

# Estrutura hierárquica e semântica do afeto

Frederico Pedrosa

2025-10-23

## Contents

<b>1</b>	<b>Preparação do ambiente</b>	<b>1</b>
<b>2</b>	<b>Estimativas com embeddings</b>	<b>2</b>
2.1	Carregar embeddings e transpor a matriz para 768 x 45 . . . . .	2
2.2	Verificar se a matriz é fatorável . . . . .	3
2.3	Análise Paralela de Horn para Componentes Principais e Análise Fatorial . . . . .	4
2.4	PCA com Varimax para tentar identificar melhor as palavras que representam os PCs . . . . .	10
2.5	Formativo vs Reflexivo . . . . .	12
<b>3</b>	<b>Estimativas com PANAS</b>	<b>20</b>
3.1	Fatorabilidade . . . . .	20
3.2	Análise Paralela de Horn . . . . .	21
3.3	Formativo vs reflexivo - PANAS . . . . .	28

## 1 Preparação do ambiente

```
if (!require("pacman")) {  
  install.packages("pacman")  
}  
  
## Carregando pacotes exigidos: pacman  
  
# Passo 2: Usar a função p_load() do pacman para instalar (se necessário) e carregar todos os pacotes.  
# Você só precisa listar os nomes dos pacotes, sem aspas.  
pacman::p_load(  
  # Leitura e Manipulação de Dados  
  readr, readxl, dplyr, tidyr, janitor, stringi, stringr,  
  
  # Análise Fatorial e Psicometria  
  psych, EFA.dimensions, GPArotation,
```

```

# Modelagem de Equações Estruturais
lavaan, semTools, seminr, cSEM,

# Análise de Redes
EGAnet,

# Visualização de Dados
ggplot2, patchwork, ggrepel, GGally, semPlot, plotly
)

# Mensagem de confirmação
cat("Todos os pacotes necessários foram verificados e carregados com sucesso.")

```

## Todos os pacotes necessários foram verificados e carregados com sucesso.

## 2 Estimativas com embeddings

### 2.1 Carregar embeddings e transpor a matriz para 768 x 45

```

data <- read_csv("embeddings_circumplex.csv")

## Rows: 45 Columns: 769
## -- Column specification -----
## Delimiter: ","
## chr (1): palavra
## dbl (768): dim_1, dim_2, dim_3, dim_4, dim_5, dim_6, dim_7, dim_8, dim_9, di...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

cat("Dimensões do arquivo original (Linhas x Colunas):", dim(data), "\n")

## Dimensões do arquivo original (Linhas x Colunas): 45 769

print("Primeiras linhas do arquivo original:")

## [1] "Primeiras linhas do arquivo original:"

print(head(data))

## # A tibble: 6 x 769
##   palavra    dim_1    dim_2    dim_3    dim_4    dim_5    dim_6    dim_7    dim_8
##   <chr>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>     <dbl>
## 1 foda     -0.0177  0.0514  -0.0165  0.0392  0.0824  0.0450  -0.00317 -0.0190
## 2 grande   -0.0474  0.0904  -0.0152  0.0436  0.102   -0.00491 -0.0896  -0.0344
## 3 saudade  -0.0177  0.125   -0.0155  -0.00331 0.105   0.0108  -0.0563  -0.0258
## 4 merda    0.0104  0.0927  -0.0157  0.0430  0.0844  0.0552  0.0201  -0.0119

```

```

## 5 bons      -0.0502 0.109  -0.0152  0.000375 0.0937 -0.00222 -0.0901  -0.0242
## 6 descanse -0.0687 0.179  -0.0157  0.0744   0.103  -0.0347  -0.0802   0.0179
## # i 760 more variables: dim_9 <dbl>, dim_10 <dbl>, dim_11 <dbl>, dim_12 <dbl>,
## #   dim_13 <dbl>, dim_14 <dbl>, dim_15 <dbl>, dim_16 <dbl>, dim_17 <dbl>,
## #   dim_18 <dbl>, dim_19 <dbl>, dim_20 <dbl>, dim_21 <dbl>, dim_22 <dbl>,
## #   dim_23 <dbl>, dim_24 <dbl>, dim_25 <dbl>, dim_26 <dbl>, dim_27 <dbl>,
## #   dim_28 <dbl>, dim_29 <dbl>, dim_30 <dbl>, dim_31 <dbl>, dim_32 <dbl>,
## #   dim_33 <dbl>, dim_34 <dbl>, dim_35 <dbl>, dim_36 <dbl>, dim_37 <dbl>,
## #   dim_38 <dbl>, dim_39 <dbl>, dim_40 <dbl>, dim_41 <dbl>, dim_42 <dbl>, ...

# --- Passo 3: Preparar o DataFrame para Análise ---
# Seleciona todas as colunas, EXCETO a primeira coluna 'palavra'
# A função select() do dplyr é mais explícita e segura para isso
embeddings_matrix <- data %>%
  select(-palavra)

print("\nÚltimas linhas da matriz de embeddings pronta para análise:")

## [1] "\nÚltimas linhas da matriz de embeddings pronta para análise:"

print(tail(embeddings_matrix))

## # A tibble: 6 x 768
##       dim_1   dim_2   dim_3   dim_4   dim_5   dim_6   dim_7   dim_8   dim_9 dim_10
##       <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1  0.0132  0.139  -0.0165  0.0440  0.0908  0.0228 -0.00486 -0.0215  -0.0283  0.0853
## 2 -0.0365  0.0471  -0.0175  0.0156  0.0795  0.0494 -0.129   -0.0490   0.00369 0.110
## 3 -0.00446 0.0765  -0.0176  0.0571  0.0892  0.0183  0.00473 -0.00311 -0.0192  0.0863
## 4 -0.00477 0.151   -0.0158  0.0310  0.0590  0.0623 -0.0435  -0.0371   0.00316 0.0854
## 5 -0.0233  0.0185  -0.0179  0.0351  0.0735  0.0101 -0.0373  -0.0110  -0.00532 0.0934
## 6 -0.0211  0.162   -0.0154  0.0256  0.0544  0.0554 -0.0346  -0.0209  0.00552 0.0771
## # i 758 more variables: dim_11 <dbl>, dim_12 <dbl>, dim_13 <dbl>, dim_14 <dbl>,
## #   dim_15 <dbl>, dim_16 <dbl>, dim_17 <dbl>, dim_18 <dbl>, dim_19 <dbl>,
## #   dim_20 <dbl>, dim_21 <dbl>, dim_22 <dbl>, dim_23 <dbl>, dim_24 <dbl>,
## #   dim_25 <dbl>, dim_26 <dbl>, dim_27 <dbl>, dim_28 <dbl>, dim_29 <dbl>,
## #   dim_30 <dbl>, dim_31 <dbl>, dim_32 <dbl>, dim_33 <dbl>, dim_34 <dbl>,
## #   dim_35 <dbl>, dim_36 <dbl>, dim_37 <dbl>, dim_38 <dbl>, dim_39 <dbl>,
## #   dim_40 <dbl>, dim_41 <dbl>, dim_42 <dbl>, dim_43 <dbl>, dim_44 <dbl>, ...

# --- Passo 1: Transpor a Matriz de Embeddings ---
transposed_matrix <- t(embeddings_matrix)

# --- Passo 2: Atribuir os Nomes das Palavras como Nomes de Coluna ---
colnames(transposed_matrix) <- data$palavra

```

## 2.2 Verificar se a matriz é fatorável

```
cat("---- PASSO 1: Teste de Esfericidade de Bartlett ---\n")
```

```
## --- PASSO 1: Teste de Esfericidade de Bartlett ---
```

```
cor_matrix <- cor(transposed_matrix, use = "pairwise.complete.obs")
bartlett_test <- cortest.bartlett(cor_matrix, n = nrow(transposed_matrix))
print(bartlett_test)
```

```
## $chisq
## [1] 82501.89
##
## $p.value
## [1] 0
##
## $df
## [1] 990
```

```
cat("\n\n--- PASSO 2: Medida de Adequação da Amostra (KMO) ---\n")
```

```
##
##
## --- PASSO 2: Medida de Adequação da Amostra (KMO) ---
```

```
kmo_test <- KMO(cor_matrix)
print(kmo_test)
```

```
## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = cor_matrix)
## Overall MSA = 0.96
## MSA for each item =
##      foda      grande      saudade      merda      bons
##      0.96      0.98      0.97      0.95      0.97
##      descanse   delicia     triste     sozinho     tristeza
##      0.93      0.98      0.95      0.98      0.95
##      maravilha    louco     dançando   charmosa     chique
##      0.96      0.99      0.96      0.92      0.91
##      linda       paz      feliz      vibe      maravilhosa
##      0.97      0.94      0.97      0.97      0.95
##      chore      chorando    suave     relaxar     tranquilidade
##      0.96      0.96      0.97      0.95      0.94
##      amo        amei      amor      gosto      coração
##      0.93      0.96      0.97      0.93      0.96
##      lembro     graça     dispara    penso      calma
##      0.96      0.96      0.95      0.95      0.95
##      top        perfeita    ruim      gostei      lixo
##      0.98      0.98      0.97      0.93      0.98
##      viciada    pedrada    inferno    pancada    recordações
##      0.98      0.96      0.94      0.97      0.94
```

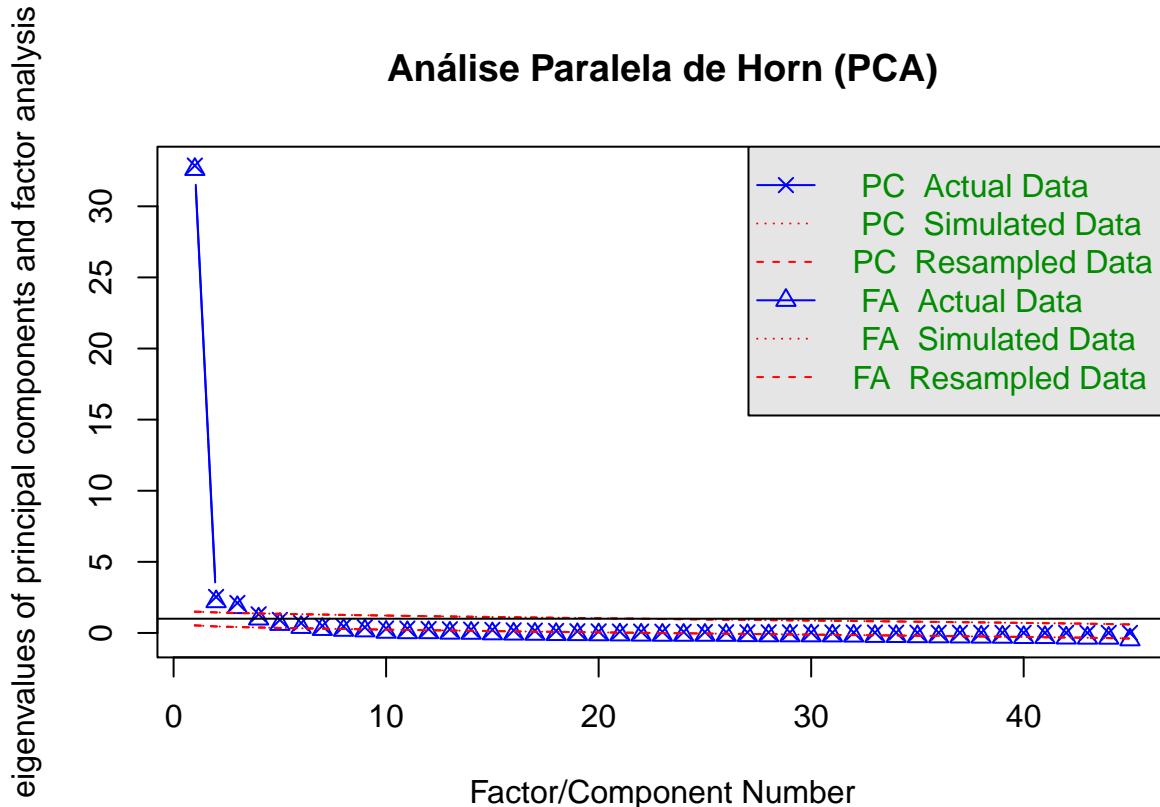
## 2.3 Análise Paralela de Horn para Componentes Principais e Análise Fatorial

```
parallel_analysis_fa <- fa.parallel(
  transposed_matrix,
```

```

    fa = "both",
    n.iter = 100,
    show.legend = TRUE,
    main = "Análise Paralela de Horn (PCA)"
)

```



```

## Parallel analysis suggests that the number of factors = 6 and the number of components = 3

# Dataframe para o gráfico da PCA
df_plot_pca <- data.frame(
  Numero = 1:length(parallel_analysis_fa$pc.values),
  Autovalor_Real = parallel_analysis_fa$pc.values,
  Autovalor_Simulado = parallel_analysis_fa$pc.sim
)

# Dataframe para o gráfico da AFE
df_plot_afe <- data.frame(
  Numero = 1:length(parallel_analysis_fa$fa.values),
  Autovalor_Real = parallel_analysis_fa$fa.values,
  Autovalor_Simulado = parallel_analysis_fa$fa.sim
)

# --- PASSO 2: Criar o gráfico da PCA ---
plot_pca <- ggplot(df_plot_pca, aes(x = Numero)) +

