites_f2_v02
limites_f2_v03  limites_f2_v04  limites_f2_v05
 limites_f2_v06;

     if _n_=1  then _name_='f1_v01';
else if _n_=2  then _name_='f1_v02';
else if _n_=3  then _name_='f1_v03';
else if _n_=4  then _name_='f1_v04';
else if _n_=5  then _name_='f1_v05';
else if _n_=6  then _name_='f1_v06';
else if _n_=7  then _name_='f2_v01';
else if _n_=8  then _name_='f2_v02';
else if _n_=9  then _name_='f2_v03';
else if _n_=10 then _name_='f2_v04';
else if _n_=11 then _name_='f2_v05';
else if _n_=12 then _name_='f2_v06';
run;

proc delete data=limites_f1_v01;
proc delete data=limites_f1_v02;
proc delete data=limites_f1_v03;
proc delete data=limites_f1_v04;
proc delete data=limites_f1_v05;
proc delete data=limites_f1_v06;
proc delete data= limites_f2_v01;
proc delete data=limites_f2_v02;
proc delete data=limites_f2_v03;
proc delete data=limites_f2_v04;
proc delete data=limites_f2_v05;
proc delete data= limites_f2_v06;

run;

proc factor data=base
            method=principal
             rotate=Varimax
            outstat=perfil
            nfactors=2 noprint;
 var V1-V6;
run;

data perfil_f1(keep=i f1_v01-f1_v06)
     perfil_f2(keep=i f2_v01-f2_v06);
 set perfil;
 if _name_='Factor1' then do;
    f1_v01= v1 ;   f1_v02= v2 ;   f1_v03= v3 ;   f1_v04=
v4 ;   f1_v05= v5 ;
    f1_v06= v6 ;
   end;
 else if _name_='Factor2' then do;
    f2_v01= v1 ;   f2_v02= v2 ;   f2_v03= v3 ;   f2_v04=
v4 ;   f2_v05= v5 ;
    f2_v06= v6 ;
 end;
    i=1;

     if _name_='Factor1' then output perfil_f1;
else if _name_='Factor2' then output perfil_f2;
run;

data Total(drop=i);
 merge perfil_f1 perfil_f2;
  by i;
run;

proc delete data=perfil;
proc delete data=perfil_f1;
proc delete data=perfil_f2;
run;

proc transpose data=total out=fator;
 var f1_v01-f1_v06 f2_v01-f2_v06;
run;

data fator;
 set fator;
  rename col1=cargas;
run;

proc transpose data=saida out=medias;
 var m_f1_v1-m_f1_v6 m_f2_v1-m_f2_v6;
 id _type_;
run;

proc transpose data=saida out=variancias;
 var v_f1_v1-v_f1_v6 v_f2_v1-v_f2_v6;
 id _type_;
run;

data medias;
 set medias;
  rename _0=media;

  if _name_='m_f1_v1' then _name_='f1_v01';
  if _name_='m_f1_v2' then _name_='f1_v02';
  if _name_='m_f1_v3' then _name_='f1_v03';
  if _name_='m_f1_v4' then _name_='f1_v04';
  if _name_='m_f1_v5' then _name_='f1_v05';
  if _name_='m_f1_v6' then _name_='f1_v06';
  if _name_='m_f2_v1 ' then _name_='f2_v01';
  if _name_='m_f2_v2 ' then _name_='f2_v02';
  if _name_='m_f2_v3 ' then _name_='f2_v03';
  if _name_='m_f2_v4 ' then _name_='f2_v04';
  if _name_='m_f2_v5 ' then _name_='f2_v05';
  if _name_='m_f2_v6 ' then _name_='f2_v06';
run;
```