```

```

geom_line(aes(y = Autovalor_Real, color = "Dados Reais (PCA)", size = 0.7) +
geom_point(aes(y = Autovalor_Real, color = "Dados Reais (PCA)", shape = 17, size = 3) +
geom_line(aes(y = Autovalor_Simulado, color = "Dados Simulados (PCA)", linetype = "dashed", size = 0.7) +
geom_hline(yintercept = 1, linetype = "dotted", color = "black") +
annotate("text", x = max(df_plot_pca$Numero) * 0.9, y = 1.3, label = "", size = 3) +
scale_color_manual(name = "Análise", values = c("Dados Reais (PCA)" = "blue", "Dados Simulados (PCA)" = "darkgreen")) +
labs(
  title = "Análise Paralela (PCA)",
  x = "Número do Componente",
  y = "Autovalor (Eigenvalue)"
) +
theme_minimal(base_size = 12) +
theme(legend.position = "top", plot.title = element_text(hjust = 0.5, face = "bold")) +
scale_x_continuous(breaks = seq(0, 45, by = 5))

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

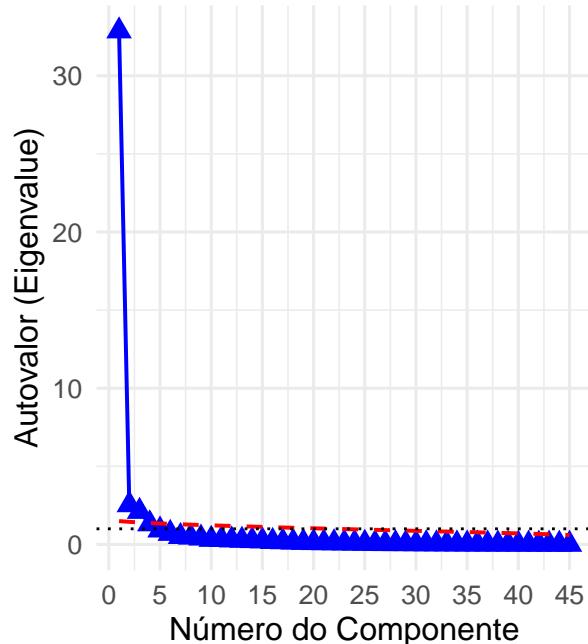
# --- PASSO 3: Criar o gráfico da AFE ---
plot_afe <- ggplot(df_plot_afe, aes(x = Numero)) +
  geom_line(aes(y = Autovalor_Real, color = "Dados Reais (AFE)", size = 0.7) +
  geom_point(aes(y = Autovalor_Real, color = "Dados Reais (AFE)", shape = 17, size = 3) +
  geom_line(aes(y = Autovalor_Simulado, color = "Dados Simulados (AFE)", linetype = "dashed", size = 0.7) +
  geom_hline(yintercept = 1, linetype = "dotted", color = "black") +
  annotate("text", x = max(df_plot_afe$Numero) * 0.9, y = 1.3, label = "", size = 3) +
  scale_color_manual(name = "Análise", values = c("Dados Reais (AFE)" = "darkgreen", "Dados Simulados (AFE)" = "blue")) +
  labs(
    title = "Análise Paralela (AFE)",
    x = "Número do Fator",
    y = "Autovalor (Eigenvalue)"
) +
  theme_minimal(base_size = 12) +
  theme(legend.position = "top", plot.title = element_text(hjust = 0.5, face = "bold")) +
  scale_x_continuous(breaks = seq(0, 45, by = 5))

# --- PASSO 4: Combinar os dois gráficos ---
# O operador '+' do patchwork une os gráficos lado a lado
plot_pca + plot_afe

```

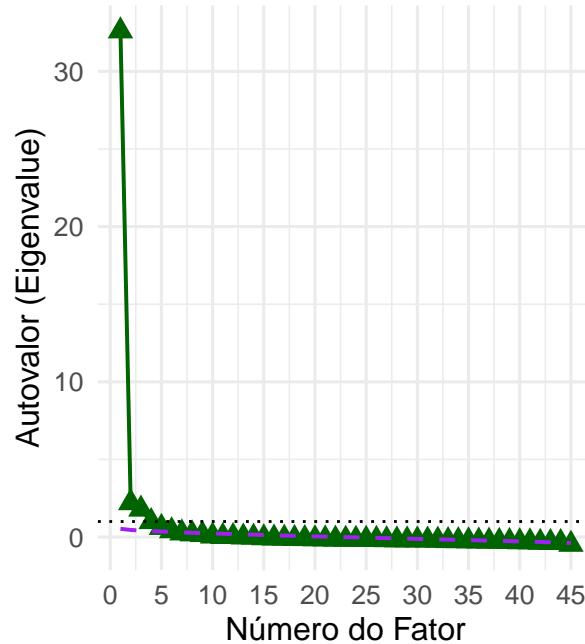
## Análise Paralela (PCA)

Analise → Dados Reais (PCA) - - - Dados Simulac



## Análise Paralela (AFE)

Analise → Dados Reais (AFE) - - - Dados Simulac



## PCA - identificação do fator geral

1a dimensão em que todas as palavras carregam de forma positiva

```
cat("\n--- Rodando PCA para extrair 3 componentes ---\n")
```

```
##  
## --- Rodando PCA para extrair 3 componentes ---
```

```
pca_results_psych <- principal(  
  r = transposed_matrix,  
  nfactors = 3,  
  rotate = "none" # Sem rotação para ver a estrutura bruta  
)  
print(pca_results_psych$loadings, cutoff = 0.3, sort = TRUE)
```

```
##  
## Loadings:  
##          PC1    PC2    PC3  
## foda      0.816  0.399  0.301  
## grande    0.946  
## saudade   0.873  
## merda     0.785  0.351  0.416  
## bons      0.925  
## descanse   0.767  
## delicia    0.869
```

```

## triste          0.865
## sozinho         0.840
## tristeza       0.837 -0.387
## maravilha      0.903
## louco           0.846  0.413
## dançando        0.822  0.396
## charmosa        0.740  0.425
## chique           0.721  0.445
## linda            0.912
## paz              0.854 -0.363
## feliz             0.870
## vibe              0.932
## maravilhosa     0.896
## chore             0.841
## chorando         0.796
## suave             0.887
## relaxar           0.790 -0.305
## tranquilidade    0.830 -0.376
## amo               0.886
## amei              0.896
## amor              0.893
## gosto             0.910
## coração          0.916
## lembro            0.833
## graça              0.934
## dispara            0.863
## penso              0.874
## calma              0.904
## top                0.817
## perfeita           0.841
## ruim                0.808      0.398
## gostei              0.852
## lixo                0.755  0.333  0.385
## viciada            0.861
## pedrada            0.852      0.307
## inferno             0.798      0.423
## pancada             0.920
## recordações        0.801      0.376
##
##                  PC1   PC2   PC3
## SS loadings      32.863 2.532 2.108
## Proportion Var  0.730 0.056 0.047
## Cumulative Var  0.730 0.787 0.833

```

```
pca_results_psych$fit.off
```

```
## [1] 0.9972998
```

### 2.3.1 A extração do fator geral faz com que vejamos o modelo circumplexo

```

loadings_df <- as.data.frame(unclass(pca_results_psych$loadings))
loadings_df$palavra <- rownames(loadings_df)

```

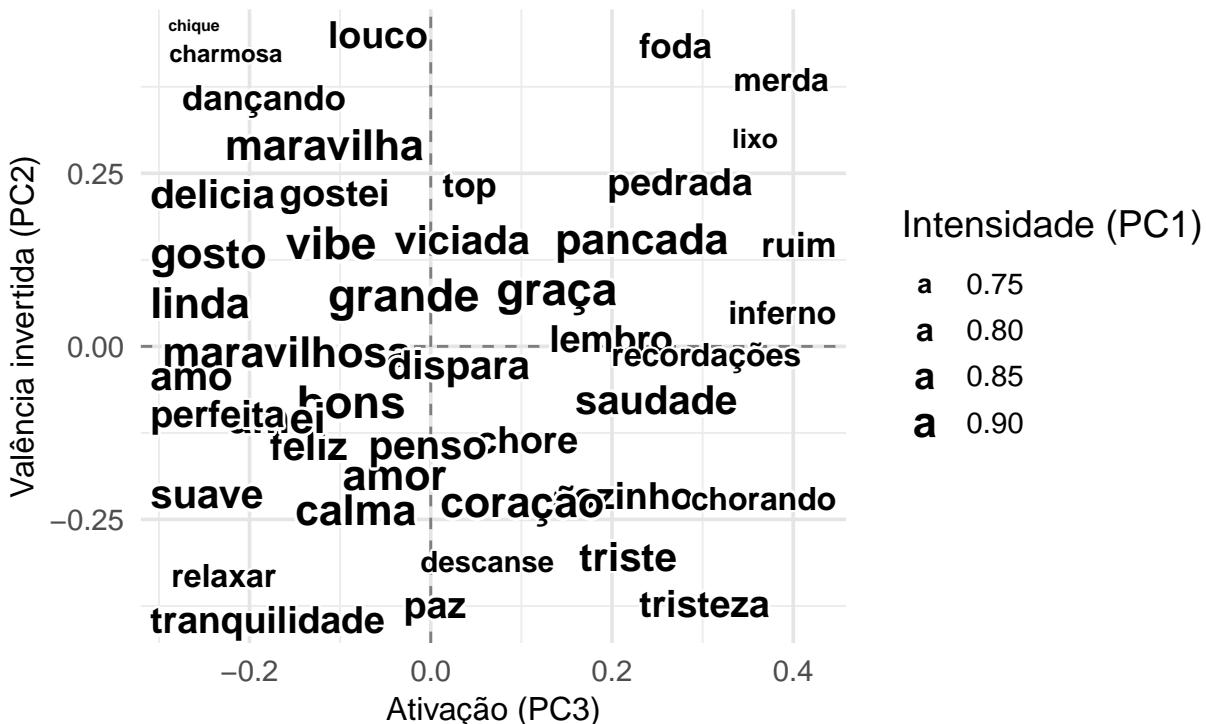
```

grafico_pca_intensidade_final <- ggplot(
  data = loadings_df,
  aes(x = PC3, y = PC2, label = palavra, size = PC1)
) +
  # Linhas de referência dos quadrantes
  geom_hline(yintercept = 0, linetype = "dashed", color = "gray50") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "gray50") +
  # Camada de texto com repulsão
  geom_text_repel(
    fontface = "bold",
    color = "black",
    bg.color = "white",
    bg.r = 0.1,
    segment.color = "transparent",
    max.overlaps = Inf
  ) +
  # Customização da escala de tamanho
  scale_size_continuous(
    range = c(2, 6), # Mantive seu range preferido
    name = "Intensidade (PC1)"
  ) +
  # ### ALTERAÇÃO 1: Horizontalizar o Gráfico #####
  # A linha 'coord_fixed()' foi REMOVIDA.
  # Agora o gráfico irá se expandir para preencher o espaço disponível.

  # ### ALTERAÇÃO 2: Simplificar os Títulos dos Eixos #####
  labs(
    title = "",
    subtitle = "",
    x = "Ativação (PC3)", # Texto da variância removido
    y = "Valência invertida (PC2)" # Texto da variância removido
  ) +
  # Tema visual limpo
  theme_minimal(base_size = 14) +
  theme(
    panel.grid.major = element_line(color = "gray90"),
    plot.title = element_text(hjust = 0.5, face = "bold", size = 18),
    plot.subtitle = element_text(hjust = 0.5, size = 11),
    legend.position = "right",
    axis.title = element_text(size = 12)
  )
  # Exibir o gráfico final

```

```
print(grafico_pca_intensidade_final)
```



## 2.4 PCA com Varimax para tentar identificar melhor as palavras que representam os PCs

```
cat("\n--- Rodando PCA para extraer 3 componentes ---\n")
```

```
##  
## --- Rodando PCA para extrair 3 componentes ---  
  
pca_results_psych <- principal(  
  r = transposed_matrix,  
  nfactors = 3,  
  rotate = "varimax"  
)  
print(pca_results_psych$loadings, cutoff = 0.3, sort = TRUE)  
  
##  
## Loadings:  
##          RC1    RC2    RC3  
## grande     0.666  0.516  0.433  
## saudade   -0.665 -0.307  0.524
```

```

## bons          0.653 0.525 0.397
## descanse     0.715      0.307
## triste       0.752      0.532
## sozinho      0.678      0.494
## tristeza     0.827      0.448
## paz          0.846 0.311
## feliz         0.720 0.449
## chore         0.685 0.335 0.398
## chorando     0.694      0.512
## suave        0.712 0.555
## relaxar       0.768 0.357
## tranquilidade 0.846 0.336
## amei         0.667 0.581
## amor          0.761 0.377 0.354
## coração      0.806 0.338 0.383
## lembro        0.600 0.385 0.441
## graça         0.626 0.474 0.511
## dispara       0.634 0.414 0.422
## penso         0.666 0.408 0.408
## calma         0.812 0.414
## perfeita      0.630 0.589
## delicia       0.471 0.706 0.327
## maravilha    0.429 0.803 0.344
## louco          0.764 0.487
## dançando      0.815 0.370
## charmosa      0.828
## chique         0.829
## linda          0.628 0.676
## vibe           0.611 0.613 0.369
## maravilhosa   0.634 0.645
## amo            0.632 0.636
## gosto          0.582 0.659 0.312
## top            0.371 0.615 0.461
## gostei         0.512 0.636 0.315
## viciada       0.428 0.575 0.518
## foda           0.506 0.780
## merda          0.393 0.840
## ruim           0.370 0.320 0.778
## lixo            0.384 0.794
## pedrada        0.340 0.451 0.759
## inferno        0.486      0.734
## pancada        0.477 0.540 0.611
## recordações   0.536      0.678
##
##                    RC1    RC2    RC3
## SS loadings     16.140 11.815 9.548
## Proportion Var  0.359  0.263  0.212
## Cumulative Var 0.359  0.621  0.833

```

```
pca_results_psych$fit.off
```

```
## [1] 0.9972998
```

## 2.5 Formativo vs Reflexivo

Por PLS-SEm vamos tentar ver se o fator geral neste modelo é formativo ou reflexivo

```
# Limpe os nomes das colunas!
# "recordações" se tornará "recordacoes", etc.
dados_df <- as.data.frame(transposed_matrix)
dados_limpos <- clean_names(dados_df)
names(dados_limpos)

## [1] "foda"          "grande"        "saudade"       "merda"
## [5] "bons"          "descanse"      "delicia"       "triste"
## [9] "sozinho"       "tristeza"      "maravilha"     "louco"
## [13] "dancando"     "charmosa"      "chique"        "linda"
## [17] "paz"           "feliz"         "vibe"          "maravilhosa"
## [21] "chore"         "chorando"     "suave"         "relaxar"
## [25] "tranquilidade" "amo"          "amei"          "amor"
## [29] "gosto"         "coracao"      "lembro"        "graca"
## [33] "dispara"       "penso"         "calma"         "top"
## [37] "perfeita"      "ruim"          "gostei"        "lixo"
## [41] "viciada"       "pedrada"      "inferno"       "pancada"
## [45] "recordacoes"

# O seu código original agora funciona perfeitamente
mm_primeira_ordem <- constructs(
  composite("AltaValencia",
    c("lembro", "chique", "charmosa", "dancando", "maravilha", "louco",
      "delicia", "linda", "gosto", "maravilhosa", "gostei", "amo",
      "top", "vibe"),
    weights = mode_A),
  composite("BaixaValencia",
    c("viciada", "merda", "lixo", "foda", "ruim", "pedrada",
      "inferno", "recordacoes", "pancada"),
    weights = mode_A)
)

sm_primeira_ordem <- relationships(
  paths(from = "AltaValencia", to = "BaixaValencia")
)

# Use o dataframe com os nomes limpos!
pls_primeira_ordem <- estimate_pls(
  data = dados_limpos, # << USANDO OS DADOS LIMPOS
  measurement_model = mm_primeira_ordem,
  structural_model = sm_primeira_ordem
)

## Generating the seminr model

## All 768 observations are valid.
```

```

# Se tudo correu bem, agora você pode ver o sumário
summary(pls_primeira_ordem)

## 
## Results from package seminr (2.3.7)
## 
## Path Coefficients:
##          BaixaValencia
## R^2           0.714
## AdjR^2        0.714
## AltaValencia  0.845
## 
## Reliability:
##      alpha rhoC   AVE rhoA
## AltaValencia 0.980 0.982 0.798 0.982
## BaixaValencia 0.974 0.977 0.828 0.977
## 
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5

# 4. Extraír os escores e juntar com os dados originais
construct_scores <- pls_primeira_ordem$construct_scores
dados_com_escores <- cbind(as.data.frame(dados_limpos), construct_scores)

```

### 2.5.1 Segunda ordem formativa

Para validar um construto formativo, ele precisa APONTAR para algo. Como não temos uma variável externa, podemos usar um “construto-âncora” Vamos criar um construto-âncora reflexivo com alguns itens do próprio RC1 Apenas para que o FatorGeral tenha um “alvo” e seus pesos possam ser estimados.

```

mm_segunda_ordem <- constructs(
  # Construtos de primeira ordem
  composite("AltaValencia", "AltaValencia"), # Item único
  composite("BaixaValencia", "BaixaValencia"),# Item único

  # O construto de segunda ordem FORMATIVO
  composite("FatorGeral",
    c("AltaValencia", "BaixaValencia"), # Formado pelos escores
    weights = mode_B)
)

# 2. Criar um modelo estrutural para testar

mm_completo <- constructs(
  composite("AltaValencia", "AltaValencia"),
  composite("BaixaValencia", "BaixaValencia"),
  composite("FatorGeral", c("AltaValencia", "BaixaValencia"), weights = mode_B),

  # Construto-âncora (Outcome) para validar o FatorGeral
  composite("Ancora", c("paz", "calma", "tristeza", "amor"), weights = mode_A)
)

sm_final <- relationships(

```

```

    paths(from = "FatorGeral", to = "Ancora")
)

# 3. Estimar o modelo final
pls_final <- estimate_pls(
  data = dados_com_escores, # Usar os dados com os escores calculados
  measurement_model = mm_completo,
  structural_model = sm_final
)

## Generating the seminr model

## All 768 observations are valid.

# 4. Analisar os resultados!
summary_final <- summary(pls_final)

print(summary_final$reliability)

##           alpha rhoC   AVE   rhoA
## FatorGeral 0.916 0.952 0.909 1.000
## Ancora      0.954 0.967 0.879 0.957
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5

print(summary_final$validity$vif_items)

## FatorGeral :
## AltaValencia BaixaValencia
##          3.498        3.498
##
## Ancora :
##      paz     calma tristeza     amor
##      5.034     5.701     4.272     4.434

# Rodar bootstrapping para obter p-valores para os pesos
boot_results <- bootstrap_model(pls_final, nboot = 100)

## Bootstrapping model using seminr...

## SEMinR Model successfully bootstrapped

summary_boot <- summary(boot_results)

# Agora olhe os p-valores dos pesos em:
summary_boot$bootstrapped_weights

##                                     Original Est. Bootstrap Mean Bootstrap SD T Stat.
## AltaValencia  ->  FatorGeral            0.744         0.742      0.040  18.414

```

```

## BaixaValencia -> FatorGeral          0.288      0.291      0.043   6.689
## paz -> Ancora                      0.258      0.258      0.003  89.148
## calma -> Ancora                     0.282      0.282      0.005  61.543
## tristeza -> Ancora                  0.246      0.246      0.003  76.362
## amor -> Ancora                     0.280      0.280      0.004  79.492
##                                         2.5% CI 97.5% CI
## AltaValencia -> FatorGeral        0.655      0.814
## BaixaValencia -> FatorGeral       0.212      0.381
## paz -> Ancora                      0.252      0.263
## calma -> Ancora                     0.275      0.292
## tristeza -> Ancora                  0.240      0.252
## amor -> Ancora                     0.273      0.288

```

## 2.5.2 Segunda ordem reflexiva

```

mm_segunda_ordem <- constructs(
  # Construtos de primeira ordem
  composite("AltaValencia", "AltaValencia"), # Item único
  composite("BaixaValencia", "BaixaValencia"),# Item único

  # O construto de segunda ordem FORMATIVO
  composite("FatorGeral",
    c("AltaValencia", "BaixaValencia"), # Formado pelos escores
    weights = mode_A)
)

# 2. Criar um modelo estrutural para testar

mm_completo <- constructs(
  composite("AltaValencia", "AltaValencia"),
  composite("BaixaValencia", "BaixaValencia"),
  composite("FatorGeral", c("AltaValencia", "BaixaValencia"), weights = mode_B),

  # Construto-âncora (Outcome) para validar o FatorGeral
  composite("Ancora", c("paz", "calma", "tristeza", "amor"), weights = mode_A)
)

sm_final <- relationships(
  paths(from = "FatorGeral", to = "Ancora")
)

# 3. Estimar o modelo final
pls_final <- estimate_pls(
  data = dados_com_escores, # Usar os dados com os escores calculados
  measurement_model = mm_completo,
  structural_model = sm_final
)

## Generating the seminr model

## All 768 observations are valid.

```

```

# 4. Analisar os resultados!
summary_final <- summary(pls_final)

print(summary_final$reliability)

##           alpha rhoC   AVE rhoA
## FatorGeral 0.916 0.952 0.909 1.000
## Ancora      0.954 0.967 0.879 0.957
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5

print(summary_final$validity$vif_items)

## FatorGeral :
## AltaValencia BaixaValencia
##            3.498          3.498
##
## Ancora :
##      paz     calma tristeza     amor
##    5.034    5.701    4.272    4.434

# Rodar bootstrapping para obter p-valores para os pesos
boot_results <- bootstrap_model(pls_final, nboot = 100)

## Bootstrapping model using seminr...

## SEMinR Model successfully bootstrapped

summary_boot <- summary(boot_results)

# Agora olhe os p-valores dos pesos em:
summary_boot$bootstrapped_weights

##                                     Original Est. Bootstrap Mean Bootstrap SD T Stat.
## AltaValencia -> FatorGeral          0.744       0.741      0.044 16.781
## BaixaValencia -> FatorGeral         0.288       0.292      0.047  6.086
## paz -> Ancora                      0.258       0.258      0.003 92.754
## calma -> Ancora                     0.282       0.282      0.004 63.577
## tristeza -> Ancora                  0.246       0.247      0.003 88.607
## amor -> Ancora                      0.280       0.280      0.004 69.938
##                               2.5% CI 97.5% CI
## AltaValencia -> FatorGeral        0.652       0.820
## BaixaValencia -> FatorGeral       0.206       0.385
## paz -> Ancora                      0.252       0.263
## calma -> Ancora                     0.274       0.290
## tristeza -> Ancora                  0.241       0.251
## amor -> Ancora                      0.274       0.289

```

### 2.5.3 Reflexivo pelo lavaan

O modelo não ajusta mesmo com excelente confiabilidade

```

modelo_semantico <- '
# Fatores de Primeira Ordem
AltaV    =~ lembro + chique + charmosa + dancando + maravilha + louco +
          delicia + linda + gosto + maravilhosa + gostei + amo+ top + vibe
BaixaV   =~ viciada + merda + lixo + foda + ruim + pedrada + inferno + recordacoes + pancada
'

fit_semantico <- cfa(modelo_semantico, data = dados_limpos, estimator = "MLR", orthogonal = F)
fitmeasures(fit_semantico, c("chisq", "df", "pvalue", "cfi", "rmsea",
                             "rmsea.ci.lower", "rmsea.ci.upper"))

##      chisq        df     pvalue      cfi      rmsea
## 15807.340 229.000 0.000 0.610 0.298
## rmsea.ci.lower
##                 0.294

summary(fit_semantico, fit.measures = TRUE, standardized = TRUE)

## lavaan 0.6-20 ended normally after 281 iterations
##
##   Estimator               ML
## Optimization method    NLMINB
## Number of model parameters    47
## 
##   Number of observations    768
## 
## Model Test User Model:
##                               Standard    Scaled
##   Test Statistic            15807.340 12525.865
##   Degrees of freedom           229      229
##   P-value (Chi-square)         0.000      0.000
##   Scaling correction factor                  1.262
##       Yuan-Bentler correction (Mplus variant)
## 
## Model Test Baseline Model:
##                               Standard    Scaled
##   Test statistic            40235.911 33410.871
##   Degrees of freedom           253      253
##   P-value                      0.000      0.000
##   Scaling correction factor                  1.204
## 
## User Model versus Baseline Model:
##                               Standard    Scaled
##   Comparative Fit Index (CFI)      0.610      0.629
##   Tucker-Lewis Index (TLI)        0.570      0.590
## 
##   Robust Comparative Fit Index (CFI)      0.611
##   Robust Tucker-Lewis Index (TLI)        0.571

```

```

## 
## Loglikelihood and Information Criteria:
## 
##   Loglikelihood user model (H0)           36194.191  36194.191
##   Scaling correction factor                  1.598
##       for the MLR correction
##   Loglikelihood unrestricted model (H1)     44097.862  44097.862
##   Scaling correction factor                  1.319
##       for the MLR correction
## 
##   Akaike (AIC)                            -72294.383 -72294.383
##   Bayesian (BIC)                           -72076.125 -72076.125
##   Sample-size adjusted Bayesian (SABIC)    -72225.371 -72225.371
## 
## Root Mean Square Error of Approximation:
## 
##   RMSEA                                0.298      0.264
##   90 Percent confidence interval - lower  0.294      0.261
##   90 Percent confidence interval - upper  0.302      0.268
##   P-value H_0: RMSEA <= 0.050            0.000      0.000
##   P-value H_0: RMSEA >= 0.080            1.000      1.000
## 
##   Robust RMSEA                            0.297
##   90 Percent confidence interval - lower  0.293
##   90 Percent confidence interval - upper  0.301
##   P-value H_0: Robust RMSEA <= 0.050    0.000
##   P-value H_0: Robust RMSEA >= 0.080    1.000
## 
## Standardized Root Mean Square Residual:
## 
##   SRMR                                 0.079      0.079
## 
## Parameter Estimates:
## 
##   Standard errors                      Sandwich
##   Information bread                   Observed
##   Observed information based on       Hessian
## 
## Latent Variables:
## 
##             Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## AltaV =~
##   lembro        1.000
##   chique        1.248  0.071 17.458  0.000  0.048  0.782
##   charmosa      1.240  0.068 18.192  0.000  0.059  0.796
##   dancando      1.209  0.054 22.320  0.000  0.058  0.812
##   maravilha     1.243  0.047 26.217  0.000  0.059  0.952
##   louco         1.146  0.050 23.046  0.000  0.055  0.879
##   delicia       1.214  0.046 26.639  0.000  0.058  0.911
##   linda          1.343  0.045 30.131  0.000  0.064  0.944
##   gosto          1.155  0.037 30.946  0.000  0.055  0.928
##   maravilhosa    1.358  0.049 27.952  0.000  0.065  0.928
##   gostei         1.180  0.041 29.047  0.000  0.056  0.881
##   amo            1.293  0.042 31.016  0.000  0.062  0.904
##   top            1.054  0.050 20.938  0.000  0.050  0.833