```
data variancias;
 set variancias;
  rename _0=variancia;
  if _name_='v_f1_v1' then _name_='f1_v01';
  if _name_='v_f1_v2' then _name_='f1_v02';
  if _name_='v_f1_v3' then _name_='f1_v03';
  if _name_='v_f1_v4' then _name_='f1_v04';
  if _name_='v_f1_v5' then _name_='f1_v05';
  if _name_='v_f1_v6' then _name_='f1_v06';
  if _name_='v_f2_v1 ' then _name_='f2_v01';
  if _name_='v_f2_v2 ' then _name_='f2_v02';
  if _name_='v_f2_v3 ' then _name_='f2_v03';
  if _name_='v_f2_v4 ' then _name_='f2_v04';
  if _name_='v_f2_v5 ' then _name_='f2_v05';
  if _name_='v_f2_v6 ' then _name_='f2_v06';
  run;
proc sort data=limites;
 by _name_;
proc sort data=fator;
 by _name_;
proc sort data=medias;
 by _name_;
proc sort data=variancias;
 by _name_;

data total_5_6(keep=fator_variavel cargas media variancia
                    LI_Jack LS_Jack sig_ic vies
erro_medio g);
 merge fator medias variancias limites;
  by _name_;
   rename _name_=fator_variavel;
   vies=cargas-media;
   erro_medio=variancia+vies**2;
   zero=0;
  if LI_Jack<=zero<=LS_Jack then sig_ic='Nao_Signif';
     else                        sig_ic='Signif    ';
  g=cargas/(variancia**(1/2));

run;
```

```
proc transpose data=total_5_6
out=g_total_5_6(drop=_name_);
 var g;
 id fator_variavel;
run;

data g_total_5_6;
 set g_total_5_6;
 _type_=0;
  rename f1_v01=g_f1_v01;
  rename f1_v02=g_f1_v02;
  rename f1_v03=g_f1_v03;
  rename f1_v04=g_f1_v04;
  rename f1_v05=g_f1_v05;
  rename f1_v06=g_f1_v06;
  rename f2_v01=g_f2_v01;
  rename f2_v02=g_f2_v02;
  rename f2_v03=g_f2_v03;
  rename f2_v04=g_f2_v04;
  rename f2_v05=g_f2_v05;
  rename f2_v06=g_f2_v06;
run;

data jack;
 set in2.jack;
  _type_=0;
run;

data jack_norm(keep=norm_f1_v01-norm_f1_v06
                    norm_f2_v01-norm_f2_v06
                    g_f1_v01-g_f1_v06
                    g_f2_v01-g_f2_v06);
 merge jack saida g_total_5_6;
  by _type_;

 norm_f1_v01=(f1_v01-m_f1_v1)/(v_f1_v1**(1/2));
 norm_f1_v02=(f1_v02-m_f1_v2)/(v_f1_v2**(1/2));
 norm_f1_v03=(f1_v03-m_f1_v3)/(v_f1_v3**(1/2));
 norm_f1_v04=(f1_v04-m_f1_v4)/(v_f1_v4**(1/2));
 norm_f1_v05=(f1_v05-m_f1_v5)/(v_f1_v5**(1/2));
 norm_f1_v06=(f1_v06-m_f1_v6)/(v_f1_v6**(1/2));
 norm_f2_v01=(f2_v01-m_f2_v1)/(v_f2_v1**(1/2));
 norm_f2_v02=(f2_v02-m_f2_v2)/(v_f2_v2**(1/2));
 norm_f2_v03=(f2_v03-m_f2_v3)/(v_f2_v3**(1/2));
 norm_f2_v04=(f2_v04-m_f2_v4)/(v_f2_v4**(1/2));
 norm_f2_v05=(f2_v05-m_f2_v5)/(v_f2_v5**(1/2));
 norm_f2_v06=(f2_v06-m_f2_v6)/(v_f2_v6**(1/2));

run;

%macro pvalor(p);

data s_&p(keep=s_&p);
 set jack_norm;
 if (g_&p>0 and norm_&p>g_&p) or
    (g_&p<0 and norm_&p<g_&p) then s_&p=1;
  else                             s_&p=0;
run;

proc means data=s_&p sum noprint;
 var s_&p;
  output out=saida_&p
         sum=;
run;

proc delete data=s_&p;

data saida_&p(keep=fator_variavel p_valor);
 set saida_&p;
  fator_variavel="&p";
  p_valor=s_&p/_freq_;
run;