```

```

##      vibe          1.214    0.034   35.380    0.000    0.058    0.936
##  BaixaV =~
##      viciada       1.000
##      merda         0.989    0.033   29.986    0.000    0.055    0.959
##      lixo          0.931    0.035   26.321    0.000    0.052    0.910
##      foda          0.991    0.029   34.005    0.000    0.055    0.963
##      ruim          0.892    0.028   32.045    0.000    0.050    0.897
##      pedrada        0.946    0.029   33.007    0.000    0.053    0.955
##      inferno        0.884    0.032   27.698    0.000    0.049    0.818
##      recordacoes    0.875    0.034   26.099    0.000    0.049    0.780
##      pancada        0.878    0.026   34.054    0.000    0.049    0.925
##
## Covariances:
##              Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##  AltaV ~~
##  BaixaV     0.002    0.000   8.313    0.000    0.829    0.829
##
## Variances:
##              Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##  .lembro      0.001    0.000  15.624    0.000    0.001    0.389
##  .chique       0.002    0.000   9.309    0.000    0.002    0.366
##  .charmosa     0.002    0.000   9.140    0.000    0.002    0.341
##  .dancando     0.001    0.000   8.442    0.000    0.001    0.227
##  .maravilha    0.000    0.000   6.286    0.000    0.000    0.094
##  .louco         0.001    0.000   8.379    0.000    0.001    0.227
##  .delicia      0.001    0.000  13.815    0.000    0.001    0.169
##  .linda         0.000    0.000   9.054    0.000    0.000    0.109
##  .gosto         0.000    0.000  12.725    0.000    0.000    0.139
##  .maravilhosa   0.001    0.000   9.896    0.000    0.001    0.139
##  .gostei        0.001    0.000  14.603    0.000    0.001    0.224
##  .amo           0.001    0.000  12.124    0.000    0.001    0.183
##  .top           0.001    0.000  14.840    0.000    0.001    0.306
##  .vibe          0.000    0.000  11.382    0.000    0.000    0.125
##  .viciada       0.001    0.000  15.545    0.000    0.001    0.295
##  .merda          0.000    0.000  10.994    0.000    0.000    0.080
##  .lixo           0.001    0.000  13.254    0.000    0.001    0.172
##  .foda           0.000    0.000   9.822    0.000    0.000    0.073
##  .ruim           0.001    0.000  15.552    0.000    0.001    0.195
##  .pedrada        0.000    0.000  11.822    0.000    0.000    0.088
##  .inferno        0.001    0.000  14.915    0.000    0.001    0.331
##  .recordacoes    0.002    0.000  15.362    0.000    0.002    0.392
##  .pancada        0.000    0.000  12.818    0.000    0.000    0.144
##  AltaV          0.002    0.000   9.033    0.000    1.000    1.000
##  BaixaV         0.003    0.000   9.506    0.000    1.000    1.000

```

```
semTools:::compRelSEM(fit_semantico)
```

```

##  AltaV BaixaV
##  0.978  0.965

```

### 3 Estimativas com PANAS

```
load("data.RData")
panas_data <- as.data.frame(data[97:116])
str(panas_data)

## 'data.frame':   457 obs. of  20 variables:
## $ PN1ativo : num  5 4 4 5 4 4 4 2 5 4 ...
## $ PN2envergo : num  1 1 4 3 2 2 2 1 2 2 ...
## $ PN3atento : num  4 4 4 4 4 3 4 4 4 3 ...
## $ PN4aflit  : num  2 5 3 4 5 3 2 2 2 4 ...
## $ PN5determ : num  4 3 4 4 4 4 4 3 5 4 ...
## $ PN6culpado: num  1 4 4 2 2 3 2 1 2 2 ...
## $ PN7empol  : num  2 2 4 2 4 3 4 3 5 5 ...
## $ PN8irrit  : num  1 4 2 5 4 3 2 1 3 4 ...
## $ PN9interes: num  5 2 4 1 5 4 4 3 5 5 ...
## $ PN10medo  : num  1 4 1 5 4 1 3 2 2 4 ...
## $ PN11orgul : num  4 1 5 1 2 4 3 4 5 4 ...
## $ PN12hostil: num  1 4 4 5 1 3 3 1 2 2 ...
## $ PN13alerta : num  4 4 4 5 3 4 3 1 5 3 ...
## $ PN14inquie: num  4 4 3 5 4 4 3 2 3 3 ...
## $ PN15entusia: num  4 1 4 1 4 4 4 3 5 4 ...
## $ PN16nervo  : num  1 4 3 5 4 4 2 1 3 4 ...
## $ PN17forte  : num  4 1 5 1 5 3 3 4 5 4 ...
## $ PN18apavo : num  1 4 4 3 3 1 2 1 2 2 ...
## $ PN19inspi : num  4 1 5 1 5 3 4 5 5 4 ...
## $ PN20chate : num  1 4 4 5 2 4 2 1 2 2 ...
```

#### 3.1 Fatorabilidade

```
cat("--- Teste de Esfericidade de Bartlett (Correto) ---\n")

## --- Teste de Esfericidade de Bartlett (Correto) ---

bartlett_results_correto <- cortest.bartlett(panas_data)

## R was not square, finding R from data

print(bartlett_results_correto)

## $chisq
## [1] 4551.737
##
## $p.value
## [1] 0
##
## $df
## [1] 190
```

```

cat("\n--- Medida de Adequação da Amostra (KMO) ---\n")

## 
## --- Medida de Adequação da Amostra (KMO) ---

kmo_results_correto <- KMO(panas_data)
print(kmo_results_correto)

## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = panas_data)
## Overall MSA = 0.9
## MSA for each item =
##   PN1ativo  PN2envergo  PN3atento  PN4aflit  PN5determ  PN6culpado
##   0.91      0.91       0.87      0.89      0.93       0.89
##   PN7empol  PN8irrit   PN9interes  PN10medo  PN11orgul  PN12hostil
##   0.91      0.91       0.92      0.91      0.93       0.88
##   PN13alerta  PN14inquie  PN15entusia  PN16nervo  PN17forte  PN18apavo
##   0.86      0.91       0.91      0.87      0.91       0.90
##   PN19inspi  PN20chate
##   0.91      0.92

```

## 3.2 Análise Paralela de Horn

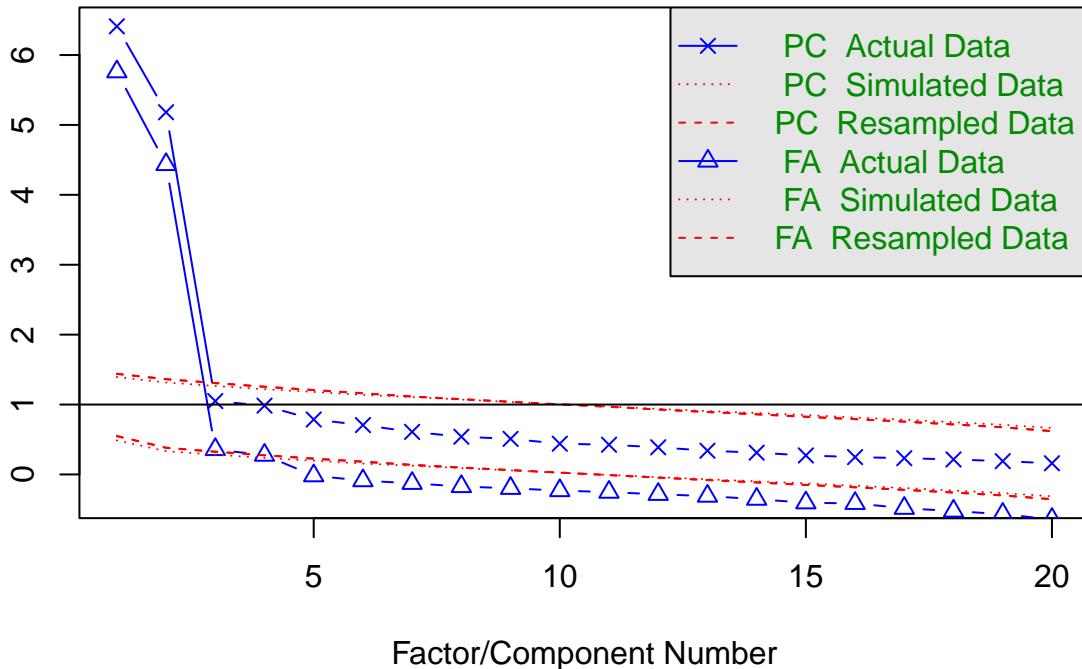
```

parallel_analysis_results <- fa.parallel(
  panas_data,
  fa = "both",
  n.iter = 100,
  show.legend = TRUE,
  cor="poly",
  main = "Análise Paralela de Horn"
)

## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect. Try a
## different factor score estimation method.

```

## Análise Paralela de Horn



```
## Parallel analysis suggests that the number of factors = 4 and the number of components = 2
```

### 3.2.1 Plot da AP

```
# Dataframe para o gráfico da PCA
df_plot_pca <- data.frame(
  Numero = 1:length(parallel_analysis_results$pc.values),
  Autovalor_Real = parallel_analysis_results$pc.values,
  Autovalor_Simulado = parallel_analysis_results$pc.sim
)

# Dataframe para o gráfico da AFE
df_plot_afe <- data.frame(
  Numero = 1:length(parallel_analysis_results$fa.values),
  Autovalor_Real = parallel_analysis_results$fa.values,
  Autovalor_Simulado = parallel_analysis_results$fa.sim
)

# --- PASSO 2: Criar o gráfico da PCA ---
plot_pca <- ggplot(df_plot_pca, aes(x = Numero)) +
  geom_line(aes(y = Autovalor_Real, color = "Dados Reais (PCA)", size = 0.7)) +
  geom_point(aes(y = Autovalor_Real, color = "Dados Reais (PCA)", shape = 17, size = 3)) +
  geom_line(aes(y = Autovalor_Simulado, color = "Dados Simulados (PCA)", linetype = "dashed", size = 0.7))
```

```

geom_hline(yintercept = 1, linetype = "dotted", color = "black") +
annotate("text", x = max(df_plot_pca$Numero) * 0.9, y = 1.3, label = "", size = 3) +
scale_color_manual(name = "Análise", values = c("Dados Reais (PCA)" = "blue", "Dados Simulados (PCA)" =
labs(
  title = "Análise Paralela (PCA)",
  x = "Número do Componente",
  y = "Autovalor (Eigenvalue)"
) +
theme_minimal(base_size = 12) +
theme(legend.position = "top", plot.title = element_text(hjust = 0.5, face = "bold")) +
scale_x_continuous(breaks = seq(0, 45, by = 5))

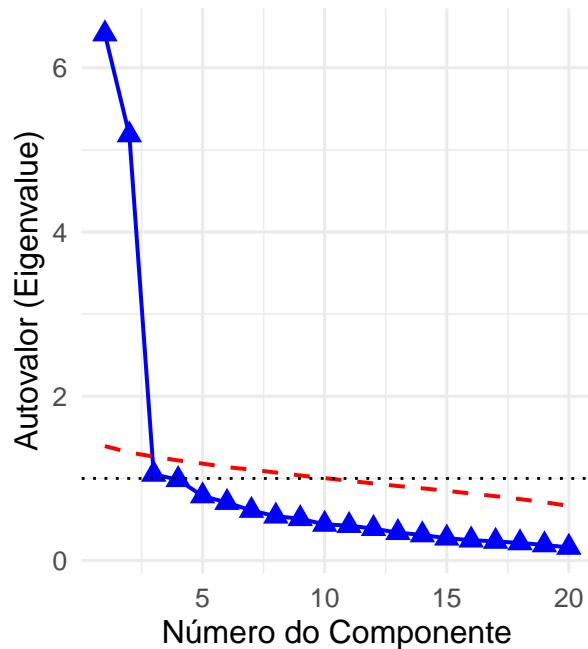
# --- PASSO 3: Criar o gráfico da AFE ---
plot_afe <- ggplot(df_plot_afe, aes(x = Numero)) +
  geom_line(aes(y = Autovalor_Real, color = "Dados Reais (AFE)", size = 0.7) +
  geom_point(aes(y = Autovalor_Real, color = "Dados Reais (AFE)", shape = 17, size = 3) +
  geom_line(aes(y = Autovalor_Simulado, color = "Dados Simulados (AFE)", linetype = "dashed", size = 0.7) +
  geom_hline(yintercept = 1, linetype = "dotted", color = "black") +
  annotate("text", x = max(df_plot_afe$Numero) * 0.9, y = 1.3, label = "", size = 3) +
  scale_color_manual(name = "Análise", values = c("Dados Reais (AFE)" = "darkgreen", "Dados Simulados (AFE)" = "red"),
  labs(
    title = "Análise Paralela (AFE)",
    x = "Número do Fator",
    y = "Autovalor (Eigenvalue)"
) +
  theme_minimal(base_size = 12) +
  theme(legend.position = "top", plot.title = element_text(hjust = 0.5, face = "bold")) +
  scale_x_continuous(breaks = seq(0, 45, by = 5))

# --- PASSO 4: Combinar os dois gráficos ---
# O operador '+' do patchwork une os gráficos lado a lado
plot_pca + plot_afe

```

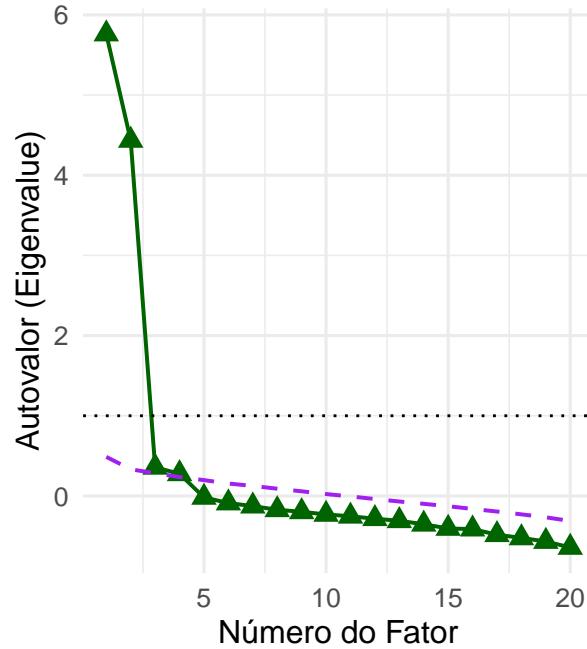
## Análise Paralela (PCA)

álise Dados Reais (PCA) - - - Dados Simulados (PCA)



## Análise Paralela (AFE)

álise Dados Reais (AFE) - - - Dados Simulados (AFE)