%mend pvalor;
%pvalor(f1_v01);
%pvalor(f1_v02);
%pvalor(f1_v03);
%pvalor(f1_v04);
%pvalor(f1_v05);
%pvalor(f1_v06);
```

```
%pvalor(f2_v01);
%pvalor(f2_v02);
%pvalor(f2_v03);
%pvalor(f2_v04);
%pvalor(f2_v05);
%pvalor(f2_v06);

data p_valor;
 set saida_f1_v01 saida_f1_v02 saida_f1_v03 saida_f1_v04
saida_f1_v05
     saida_f1_v06 saida_f2_v01 saida_f2_v02 saida_f2_v03
saida_f2_v04
     saida_f2_v05 saida_f2_v06;
  if p_valor>0.05  then sig_vp='Nao_Signif';
     else               sig_vp='Signif    ';

run;

proc delete data=saida_f1_v01;
proc delete data=saida_f1_v02;
proc delete data=saida_f1_v03;
proc delete data=saida_f1_v04;
proc delete data=saida_f1_v05;
proc delete data=saida_f1_v06;
proc delete data=saida_f2_v01;
proc delete data=saida_f2_v02;
proc delete data=saida_f2_v03;
proc delete data=saida_f2_v04;
proc delete data=saida_f2_v05;
proc delete data=saida_f2_v06;
run;

data total_5_6(drop=g);
 merge total_5_6 p_valor;
  by fator_variavel;
run;