### PCA do PANAS

O primeiro componente parece ter o eixo valência também enquanto o segundo parece ter o Intensidade/Saliência, com todas as palavras carregando positivamente

```
cat("\n--- Rodando PCA para extrair 2 componentes ---\n")
```

```
##  
## --- Rodando PCA para extrair 2 componentes ---
```

```
pca_results_psych <- principal(  
  r = panas_data,  
  nfactors = 2,  
  rotate = "none"  
)  
  
print(pca_results_psych$loadings, cutoff = 0.30, sort = TRUE)
```

```
##  
## Loadings:  
##          PC1    PC2  
## PN1ativo   0.637  0.363  
## PN3atento   0.592  0.405  
## PN5determ   0.653  0.415  
## PN6culpado -0.503  0.474  
## PN7empol    0.695  0.384
```

```

## PN9interes  0.596  0.494
## PN11orgul   0.668  0.380
## PN15entusia  0.692  0.426
## PN17forte    0.574  0.432
## PN18apavo   -0.580  0.407
## PN19inspi    0.610  0.378
## PN4aflit     -0.527  0.570
## PN8irrit     -0.366  0.619
## PN10medo     -0.496  0.590
## PN13alerta    0.560
## PN14inquie   -0.373  0.599
## PN16nervo    -0.457  0.699
## PN20chate    -0.540  0.550
## PN2envergo   -0.345  0.412
## PN12hostil   -0.325  0.482
##
##                  PC1    PC2
## SS loadings      5.791  4.826
## Proportion Var  0.290  0.241
## Cumulative Var  0.290  0.531

```

```
pca_results_psych$fit.off
```

```
## [1] 0.9696769
```

### 3.2.2 Plot

```

# Prepara o dataframe para o plot
loadings_df <- as.data.frame(unclass(pca_results_psych$loadings))
loadings_df$palavra <- rownames(loadings_df)

mapa_nomes <- c(
  "PN1ativo"      = "Ativo", "PN2envergo"    = "Envergonhado", "PN3atento"    = "Atento",
  "PN4aflit"       = "Aflito", "PN5determ"     = "Determinado", "PN6culpado"   = "Culpado",
  "PN7empol"       = "Empolgado", "PN8irrit"     = "Irritado", "PN9interes"   = "Interessado",
  "PN10medo"       = "Medo", "PN11orgul"     = "Orgulhoso", "PN12hostil"   = "Hostil",
  "PN13alerta"     = "Alerta", "PN14inquie"   = "Inquieto", "PN15entusia" = "Entusiasmado",
  "PN16nervo"       = "Nervoso", "PN17forte"    = "Forte", "PN18apavo"    = "Apavorado",
  "PN19inspi"       = "Inspirado", "PN20chate"   = "Chateado"
)
loadings_df$palavra <- mapa_nomes[loadings_df$palavra]

# --- INÍCIO DO CÓDIGO DO GRÁFICO MODIFICADO ---

grafico_panas_2D_simples <- ggplot(
  data = loadings_df,
  # MUDANÇA 1: Removemos 'size' do mapeamento estético (aes).
  aes(x = PC1, y = PC2, label = palavra)
) +

```

```

# Linhas de referência dos quadrantes (centro em 0,0)
geom_hline(yintercept = 0, linetype = "dashed", color = "gray50") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "gray50") +


# Camada de texto com repulsão.
# MUDANÇA 2: Definimos um tamanho FIXO para todas as palavras.
geom_text_repel(
  fontface = "bold",
  color = "black",
  bg.color = "white",
  bg.r = 0.1,
  segment.color = "transparent",
  max.overlaps = Inf,
  size = 4 # Todas as palavras terão este tamanho de fonte.
) +


# MUDANÇA 3: A linha 'scale_size_continuous(...)' foi completamente REMOVIDA.
# Ela não é mais necessária e causaria um erro.

# MUDANÇA 4: Títulos e eixos atualizados para refletir a nova visualização.
labs(
  title = "Mapa Afetivo da PANAS",
  subtitle = "Estrutura Bidimensional: Valência vs. Intensidade",
  x = "Valência (PC1)",
  y = "Intensidade (PC2)"
) +

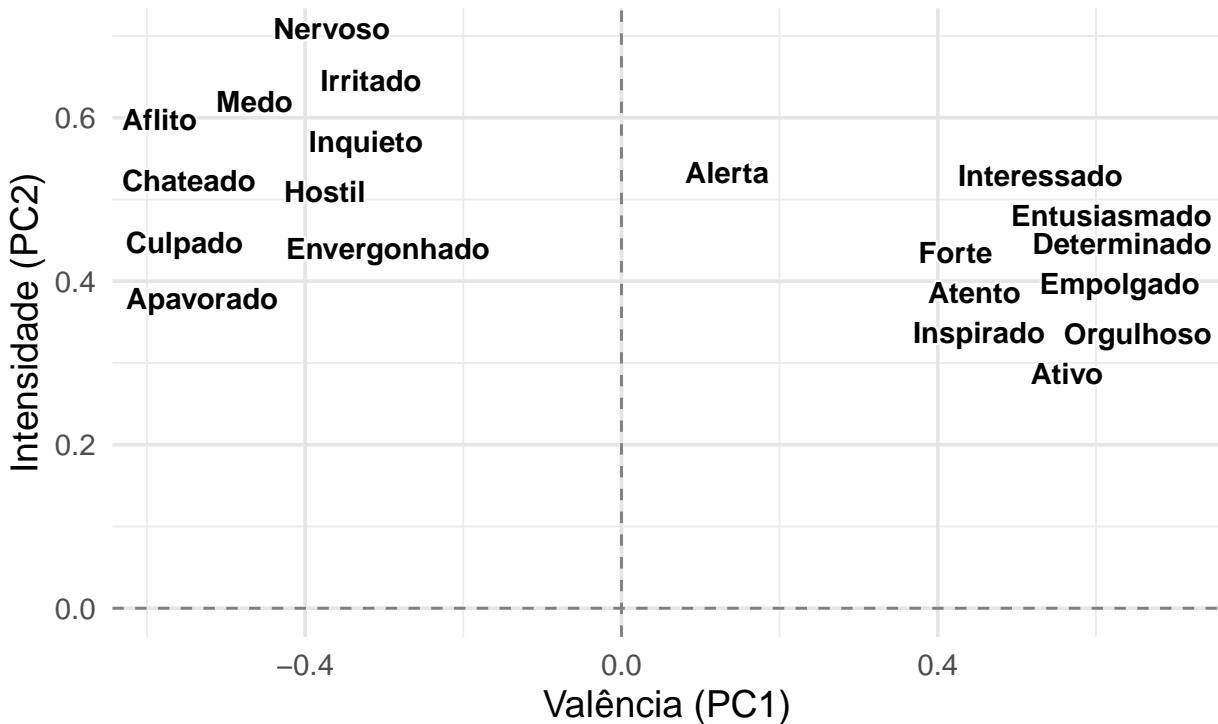

# Tema visual limpo
theme_minimal(base_size = 14) +
  theme(
    panel.grid.major = element_line(color = "gray90"),
    plot.title = element_text(hjust = 0.5, face = "bold", size = 18),
    plot.subtitle = element_text(hjust = 0.5, size = 11),
    # A legenda desaparece automaticamente porque não há mais estéticas para legendar.
    legend.position = "none"
  ) +


# Exibir o gráfico final
print(grafico_panas_2D_simples)

```

## Mapa Afetivo da PANAS

Estrutura Bidimensional: Valência vs. Intensidade



### 3.2.3 Com rotação para extrair os componentes

```
cat("\n--- Rodando PCA para extrair 2 componentes ---\n")

## --- Rodando PCA para extrair 2 componentes ---

pca_results_psych <- principal(
  r = panas_data,
  nfactors = 2,
  rotate = "varimax" # Sem rotação para ver a estrutura bruta
)

# --- PASSO 2: Analisar a Matriz de Cargas ---
print(pca_results_psych$loadings, cutoff = 0.30, sort = TRUE)

## Loadings:
##          RC1     RC2
## PN1ativo  0.727
## PN3atento  0.717
## PN5determ  0.771
## PN7empol   0.786
```

```

## PN9interes 0.773
## PN11orgul 0.762
## PN15entusia 0.809
## PN17forte 0.718
## PN19inspi 0.715
## PN2envergo 0.537
## PN4aflit 0.772
## PN6culpado 0.681
## PN8irrit 0.715
## PN10medo 0.769
## PN12hostil 0.581
## PN14inquie 0.703
## PN16nervo 0.833
## PN18apavo 0.673
## PN20chate 0.764
## PN13alerta 0.483 0.338
##
##          RC1   RC2
## SS loadings 5.444 5.173
## Proportion Var 0.272 0.259
## Cumulative Var 0.272 0.531

```

```
pca_results_psych$fit.off
```

```
## [1] 0.9696769
```

```

loadings_df <- as.data.frame(unclass(pca_results_psych$loadings))
loadings_df$palavra <- rownames(loadings_df)

cat("\n--- Gerando o mapa afetivo com os Componentes 1 e 3 ---\n")

```

```

## 
## --- Gerando o mapa afetivo com os Componentes 1 e 3 ---

# Extrair a variância explicada de forma correta
variancia_exPLICADA_pca <- pca_results_psych$Vaccounted["Proportion Var",]

```

### 3.3 Formativo vs reflexivo - PANAS

```

# 1. Definir o modelo de mensuração de primeira ordem (AP e AN)
mm_panas_1a_ordem <- constructs(
  composite("AP", c("PN1ativo", "PN3atento", "PN5determ", "PN7empol",
    "PN9interes", "PN11orgul", "PN13alerta", "PN15entusia",
    "PN17forte", "PN19inspi"),
    weights = mode_A),
  composite("AN", c("PN2envergo", "PN4aflit", "PN6culpado", "PN8irrit",
    "PN10medo", "PN12hostil", "PN14inquie", "PN16nervo",
    "PN18apavo", "PN20chate"),
    weights = mode_A)

```

```

)

# 2. Criar modelo estrutural mínimo
sm_panas_1a_ordem <- relationships(
  paths(from = "AP", to = "AN")
)

# 3. Estimar e obter escores
pls_panas_1a_ordem <- estimate_pls(
  data = panas_data,
  measurement_model = mm_panas_1a_ordem,
  structural_model = sm_panas_1a_ordem
)

## Generating the seminr model

## All 457 observations are valid.

# Se rodou sem erro, podemos continuar
summary(pls_panas_1a_ordem)

## 
## Results from package seminr (2.3.7)
##
## Path Coefficients:
##           AN
## R^2      0.068
## AdjR^2   0.066
## AP       -0.261
##
## Reliability:
##   alpha rhoC  AVE rhoA
## AP    0.901 0.894 0.477 0.787
## AN    0.888 0.906 0.494 0.933
##
## Alpha, rhoC, and rhoA should exceed 0.7 while AVE should exceed 0.5

# O resto do código da Etapa 1 e Etapa 2 deve funcionar agora.
panas_scores <- pls_panas_1a_ordem$construct_scores
panas_com_escores <- cbind(panas_data, panas_scores)
panas_com_escores <- as.data.frame(panas_com_escores)

```

### 3.3.1 Formativo - PANAS

```

mm_panas_formativo <- constructs(
  # Primeiro, defina os construtos de primeira ordem que servirão de indicadores
  # Eles são compostos de itens únicos, que são os escores que já calculamos
  composite("AP_first_order", "AP", weights = mode_A),
  composite("AN_first_order", "AN", weights = mode_A),

```

```

# Defina também a sua âncora
composite("Ancora_Ativacao", c("PN7empol", "PN15entusia", "PN8irrit", "PN16nervo"), weights = mode_A)

# AGORA, defina o construto de segunda ordem FORMATIVO, referenciando os NOMES DOS CONSTRUTOS acima
composite("G_Factor_Formativo", c("AP", "AN"), weights = mode_B)
)

# 2. Definir o modelo estrutural (continua o mesmo)
sm_panas_formativo <- relationships(
  paths(from = "G_Factor_Formativo", to = "Ancora_Ativacao")
)

# 3. Estimar o modelo final com o novo mm_panas_formativo
pls_panas_final <- estimate_pls(
  data = panas_com_escores,
  measurement_model = mm_panas_formativo,
  structural_model = sm_panas_formativo
)

```

## Generating the seminr model

## All 457 observations are valid.

```

# 4. Analisar os resultados!
summary_panas_final <- summary(pls_panas_final)

print(summary_panas_final$validity$vif_items)

```

## G\_Factor\_Formativo :

	AP	AN		
##	1.073	1.073		
##				
## Ancora_Ativacao :				
##	PN7empol	PN15entusia	PN8irrit	PN16nervo
##	1.880	1.872	1.623	1.632

```

# 5. Bootstrapping para significância
boot_panas <- bootstrap_model(pls_panas_final, nboot = 100)

```

## Bootstrapping model using seminr...

## SEMinR Model successfully bootstrapped

```

summary(boot_panas)$bootstrapped_weights

```

	Original	Est.	Bootstrap Mean	Bootstrap SD
## PN7empol -> Ancora_Ativacao	0.539	0.531	0.030	
## PN15entusia -> Ancora_Ativacao	0.512	0.504	0.028	
## PN8irrit -> Ancora_Ativacao	-0.091	-0.044	0.118	
## PN16nervo -> Ancora_Ativacao	-0.153	-0.095	0.135	

		0.954	0.955	0.066
## AP -> G_Factor_Formativo		-0.141	-0.048	0.227
##	T Stat.	2.5%	CI 97.5%	CI
## PN7empol -> Ancora_Ativacao	18.082	0.456	0.568	
## PN15entusia -> Ancora_Ativacao	18.102	0.438	0.536	
## PN8irrit -> Ancora_Ativacao	-0.769	-0.226	0.198	
## PN16nervo -> Ancora_Ativacao	-1.136	-0.287	0.182	
## AP -> G_Factor_Formativo	14.509	0.814	1.030	
## AN -> G_Factor_Formativo	-0.622	-0.374	0.433	

### 3.3.2 Reflexivo - PANAS

```

mm_panas_formativo <- constructs(
  # Primeiro, defina os construtos de primeira ordem que servirão de indicadores
  # Eles são compostos de itens únicos, que são os escores que já calculamos
  composite("AP_first_order", "AP", weights = mode_A),
  composite("AN_first_order", "AN", weights = mode_A),

  # Defina também a sua âncora
  composite("Ancora_Ativacao", c("PN7empol", "PN15entusia", "PN8irrit", "PN16nervo"), weights = mode_A)

  # AGORA, defina o construto de segunda ordem FORMATIVO, referenciando os NOMES DOS CONSTRUTOS acima
  composite("G_Factor_Formativo", c("AP", "AN"), weights = mode_A)
)

# 2. Definir o modelo estrutural (continua o mesmo)
sm_panas_formativo <- relationships(
  paths(from = "G_Factor_Formativo", to = "Ancora_Ativacao")
)

# 3. Estimar o modelo final com o novo mm_panas_formativo
pls_panas_final <- estimate_pls(
  data = panas_com_escores,
  measurement_model = mm_panas_formativo,
  structural_model = sm_panas_formativo
)

## Generating the seminr model

## All 457 observations are valid.

# 4. Analisar os resultados!
summary_panas_final <- summary(pls_panas_final)

print(summary_panas_final$validity$vif_items)

## G_Factor_Formativo :
##   AP     AN
## 1 1.073 1.073
##
## Ancora_Ativacao :

```

```

##      PN7empol PN15entusia    PN8irrit    PN16nervo
##          1.880        1.872        1.623        1.632

# 5. Bootstrapping para significância
boot_panas <- bootstrap_model(pls_panas_final, nboot = 100)

## Bootstrapping model using seminr...

## SEMinR Model successfully bootstrapped

summary(boot_panas)$bootstrapped_weights

##                                     Original Est. Bootstrap Mean Bootstrap SD
## PN7empol  -> Ancora_Ativacao      0.463      0.472      0.035
## PN15entusia -> Ancora_Ativacao     0.436      0.444      0.034
## PN8irrit   -> Ancora_Ativacao     -0.250     -0.228      0.060
## PN16nervo   -> Ancora_Ativacao     -0.335     -0.313      0.062
## AP         -> G_Factor_Formativo    0.726      0.749      0.067
## AN         -> G_Factor_Formativo    -0.524     -0.490      0.092
##                                     T Stat. 2.5% CI 97.5% CI
## PN7empol  -> Ancora_Ativacao     13.263     0.415      0.548
## PN15entusia -> Ancora_Ativacao    12.950     0.391      0.517
## PN8irrit   -> Ancora_Ativacao    -4.167     -0.302     -0.059
## PN16nervo   -> Ancora_Ativacao    -5.420     -0.380     -0.139
## AP         -> G_Factor_Formativo   10.804     0.671      0.931
## AN         -> G_Factor_Formativo   -5.706     -0.566     -0.217

```

### 3.3.3 Reflexivo com lavaan

O modelo teórico não funciona no BR

```

model_syntax <- '
AP =~ PN1ativo + PN3atento + PN5determ + PN7empol + PN9interes + PN11orgul +
PN13alerta + PN15entusia + PN17forte + PN19inspi

AN =~ PN2envergo + PN4aflit + PN6culpado + PN8irrit + PN10medo + PN12hostil + PN14inquie +
+ PN16nervo + PN18apavo + PN20chate

fit_panas <- cfa(model_syntax, data = panas_data, ordered = T,
estimator = "WLSMV", std.lv=TRUE)

fitmeasures(fit_panas, c("chisq", "df", "pvalue", "cfi", "rmsea",
"rmsea.ci.lower","rmsea.ci.upper"))

##           chisq       df      pvalue      cfi      rmsea
##          953.755    169.000      0.000      0.970      0.101
## rmsea.ci.lower rmsea.ci.upper
##          0.095      0.107

```

```
# Ver os resultados resumidos
#summary(fit_panas, fit.measures = TRUE, standardized = TRUE)
semTools:::compRelSEM(fit_panas)
```

```
##      AP      AN
## 0.913 0.911
```

### 3.3.4 Ajuste no modelo teórico

A literatura aponta que alerta carrega nos dois fatores

```
model_syntax <- '
AP =~ PN1ativo + PN3atento + PN5determ + PN7empol + PN9interes + PN11orgul +
PN13alerta + PN15entusia + PN17forte + PN19inspi
AN =~ PN2envergo + PN4aflit + PN6culpado + PN8irrit + PN10medo + PN12hostil + PN14inquie +
+ PN16nervo + PN18apavo + PN20chate + PN13alerta
'

fit_panas <- cfa(model_syntax, data = panas_data, ordered = T,
estimator = "WLSMV", std.lv=TRUE)

fitmeasures(fit_panas, c("chisq", "df", "pvalue", "cfi", "rmsea",
"rmsea.ci.lower", "rmsea.ci.upper"))

##          chisq        df     pvalue      cfi      rmsea
## 638.313    168.000     0.000    0.982    0.078
##  rmsea.ci.lower  rmsea.ci.upper
## 0.072       0.085

# Ver os resultados resumidos
summary(fit_panas, fit.measures = TRUE, standardized = TRUE)

## lavaan 0.6-20 ended normally after 23 iterations
##
##   Estimator           DWLS
##   Optimization method NLMINB
##   Number of model parameters      102
## 
##   Number of observations        457
## 
## Model Test User Model:
##                               Standard     Scaled
##   Test Statistic            638.313   682.235
##   Degrees of freedom         168        168
##   P-value (Chi-square)      0.000     0.000
##   Scaling correction factor          1.058
##   Shift parameter             78.647
```

```

##      simple second-order correction
##
## Model Test Baseline Model:
##
##      Test statistic          26632.140    8829.609
##      Degrees of freedom       190         190
##      P-value                  0.000      0.000
##      Scaling correction factor           3.061
##
## User Model versus Baseline Model:
##
##      Comparative Fit Index (CFI)        0.982    0.940
##      Tucker-Lewis Index (TLI)          0.980    0.933
##
##      Robust Comparative Fit Index (CFI)        0.845
##      Robust Tucker-Lewis Index (TLI)          0.825
##
## Root Mean Square Error of Approximation:
##
##      RMSEA                      0.078    0.082
##      90 Percent confidence interval - lower   0.072    0.076
##      90 Percent confidence interval - upper   0.085    0.088
##      P-value H_0: RMSEA <= 0.050            0.000    0.000
##      P-value H_0: RMSEA >= 0.080            0.344    0.697
##
##      Robust RMSEA                  0.108
##      90 Percent confidence interval - lower   0.101
##      90 Percent confidence interval - upper   0.115
##      P-value H_0: Robust RMSEA <= 0.050      0.000
##      P-value H_0: Robust RMSEA >= 0.080      1.000
##
## Standardized Root Mean Square Residual:
##
##      SRMR                      0.072    0.072
##
## Parameter Estimates:
##
##      Parameterization             Delta
##      Standard errors              Robust.sem
##      Information                  Expected
##      Information saturated (h1) model Unstructured
##
## Latent Variables:
##      Estimate  Std.Err  z-value  P(>|z|)  Std.lv  Std.all
##      AP =~
##      PN1ativo      0.738    0.025  29.384  0.000    0.738    0.738
##      PN3atento     0.727    0.025  29.429  0.000    0.727    0.727
##      PN5determ     0.787    0.023  33.952  0.000    0.787    0.787
##      PN7empol      0.828    0.016  51.124  0.000    0.828    0.828
##      PN9interes    0.790    0.021  37.248  0.000    0.790    0.790
##      PN11orgul     0.774    0.019  40.463  0.000    0.774    0.774
##      PN13alerta    0.479    0.034  14.015  0.000    0.479    0.479
##      PN15entusia   0.840    0.015  55.769  0.000    0.840    0.840
##      PN17forte     0.722    0.023  31.430  0.000    0.722    0.722

```

```

##   PN19inspi      0.726    0.022   33.144    0.000    0.726    0.726
## AN =~
##   PN2envergo     0.524    0.036   14.629    0.000    0.524    0.524
##   PN4aflit       0.790    0.019   40.603    0.000    0.790    0.790
##   PN6culpado     0.690    0.029   23.874    0.000    0.690    0.690
##   PN8irrit        0.721    0.024   29.952    0.000    0.721    0.721
##   PN10medo       0.781    0.022   36.113    0.000    0.781    0.781
##   PN12hostil      0.576    0.034   16.874    0.000    0.576    0.576
##   PN14inquie      0.687    0.027   25.636    0.000    0.687    0.687
##   PN16nervo        0.848    0.017   49.936    0.000    0.848    0.848
##   PN18apavo       0.727    0.026   27.686    0.000    0.727    0.727
##   PN20chate       0.785    0.021   37.766    0.000    0.785    0.785
##   PN13alerta      0.361    0.038    9.630    0.000    0.361    0.361
##
## Covariances:
##   AP ~~
##   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   AN      -0.144  0.048 -2.983  0.003 -0.144 -0.144
##
## Thresholds:
##   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   PN1ativo|t1   -1.872  0.117 -16.056  0.000 -1.872 -1.872
##   PN1ativo|t2   -1.399  0.085 -16.425  0.000 -1.399 -1.399
##   PN1ativo|t3   -0.885  0.068 -13.038  0.000 -0.885 -0.885
##   PN1ativo|t4    0.488  0.061  7.955  0.000  0.488  0.488
##   PN3atento|t1  -1.904  0.120 -15.923  0.000 -1.904 -1.904
##   PN3atento|t2  -1.399  0.085 -16.425  0.000 -1.399 -1.399
##   PN3atento|t3  -0.829  0.067 -12.441  0.000 -0.829 -0.829
##   PN3atento|t4   0.732  0.065  11.305  0.000  0.732  0.732
##   PN5determ|t1  -1.939  0.123 -15.768  0.000 -1.939 -1.939
##   PN5determ|t2  -1.303  0.081 -16.102  0.000 -1.303 -1.303
##   PN5determ|t3  -0.622  0.063 -9.872  0.000 -0.622 -0.622
##   PN5determ|t4   0.662  0.064  10.413  0.000  0.662  0.662
##   PN7empol|t1   -1.939  0.123 -15.768  0.000 -1.939 -1.939
##   PN7empol|t2   -1.266  0.079 -15.936  0.000 -1.266 -1.266
##   PN7empol|t3   -0.350  0.060 -5.829  0.000 -0.350 -0.350
##   PN7empol|t4    0.740  0.065  11.394  0.000  0.740  0.740
##   PN9interes|t1  -1.872  0.117 -16.056  0.000 -1.872 -1.872
##   PN9interes|t2  -1.444  0.087 -16.529  0.000 -1.444 -1.444
##   PN9interes|t3  -0.732  0.065 -11.305  0.000 -0.732 -0.732
##   PN9interes|t4   0.622  0.063  9.872  0.000  0.622  0.622
##   PN11orgul|t1  -1.544  0.093 -16.650  0.000 -1.544 -1.544
##   PN11orgul|t2  -0.885  0.068 -13.038  0.000 -0.885 -0.885
##   PN11orgul|t3  -0.140  0.059 -2.382  0.017 -0.140 -0.140
##   PN11orgul|t4   0.869  0.067  12.869  0.000  0.869  0.869
##   PN13alerta|t1  -1.399  0.085 -16.425  0.000 -1.399 -1.399
##   PN13alerta|t2  -0.943  0.069 -13.619  0.000 -0.943 -0.943
##   PN13alerta|t3  -0.235  0.059 -3.968  0.000 -0.235 -0.235
##   PN13alerta|t4   0.987  0.070  14.024  0.000  0.987  0.987
##   PN15entusia|t1 -1.733  0.105 -16.481  0.000 -1.733 -1.733
##   PN15entusia|t2 -1.061  0.072 -14.646  0.000 -1.061 -1.061
##   PN15entusia|t3 -0.286  0.060 -4.806  0.000 -0.286 -0.286
##   PN15entusia|t4   0.837  0.067  12.527  0.000  0.837  0.837
##   PN17forte|t1   -1.509  0.091 -16.625  0.000 -1.509 -1.509