proc print data=total_5_6;
 title'Analise Fatorial - Jacknife';
run;
```

Anexo 4

Tabelas dos Testes de Convergência

Tabela 10: Caso 1: Base Busca de Fatores-EMQ - Fator 1

Variáveis	B=25	B=100	B=200	B=1000
V1	1,08E-02	1,12E-02	1,12E-02	1,16E-02
V2	8,75E-03	9,00E-03	9,00E-03	9,46E-03
V3	9,93E-03	5,15E-03	5,15E-03	5,43E-03
V4	2,95E-03	1,65E-03	1,65E-03	1,33E-03
V5	2,86E-03	2,02E-03	2,02E-03	1,91E-03
V6	8,55E-03	6,01E-03	6,01E-03	5,58E-03
V7	1,43E-03	2,04E-03	2,04E-03	2,15E-03
V8	4,14E-03	5,11E-03	5,11E-03	4,99E-03
V9	5,95E-03	5,52E-03	5,52E-03	6,72E-03
V10	8,20E-03	6,07E-03	6,07E-03	5,76E-03
V11	5,88E-03	5,45E-03	5,45E-03	5,30E-03
V12	1,23E-02	1,34E-02	1,34E-02	1,51E-02
V13	6,60E-03	7,24E-03	7,24E-03	6,05E-03
V14	9,45E-03	8,11E-03	8,11E-03	7,17E-03
V15	1,42E-03	1,20E-03	1,20E-03	1,31E-03
V16	6,06E-03	3,70E-03	3,70E-03	3,62E-03
V17	6,50E-03	3,88E-03	3,88E-03	3,77E-03
V18	3,11E-03	3,84E-03	3,84E-03	4,64E-03
V19	1,93E-03	1,81E-03	1,81E-03	1,91E-03
V20	6,16E-03	5,18E-03	5,18E-03	5,52E-03
V21	1,32E-02	1,33E-02	1,33E-02	1,61E-02
V22	1,44E-03	1,84E-03	1,84E-03	1,80E-03
V23	6,02E-03	6,25E-03	6,25E-03	1,16E-02

Tabela 11: Caso 1 : Base Busca de Fatores-EMQ - Fator 2

Variáveis	B=25	B=100	B=200	B=1000
V1	7,79E-03	8,80E-03	8,44E-03	6,93E-03
V2	8,14E-03	1,18E-02	1,28E-02	8,06E-03
V3	1,11E-02	9,81E-03	1,02E-02	1,31E-02
V4	5,75E-03	5,58E-03	5,28E-03	8,98E-03
V5	6,97E-03	7,77E-03	7,76E-03	5,87E-03
V6	1,05E-02	8,81E-03	9,18E-03	7,20E-03
V7	6,75E-03	9,41E-03	8,06E-03	9,07E-03
V8	1,44E-02	1,28E-02	1,20E-02	8,34E-03
V9	1,80E-02	1,42E-02	1,49E-02	1,11E-02
V10	8,14E-03	8,07E-03	7,28E-03	1,41E-02
V11	7,72E-03	8,38E-03	7,62E-03	6,87E-03
V12	6,53E-03	3,55E-03	3,00E-03	7,20E-03
V13	1,40E-02	9,47E-03	8,75E-03	3,30E-03
V14	9,86E-03	6,61E-03	7,13E-03	9,37E-03
V15	5,57E-03	4,05E-03	4,86E-03	7,80E-03
V16	1,13E-02	8,64E-03	7,32E-03	4,74E-03
V17	1,32E-02	1,13E-02	1,04E-02	7,70E-03
V18	7,09E-03	8,09E-03	8,04E-03	9,47E-03
V19	8,71E-03	7,49E-03	7,12E-03	8,17E-03
V20	1,20E-02	1,03E-02	9,55E-03	6,17E-03
V21	5,90E-03	3,35E-03	2,69E-03	9,67E-03
V22	1,02E-02	7,22E-03	8,07E-03	3,03E-03
V23	8,03E-03	7,49E-03	8,86E-03	8,17E-03

Tabela 12: Caso 1: Base Busca de Fatores-Intervalo de Confiança - Fator 1

Variáveis	B=25		B=100		B=200		B=1000	
	LI	LS	LI	LS	LI	LS	LI	LS
V1	-0,28	0,10	-0,28	0,10	-0,33	0,10	-0,31	0,10
V2	0,11	0,49	0,11	0,46	0,08	0,49	0,10	0,47
V3	0,26	0,62	0,32	0,62	0,29	0,62	0,33	0,62
V4	0,60	0,80	0,60	0,77	0,61	0,78	0,63	0,78
V5	0,58	0,78	0,59	0,76	0,58	0,76	0,59	0,76
V6	0,25	0,65	0,33	0,61	0,31	0,61	0,33	0,61
V7	0,58	0,72	0,55	0,72	0,54	0,72	0,55	0,73