```

```

##   PN17forte|t2    -0.951    0.069   -13.701    0.000   -0.951   -0.951
##   PN17forte|t3    -0.151    0.059   -2.569     0.010   -0.151   -0.151
##   PN17forte|t4     0.918    0.069   13.372     0.000    0.918    0.918
##   PN19inspi|t1    -1.758    0.107   -16.422    0.000   -1.758   -1.758
##   PN19inspi|t2    -1.061    0.072   -14.646    0.000   -1.061   -1.061
##   PN19inspi|t3    -0.190    0.059   -3.222     0.001   -0.190   -0.190
##   PN19inspi|t4     0.747    0.065   11.482     0.000    0.747    0.747
##   PN2envergo|t1   -0.635    0.063   -10.053    0.000   -0.635   -0.635
##   PN2envergo|t2     0.074    0.059    1.262     0.207    0.074    0.074
##   PN2envergo|t3     0.814    0.066   12.269     0.000    0.814    0.814
##   PN2envergo|t4     2.016    0.131   15.377     0.000    2.016    2.016
##   PN4aflit|t1    -1.090    0.073   -14.870    0.000   -1.090   -1.090
##   PN4aflit|t2    -0.488    0.061   -7.955     0.000   -0.488   -0.488
##   PN4aflit|t3     0.030    0.059    0.514     0.607    0.030    0.030
##   PN4aflit|t4     1.141    0.075   15.229     0.000    1.141    1.141
##   PN6culpado|t1   -0.602    0.063   -9.600     0.000   -0.602   -0.602
##   PN6culpado|t2     0.008    0.059    0.140     0.889    0.008    0.008
##   PN6culpado|t3     0.690    0.064   10.772     0.000    0.690    0.690
##   PN6culpado|t4     1.621    0.097   16.641     0.000    1.621    1.621
##   PN8irrit|t1    -1.303    0.081   -16.102    0.000   -1.303   -1.303
##   PN8irrit|t2    -0.451    0.061   -7.402     0.000   -0.451   -0.451
##   PN8irrit|t3     0.163    0.059    2.756     0.006    0.163    0.163
##   PN8irrit|t4     1.356    0.083   16.299     0.000    1.356    1.356
##   PN10medo|t1   -0.877    0.068   -12.953    0.000   -0.877   -0.877
##   PN10medo|t2   -0.163    0.059   -2.756     0.006   -0.163   -0.163
##   PN10medo|t3     0.344    0.060    5.736     0.000    0.344    0.344
##   PN10medo|t4     1.184    0.076   15.501     0.000    1.184    1.184
##   PN12hostil|t1   -0.367    0.060   -6.107     0.000   -0.367   -0.367
##   PN12hostil|t2     0.275    0.060    4.620     0.000    0.275    0.275
##   PN12hostil|t3     1.090    0.073   14.870     0.000    1.090    1.090
##   PN12hostil|t4     1.976    0.127   15.588     0.000    1.976    1.976
##   PN14inquiet|t1   -1.329    0.082   -16.204    0.000   -1.329   -1.329
##   PN14inquiet|t2   -0.635    0.063   -10.053    0.000   -0.635   -0.635
##   PN14inquiet|t3   -0.052    0.059   -0.888     0.375   -0.052   -0.052
##   PN14inquiet|t4     0.960    0.070   13.782     0.000    0.960    0.960
##   PN16nervo|t1   -1.042    0.072   -14.494    0.000   -1.042   -1.042
##   PN16nervo|t2   -0.367    0.060   -6.107     0.000   -0.367   -0.367
##   PN16nervo|t3     0.213    0.059    3.595     0.000    0.213    0.213
##   PN16nervo|t4     1.195    0.077   15.567     0.000    1.195    1.195
##   PN18apavo|t1   -0.207    0.059   -3.502     0.000   -0.207   -0.207
##   PN18apavo|t2     0.421    0.061    6.940     0.000    0.421    0.421
##   PN18apavo|t3     1.120    0.074   15.088     0.000    1.120    1.120
##   PN18apavo|t4     1.758    0.107   16.422     0.000    1.758    1.758
##   PN20chate|t1   -0.877    0.068   -12.953    0.000   -0.877   -0.877
##   PN20chate|t2   -0.247    0.059   -4.155     0.000   -0.247   -0.247
##   PN20chate|t3     0.275    0.060    4.620     0.000    0.275    0.275
##   PN20chate|t4     1.241    0.078   15.818     0.000    1.241    1.241
##
## Variances:
##          Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## .PN1ativo      0.456
## .PN3atento      0.471
## .PN5determ      0.380
## .PN7empol       0.315

```

```

##   .PN9interes      0.376      0.376      0.376
##   .PN11orgul       0.401      0.401      0.401
##   .PN13alerta      0.690      0.690      0.690
##   .PN15entusia     0.294      0.294      0.294
##   .PN17forte       0.479      0.479      0.479
##   .PN19inspi        0.473      0.473      0.473
##   .PN2envergo      0.725      0.725      0.725
##   .PN4aflit        0.377      0.377      0.377
##   .PN6culpado      0.525      0.525      0.525
##   .PN8irrit         0.480      0.480      0.480
##   .PN10medo        0.390      0.390      0.390
##   .PN12hostil       0.668      0.668      0.668
##   .PN14inquie       0.528      0.528      0.528
##   .PN16nervo        0.280      0.280      0.280
##   .PN18apavo        0.471      0.471      0.471
##   .PN20chate        0.384      0.384      0.384
##   AP              1.000      1.000      1.000
##   AN              1.000      1.000      1.000

```

```
semTools:::compRelSEM(fit_panas)
```

```

##   AP      AN
## 0.934 0.922

```

### 3.3.5 Modelo bifatorial reflexivo se ajute melhor aos dados

Mas o modelo é bem sujo

```

bifactor_model_syntax <- '

G_Factor =~ PN1ativo + PN2envergo + PN3atento + PN4aflit + PN5determ +
PN6culpado + PN7empol + PN8irrit + PN9interes + PN10medo +
PN11orgul + PN12hostil + PN13alerta + PN14inquie +
PN15entusia + PN16nervo + PN17forte + PN18apavo +
PN19inspi + PN20chate

AP =~ PN1ativo + PN3atento + PN5determ + PN7empol + PN9interes + PN11orgul +
PN13alerta + PN15entusia + PN17forte + PN19inspi

AN =~ PN2envergo + PN4aflit + PN6culpado + PN8irrit + PN10medo + PN12hostil + PN14inquie +
+ PN16nervo + PN18apavo + PN20chate + PN13alerta
'

# Tentar rodar este novo modelo
fit_bifactor <- cfa(bifactor_model_syntax, data = panas_data, ordered = T, orthogonal = T,
estimator = "WLSMV", std.lv=TRUE)

fitmeasures(fit_bifactor, c("chisq", "df", "pvalue", "cfi", "rmsea",
"rmsea.ci.lower","rmsea.ci.upper"))

```

	chisq	df	pvalue	cfi	rmsea
##	294.436	149.000	0.000	0.994	0.046

```

## rmsea.ci.lower rmsea.ci.upper
##          0.038          0.054

semTools:::compRelSEM(fit_bifactor)

## G_Factor      AP      AN
## 0.246     0.091    0.899

# Verificar o resultado
summary(fit_bifactor, standardized = TRUE)

## lavaan 0.6-20 ended normally after 63 iterations
##
##   Estimator           DWLS
## Optimization method        NLMINB
## Number of model parameters       121
## 
## Number of observations        457
## 
## Model Test User Model:
##                               Standard      Scaled
## Test Statistic            294.436    479.716
## Degrees of freedom         149        149
## P-value (Chi-square)      0.000      0.000
## Scaling correction factor      0.701
## Shift parameter             59.820
##   simple second-order correction
## 
## Parameter Estimates:
##                               Delta
## Parameterization
## Standard errors           Robust.sem
## Information                Expected
## Information saturated (h1) model Unstructured
## 
## Latent Variables:
##             Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## G_Factor =~
##   PN1ativo      0.793  0.026  30.559  0.000    0.793   0.793
##   PN2envergo   -0.091  0.051 -1.803  0.071   -0.091  -0.091
##   PN3atento     0.832  0.047 17.688  0.000    0.832   0.832
##   PN4aflit     -0.172  0.049 -3.530  0.000   -0.172  -0.172
##   PN5determ     0.818  0.022 36.797  0.000    0.818   0.818
##   PN6culpado   -0.194  0.049 -3.970  0.000   -0.194  -0.194
##   PN7empol      0.729  0.081  9.044  0.000    0.729   0.729
##   PN8irrit      0.022  0.047  0.477  0.633    0.022   0.022
##   PN9interes    0.718  0.071 10.048  0.000    0.718   0.718
##   PN10medo    -0.127  0.048 -2.648  0.008   -0.127  -0.127
##   PN11orgul     0.707  0.067 10.531  0.000    0.707   0.707
##   PN12hostil   -0.048  0.051 -0.931  0.352   -0.048  -0.048
##   PN13alerta    0.443  0.040 10.993  0.000    0.443   0.443
##   PN14inquie    0.001  0.048  0.017  0.986    0.001   0.001
##   PN15entusia   0.710  0.097  7.361  0.000    0.710   0.710

```

```

##   PN16nervo    -0.010    0.049   -0.201    0.841   -0.010   -0.010
##   PN17forte     0.633    0.077    8.228    0.000    0.633    0.633
##   PN18apavo    -0.313    0.048   -6.475    0.000   -0.313   -0.313
##   PN19inspi     0.610    0.088    6.960    0.000    0.610    0.610
##   PN20chate    -0.175    0.046   -3.769    0.000   -0.175   -0.175
## AP =~
##   PN1ativo     -0.058    0.157   -0.371    0.711   -0.058   -0.058
##   PN3atento    -0.214    0.163   -1.309    0.191   -0.214   -0.214
##   PN5determ     0.033    0.159    0.209    0.834    0.033    0.033
##   PN7empol      0.411    0.137    2.995    0.003    0.411    0.411
##   PN9interes     0.344    0.139    2.474    0.013    0.344    0.344
##   PN11orgul     0.329    0.138    2.391    0.017    0.329    0.329
##   PN13alerta     0.042    0.110    0.383    0.701    0.042    0.042
##   PN15entusia    0.506    0.132    3.817    0.000    0.506    0.506
##   PN17forte     0.376    0.126    2.975    0.003    0.376    0.376
##   PN19inspi     0.439    0.119    3.698    0.000    0.439    0.439
## AN =~
##   PN2envergo     0.516    0.036   14.363    0.000    0.516    0.516
##   PN4aflit       0.769    0.022   34.769    0.000    0.769    0.769
##   PN6culpado     0.661    0.030   22.020    0.000    0.661    0.661
##   PN8irrit       0.738    0.023   32.379    0.000    0.738    0.738
##   PN10medo      0.770    0.022   34.819    0.000    0.770    0.770
##   PN12hostil     0.579    0.033   17.316    0.000    0.579    0.579
##   PN14inquiet     0.702    0.026   27.283    0.000    0.702    0.702
##   PN16nervo      0.863    0.016   54.512    0.000    0.863    0.863
##   PN18apavo      0.677    0.029   23.641    0.000    0.677    0.677
##   PN20chate      0.765    0.022   34.702    0.000    0.765    0.765
##   PN13alerta     0.359    0.037   9.756    0.000    0.359    0.359
##
## Covariances:
##   G_Factor ~~~
##   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   AP          0.000
##   AN          0.000
## AP ~~
##   AN          0.000
## 
## Thresholds:
##   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   PN1ativo|t1   -1.872   0.117  -16.056   0.000  -1.872  -1.872
##   PN1ativo|t2   -1.399   0.085  -16.425   0.000  -1.399  -1.399
##   PN1ativo|t3   -0.885   0.068  -13.038   0.000  -0.885  -0.885
##   PN1ativo|t4    0.488   0.061   7.955   0.000   0.488  0.488
##   PN2envergo|t1  -0.635   0.063  -10.053   0.000  -0.635  -0.635
##   PN2envergo|t2   0.074   0.059   1.262   0.207   0.074  0.074
##   PN2envergo|t3   0.814   0.066  12.269   0.000   0.814  0.814
##   PN2envergo|t4   2.016   0.131  15.377   0.000   2.016  2.016
##   PN3atento|t1   -1.904   0.120  -15.923   0.000  -1.904  -1.904
##   PN3atento|t2   -1.399   0.085  -16.425   0.000  -1.399  -1.399
##   PN3atento|t3   -0.829   0.067  -12.441   0.000  -0.829  -0.829
##   PN3atento|t4    0.732   0.065  11.305   0.000   0.732  0.732
##   PN4aflit|t1   -1.090   0.073  -14.870   0.000  -1.090  -1.090
##   PN4aflit|t2   -0.488   0.061  -7.955   0.000  -0.488  -0.488
##   PN4aflit|t3    0.030   0.059   0.514   0.607   0.030  0.030