V8	0,46	0,68	0,39	0,68	0,39	0,68	0,41	0,69
V9	0,29	0,57	0,29	0,57	0,25	0,57	0,27	0,58
V10	0,38	0,73	0,46	0,75	0,45	0,76	0,45	0,75
V11	0,47	0,72	0,47	0,75	0,48	0,75	0,46	0,75
V12	-0,07	0,34	-0,16	0,33	-0,18	0,32	-0,17	0,30
V13	0,29	0,59	0,23	0,57	0,21	0,57	0,25	0,55
V14	0,34	0,68	0,31	0,65	0,26	0,64	0,30	0,64
V15	0,70	0,84	0,69	0,84	0,69	0,84	0,70	0,84
V16	0,51	0,83	0,53	0,80	0,56	0,80	0,56	0,79
V17	0,48	0,76	0,52	0,76	0,53	0,77	0,53	0,77
V18	0,51	0,72	0,49	0,72	0,46	0,72	0,45	0,72
V19	0,58	0,73	0,55	0,72	0,55	0,73	0,56	0,73
V20	0,25	0,58	0,24	0,51	0,24	0,52	0,23	0,52
V21	-0,09	0,41	-0,11	0,36	-0,14	0,36	-0,13	0,36
V22	0,61	0,77	0,60	0,77	0,60	0,77	0,60	0,77
V23	0,26	0,55	0,25	0,55	0,22	0,56	0,23	0,56

Tabela 13: Caso 1: Base Busca de Fatores-Intervalo de Confiança - Fator 2

Variáveis	B=25		B=100		B=200		B=1000	
	LI	LS	LI	LS	LI	LS	LI	LS
V1	0,29	0,55	0,24	0,59	0,27	0,60	0,26	0,60
V2	0,13	0,48	0,11	0,50	0,08	0,52	0,08	0,53
V3	-0,04	0,35	-0,01	0,35	-0,02	0,37	-0,02	0,35
V4	-0,13	0,15	-0,13	0,17	-0,11	0,17	-0,12	0,19
V5	-0,27	0,12	-0,22	0,12	-0,22	0,16	-0,18	0,15
V6	0,05	0,49	0,06	0,48	0,06	0,48	0,06	0,43
V7	-0,15	0,17	-0,18	0,23	-0,18	0,21	-0,16	0,22
V8	-0,08	0,40	-0,07	0,40	-0,06	0,40	-0,04	0,38
V9	-0,04	0,40	0,00	0,45	0,00	0,47	-0,01	0,43
V10	-0,74	-0,30	-0,65	-0,31	-0,65	-0,31	-0,67	-0,36
V11	-0,71	-0,29	-0,64	-0,29	-0,63	-0,29	-0,65	-0,32
V12	0,50	0,79	0,58	0,81	0,60	0,80	0,59	0,80
V13	0,08	0,57	0,12	0,50	0,14	0,51	0,12	0,49
V14	0,28	0,63	0,35	0,68	0,35	0,68	0,32	0,67
V15	-0,41	-0,17	-0,41	-0,15	-0,41	-0,12	-0,40	-0,13
V16	-0,67	-0,25	-0,62	-0,25	-0,57	-0,27	-0,59	-0,27
V17	-0,65	-0,20	-0,61	-0,18	-0,58	-0,18	-0,57	-0,20
V18	0,05	0,31	-0,03	0,31	-0,03	0,33	-0,04	0,33
V19	-0,11	0,31	0,00	0,34	0,00	0,34	0,01	0,32
V20	0,01	0,40	0,02	0,40	0,02	0,41	0,00	0,38
V21	0,52	0,81	0,61	0,84	0,62	0,82	0,60	0,82
V22	-0,23	0,23	-0,23	0,12	-0,20	0,14	-0,20	0,17
V23	0,24	0,54	0,23	0,53	0,21	0,59	0,21	0,57

Tabela 14: Caso 2: Base Creme Dental-EMQ - Fator 1

Variáveis	B=25	B=100	B=200	B=1000
V1	9,14E-05	3,65E-05	1,82E-05	3,62E-06
V2	1,23E-02	3,98E-03	1,98E-03	3,94E-04
V3	1,76E-04	4,59E-05	2,28E-05	4,55E-06
V4	1,01E-02	3,22E-03	1,60E-03	3,20E-04
V5	3,46E-04	1,20E-04	5,96E-05	1,19E-05
V6	1,16E-02	3,80E-03	1,89E-03	3,76E-04

Tabela 15: Caso 2: Base Creme Dental-EMQ - Fator 2

Variáveis	B=25	B=100	B=200	B=1000
V1	9,04E-03	2,96E-03	1,47E-03	2,93E-04
V2	2,24E-04	7,85E-05	3,91E-05	7,79E-06
V3	9,42E-03	3,05E-03	1,52E-03	3,02E-04
V4	1,05E-03	3,57E-04	1,78E-04	3,54E-05
V5	7,98E-03	2,57E-03	1,28E-03	2,55E-04
V6	4,06E-04	1,27E-04	6,30E-05	1,26E-05

Tabela 16: Caso 2: Base Creme Dental-Intervalo de Confiança - Fator 1

Variáveis	**B=25**		**B=100**		**B=200**		**B=1000**	
	LI	**LS**	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
V1	0,92	0,95	0,93	0,95	0,94	0,95	0,95	0,95
V2	-0,27	0,20	-0,19	0,15	-0,17	0,06	-0,05	-0,05
V3	0,88	0,94	0,90	0,93	0,92	0,93	0,93	0,93
V4	-0,43	0,00	-0,37	-0,10	-0,36	-0,16	-0,24	-0,24
V5	-0,93	-0,85	-0,92	-0,88	-0,92	-0,88	-0,91	-0,91
V6	-0,14	0,33	-0,09	0,26	-0,05	0,16	0,08	0,08

Tabela 17: Caso 2: Base Creme Dental-Intervalo de Confiança - Fator 2

Variáveis	**B=25**		**B=100**		**B=200**		**B=1000**	
	LI	**LS**	**LI**	**LS**	**LI**	**LS**	**LI**	**LS**
V1	-0,16	0,25	-0,08	0,22	-0,01	0,18	0,07	0,07
V2	0,79	0,85	0,81	0,85	0,81	0,84	0,84	0,84
V3	-0,30	0,13	-0,22	0,07	-0,14	0,05	-0,06	-0,06
V4	0,75	0,88	0,78	0,86	0,79	0,86	0,83	0,83
V5	-0,36	0,00	-0,30	-0,03	-0,29	-0,10	-0,18	-0,18
V6	0,80	0,88	0,83	0,88	0,85	0,87	0,87	0,87

Tabela 18: Caso 3: Base Modo de Vida-EMQ - Fator 1

Variáveis	B=25	B=100	B=200	B=1000
V1	1,18E-03	2,86E-04	1,42E-04	2,83E-05
V2	1,03E-02	2,50E-03	1,24E-03	2,48E-04
V3	1,03E-04	2,50E-05	1,24E-05	2,48E-06
V4	7,44E-03	1,80E-03	8,97E-04	1,79E-04
V5	2,36E-03	5,73E-04	2,85E-04	5,68E-05
V6	1,15E-02	2,79E-03	1,39E-03	2,76E-04
V7	1,96E-03	4,75E-04	2,36E-04	4,71E-05

Tabela 19: Caso 3 : Base Modo de Vida-EMQ - Fator 2

Variáveis	B=25	B=100	B=200	B=1000
V1	2,15E-02	5,22E-03	2,60E-03	5,18E-04
V2	9,88E-02	2,40E-02	1,19E-02	2,38E-03
V3	4,25E-03	1,03E-03	5,12E-04	1,02E-04
V4	6,05E-02	1,47E-02	7,31E-03	1,46E-03
V5	2,93E-02	7,11E-03	3,54E-03	7,05E-04
V6	7,14E-02	1,73E-02	8,63E-03	1,72E-03
V7	2,59E-02	6,28E-03	3,12E-03	6,22E-04

Tabela 20: Caso 3 : Base Modo de Vida-Intervalo de Confiança - Fator 1

Variáveis	B=25		B=100		B=200		B=1000	
	LI	LS	LI	LS	LI	LS	LI	LS
V1	0,75	0,89	0,76	0,87	0,78	0,85	0,82	0,82
V2	0,12	0,50	0,14	0,43	0,16	0,38	0,28	0,28
V3	0,87	0,91	0,87	0,90	0,88	0,89	0,89	0,89
V4	-0,35	0,02	-0,32	-0,03	-0,27	-0,15	-0,20	-0,20
V5	0,60	0,79	0,61	0,78	0,62	0,70	0,66	0,66
V6	-0,18	0,27	-0,09	0,22	-0,04	0,15	0,05	0,05
V7	-0,77	-0,59	-0,74	-0,59	-0,72	-0,64	-0,68	-0,68

Tabela 21: Caso 3: Base Modo de Vida-Intervalo de Confiança - Fator 2

Variáveis	B=25		B=100		B=200		B=1000	
	LI	LS	LI	LS	LI	LS	LI	LS
V1	-0,26	0,50	0,23	0,47	0,31	0,46	0,38	0,38
V2	-0,80	0,80	-0,78	-0,60	-0,74	-0,64	-0,71	-0,71
V3	-0,18	0,09	-0,13	0,08	-0,10	0,02	-0,03	-0,03
V4	-0,54	0,71	0,51	0,69	0,55	0,67	0,63	0,63
V5	-0,24	0,63	0,26	0,62	0,44	0,58	0,51	0,51
V6	-0,72	0,67	-0,69	-0,49	-0,65	-0,53	-0,60	-0,60
V7	-0,34	0,52	0,23	0,49	0,32	0,44	0,38	0,38