```

##	PN4aflit t4	1.141	0.075	15.229	0.000	1.141	1.141
##	PN5determ t1	-1.939	0.123	-15.768	0.000	-1.939	-1.939
##	PN5determ t2	-1.303	0.081	-16.102	0.000	-1.303	-1.303
##	PN5determ t3	-0.622	0.063	-9.872	0.000	-0.622	-0.622
##	PN5determ t4	0.662	0.064	10.413	0.000	0.662	0.662
##	PN6culpado t1	-0.602	0.063	-9.600	0.000	-0.602	-0.602
##	PN6culpado t2	0.008	0.059	0.140	0.889	0.008	0.008
##	PN6culpado t3	0.690	0.064	10.772	0.000	0.690	0.690
##	PN6culpado t4	1.621	0.097	16.641	0.000	1.621	1.621
##	PN7empol t1	-1.939	0.123	-15.768	0.000	-1.939	-1.939
##	PN7empol t2	-1.266	0.079	-15.936	0.000	-1.266	-1.266
##	PN7empol t3	-0.350	0.060	-5.829	0.000	-0.350	-0.350
##	PN7empol t4	0.740	0.065	11.394	0.000	0.740	0.740
##	PN8irrit t1	-1.303	0.081	-16.102	0.000	-1.303	-1.303
##	PN8irrit t2	-0.451	0.061	-7.402	0.000	-0.451	-0.451
##	PN8irrit t3	0.163	0.059	2.756	0.006	0.163	0.163
##	PN8irrit t4	1.356	0.083	16.299	0.000	1.356	1.356
##	PN9interes t1	-1.872	0.117	-16.056	0.000	-1.872	-1.872
##	PN9interes t2	-1.444	0.087	-16.529	0.000	-1.444	-1.444
##	PN9interes t3	-0.732	0.065	-11.305	0.000	-0.732	-0.732
##	PN9interes t4	0.622	0.063	9.872	0.000	0.622	0.622
##	PN10medo t1	-0.877	0.068	-12.953	0.000	-0.877	-0.877
##	PN10medo t2	-0.163	0.059	-2.756	0.006	-0.163	-0.163
##	PN10medo t3	0.344	0.060	5.736	0.000	0.344	0.344
##	PN10medo t4	1.184	0.076	15.501	0.000	1.184	1.184
##	PN11orgul t1	-1.544	0.093	-16.650	0.000	-1.544	-1.544
##	PN11orgul t2	-0.885	0.068	-13.038	0.000	-0.885	-0.885
##	PN11orgul t3	-0.140	0.059	-2.382	0.017	-0.140	-0.140
##	PN11orgul t4	0.869	0.067	12.869	0.000	0.869	0.869
##	PN12hostil t1	-0.367	0.060	-6.107	0.000	-0.367	-0.367
##	PN12hostil t2	0.275	0.060	4.620	0.000	0.275	0.275
##	PN12hostil t3	1.090	0.073	14.870	0.000	1.090	1.090
##	PN12hostil t4	1.976	0.127	15.588	0.000	1.976	1.976
##	PN13alerta t1	-1.399	0.085	-16.425	0.000	-1.399	-1.399
##	PN13alerta t2	-0.943	0.069	-13.619	0.000	-0.943	-0.943
##	PN13alerta t3	-0.235	0.059	-3.968	0.000	-0.235	-0.235
##	PN13alerta t4	0.987	0.070	14.024	0.000	0.987	0.987
##	PN14inquiet t1	-1.329	0.082	-16.204	0.000	-1.329	-1.329
##	PN14inquiet t2	-0.635	0.063	-10.053	0.000	-0.635	-0.635
##	PN14inquiet t3	-0.052	0.059	-0.888	0.375	-0.052	-0.052
##	PN14inquiet t4	0.960	0.070	13.782	0.000	0.960	0.960
##	PN15entusia t1	-1.733	0.105	-16.481	0.000	-1.733	-1.733
##	PN15entusia t2	-1.061	0.072	-14.646	0.000	-1.061	-1.061
##	PN15entusia t3	-0.286	0.060	-4.806	0.000	-0.286	-0.286
##	PN15entusia t4	0.837	0.067	12.527	0.000	0.837	0.837
##	PN16nervo t1	-1.042	0.072	-14.494	0.000	-1.042	-1.042
##	PN16nervo t2	-0.367	0.060	-6.107	0.000	-0.367	-0.367
##	PN16nervo t3	0.213	0.059	3.595	0.000	0.213	0.213
##	PN16nervo t4	1.195	0.077	15.567	0.000	1.195	1.195
##	PN17forte t1	-1.509	0.091	-16.625	0.000	-1.509	-1.509
##	PN17forte t2	-0.951	0.069	-13.701	0.000	-0.951	-0.951
##	PN17forte t3	-0.151	0.059	-2.569	0.010	-0.151	-0.151
##	PN17forte t4	0.918	0.069	13.372	0.000	0.918	0.918
##	PN18apavo t1	-0.207	0.059	-3.502	0.000	-0.207	-0.207

```

##   PN18apavo|t2    0.421    0.061    6.940    0.000    0.421    0.421
##   PN18apavo|t3    1.120    0.074   15.088    0.000    1.120    1.120
##   PN18apavo|t4    1.758    0.107   16.422    0.000    1.758    1.758
##   PN19inspi|t1   -1.758    0.107  -16.422    0.000   -1.758   -1.758
##   PN19inspi|t2   -1.061    0.072  -14.646    0.000   -1.061   -1.061
##   PN19inspi|t3   -0.190    0.059  -3.222    0.001   -0.190   -0.190
##   PN19inspi|t4    0.747    0.065  11.482    0.000    0.747    0.747
##   PN20chate|t1   -0.877    0.068  -12.953    0.000   -0.877   -0.877
##   PN20chate|t2   -0.247    0.059  -4.155    0.000   -0.247   -0.247
##   PN20chate|t3    0.275    0.060   4.620    0.000    0.275    0.275
##   PN20chate|t4    1.241    0.078  15.818    0.000    1.241    1.241
##
## Variances:
##                               Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## .PN1ativo            0.368      0.368
## .PN2envergo          0.726      0.726
## .PN3atento           0.262      0.262
## .PN4aflit            0.378      0.378
## .PN5determ           0.329      0.329
## .PN6culpado          0.526      0.526
## .PN7empol             0.299      0.299
## .PN8irrit             0.454      0.454
## .PN9interes           0.366      0.366
## .PN10medo            0.390      0.390
## .PN11orgul            0.392      0.392
## .PN12hostil           0.662      0.662
## .PN13alerta           0.673      0.673
## .PN14inquie           0.507      0.507
## .PN15entusia          0.239      0.239
## .PN16nervo             0.255      0.255
## .PN17forte            0.457      0.457
## .PN18apavo            0.444      0.444
## .PN19inspi            0.435      0.435
## .PN20chate            0.384      0.384
## G_Factor              1.000      1.000
## AP                    1.000      1.000
## AN                    1.000      1.000

```

### 3.3.6 Modelo bifatorial constrangido

Então fazemos algumas constrições para que ele seja mais parcimonioso

```

bifactor_model_syntax <- '
  G_Factor =~ PN1ativo + 0*PN2envergo + PN3atento + 0*PN4aflit + PN5determ +
  0*PN6culpado + PN7empol + 0*PN8irrit + PN9interes + 0*PN10medo +
  PN11orgul + 0*PN12hostil + PN13alerta + 0*PN14inquie +
  PN15entusia + 0*PN16nervo + PN17forte + PN18apavo +
  PN19inspi + 0*PN20chate

  AP =~ 0*PN1ativo + 0*PN3atento + 0*PN5determ + PN7empol + PN9interes + PN11orgul +
  0*PN13alerta + PN15entusia + PN17forte + PN19inspi

```

```

AN =~ PN2envergo + PN4aflit + PN6culpado + PN8irrit + PN10medo + PN12hostil + PN14inquie +
+ PN16nervo + PN18apavo + PN20chate + PN13alerta

'

# Tentar rodar este novo modelo
fit_bifactor <- cfa(bifactor_model_syntax, data = panas_data, ordered = T, orthogonal = T,
                      estimator = "WLSMV", std.lv=TRUE)
fitmeasures(fit_bifactor, c("chisq", "df", "pvalue", "cfi", "rmsea",
                            "rmsea.ci.lower","rmsea.ci.upper"))

##          chisq           df      pvalue       cfi      rmsea
##      591.050      162.000      0.000      0.984      0.076
##  rmsea.ci.lower rmsea.ci.upper
##      0.070      0.083

semTools::compRelSEM(fit_bifactor)

## G_Factor      AP      AN
##    0.736     0.276     0.909

summary(fit_bifactor, standardized = TRUE)

## lavaan 0.6-20 ended normally after 32 iterations
##
##   Estimator               DWLS
##   Optimization method      NLMINB
##   Number of model parameters      108
##
##   Number of observations      457
##
## Model Test User Model:
##                               Standard      Scaled
##   Test Statistic            591.050    357.748
##   Degrees of freedom         162        162
##   P-value (Chi-square)      0.000      0.000
##   Scaling correction factor      2.423
##   Shift parameter            113.830
##   simple second-order correction
##
## Parameter Estimates:
##
##   Parameterization             Delta
##   Standard errors              Robust.sem
##   Information                  Expected
##   Information saturated (h1) model Unstructured
##
## Latent Variables:
##                   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   G_Factor =~
##     PNiativo          0.795   0.024  33.147   0.000    0.795    0.795

```

```

##   PN2envergo      0.000      0.000      0.000      0.000
##   PN3atento       0.780     0.023    34.278      0.000      0.780      0.780
##   PN4aflit        0.000      0.000      0.000      0.000      0.000      0.000
##   PN5determ       0.859     0.022    39.063      0.000      0.859      0.859
##   PN6culpado      0.000      0.000      0.000      0.000      0.000      0.000
##   PN7empol         0.687     0.027    25.396      0.000      0.687      0.687
##   PN8irrit         0.000      0.000      0.000      0.000      0.000      0.000
##   PN9interes       0.694     0.028    24.351      0.000      0.694      0.694
##   PN10medo         0.000      0.000      0.000      0.000      0.000      0.000
##   PN11orgul        0.687     0.026    26.075      0.000      0.687      0.687
##   PN12hostil       0.000      0.000      0.000      0.000      0.000      0.000
##   PN13alerta        0.464     0.038    12.084      0.000      0.464      0.464
##   PN14inquie        0.000      0.000      0.000      0.000      0.000      0.000
##   PN15entusia       0.668     0.027    24.995      0.000      0.668      0.668
##   PN16nervo         0.000      0.000      0.000      0.000      0.000      0.000
##   PN17forte         0.620     0.030    20.469      0.000      0.620      0.620
##   PN18apavo         -0.315    0.049    -6.467      0.000     -0.315     -0.315
##   PN19inspi         0.579     0.032    18.274      0.000      0.579      0.579
##   PN20chate         0.000      0.000      0.000      0.000      0.000      0.000
## AP =~
##   PN1ativo         0.000      0.000      0.000      0.000      0.000      0.000
##   PN3atento        0.000      0.000      0.000      0.000      0.000      0.000
##   PN5determ        0.000      0.000      0.000      0.000      0.000      0.000
##   PN7empol          0.483     0.034    14.324      0.000      0.483      0.483
##   PN9interes        0.404     0.037    11.064      0.000      0.404      0.404
##   PN11orgul         0.358     0.041     8.668      0.000      0.358      0.358
##   PN13alerta        0.000      0.000      0.000      0.000      0.000      0.000
##   PN15entusia       0.564     0.031    18.304      0.000      0.564      0.564
##   PN17forte         0.387     0.040     9.743      0.000      0.387      0.387
##   PN19inspi         0.477     0.037    13.045      0.000      0.477      0.477
## AN =~
##   PN2envergo        0.524     0.036    14.699      0.000      0.524      0.524
##   PN4aflit          0.784     0.020    39.943      0.000      0.784      0.784
##   PN6culpado        0.681     0.028    23.932      0.000      0.681      0.681
##   PN8irrit          0.730     0.024    31.024      0.000      0.730      0.730
##   PN10medo          0.782     0.021    36.825      0.000      0.782      0.782
##   PN12hostil         0.581     0.034    17.314      0.000      0.581      0.581
##   PN14inquie        0.695     0.026    26.494      0.000      0.695      0.695
##   PN16nervo          0.855     0.016    52.130      0.000      0.855      0.855
##   PN18apavo          0.714     0.027    26.328      0.000      0.714      0.714
##   PN20chate          0.779     0.021    37.085      0.000      0.779      0.779
##   PN13alerta         0.310     0.040     7.665      0.000      0.310      0.310
##
## Covariances:
##   G_Factor ~~
##   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   AP           0.000      0.000      0.000      0.000      0.000
##   AN           0.000      0.000      0.000      0.000      0.000
##   AP ~~
##   AN           0.000      0.000      0.000      0.000      0.000
##   ##
## Thresholds:
##   PN1ativo|t1   Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   PN1ativo|t1   -1.872    0.117   -16.056    0.000   -1.872   -1.872

```

##	PN1ativo t2	-1.399	0.085	-16.425	0.000	-1.399	-1.399
##	PN1ativo t3	-0.885	0.068	-13.038	0.000	-0.885	-0.885
##	PN1ativo t4	0.488	0.061	7.955	0.000	0.488	0.488
##	PN2envergo t1	-0.635	0.063	-10.053	0.000	-0.635	-0.635
##	PN2envergo t2	0.074	0.059	1.262	0.207	0.074	0.074
##	PN2envergo t3	0.814	0.066	12.269	0.000	0.814	0.814
##	PN2envergo t4	2.016	0.131	15.377	0.000	2.016	2.016
##	PN3atento t1	-1.904	0.120	-15.923	0.000	-1.904	-1.904
##	PN3atento t2	-1.399	0.085	-16.425	0.000	-1.399	-1.399
##	PN3atento t3	-0.829	0.067	-12.441	0.000	-0.829	-0.829
##	PN3atento t4	0.732	0.065	11.305	0.000	0.732	0.732
##	PN4aflit t1	-1.090	0.073	-14.870	0.000	-1.090	-1.090
##	PN4aflit t2	-0.488	0.061	-7.955	0.000	-0.488	-0.488
##	PN4aflit t3	0.030	0.059	0.514	0.607	0.030	0.030
##	PN4aflit t4	1.141	0.075	15.229	0.000	1.141	1.141
##	PN5determ t1	-1.939	0.123	-15.768	0.000	-1.939	-1.939
##	PN5determ t2	-1.303	0.081	-16.102	0.000	-1.303	-1.303
##	PN5determ t3	-0.622	0.063	-9.872	0.000	-0.622	-0.622
##	PN5determ t4	0.662	0.064	10.413	0.000	0.662	0.662
##	PN6culpado t1	-0.602	0.063	-9.600	0.000	-0.602	-0.602
##	PN6culpado t2	0.008	0.059	0.140	0.889	0.008	0.008
##	PN6culpado t3	0.690	0.064	10.772	0.000	0.690	0.690
##	PN6culpado t4	1.621	0.097	16.641	0.000	1.621	1.621
##	PN7empol t1	-1.939	0.123	-15.768	0.000	-1.939	-1.939
##	PN7empol t2	-1.266	0.079	-15.936	0.000	-1.266	-1.266
##	PN7empol t3	-0.350	0.060	-5.829	0.000	-0.350	-0.350
##	PN7empol t4	0.740	0.065	11.394	0.000	0.740	0.740
##	PN8irrit t1	-1.303	0.081	-16.102	0.000	-1.303	-1.303
##	PN8irrit t2	-0.451	0.061	-7.402	0.000	-0.451	-0.451
##	PN8irrit t3	0.163	0.059	2.756	0.006	0.163	0.163
##	PN8irrit t4	1.356	0.083	16.299	0.000	1.356	1.356
##	PN9interes t1	-1.872	0.117	-16.056	0.000	-1.872	-1.872
##	PN9interes t2	-1.444	0.087	-16.529	0.000	-1.444	-1.444
##	PN9interes t3	-0.732	0.065	-11.305	0.000	-0.732	-0.732
##	PN9interes t4	0.622	0.063	9.872	0.000	0.622	0.622
##	PN10medo t1	-0.877	0.068	-12.953	0.000	-0.877	-0.877
##	PN10medo t2	-0.163	0.059	-2.756	0.006	-0.163	-0.163
##	PN10medo t3	0.344	0.060	5.736	0.000	0.344	0.344
##	PN10medo t4	1.184	0.076	15.501	0.000	1.184	1.184
##	PN11orgul t1	-1.544	0.093	-16.650	0.000	-1.544	-1.544
##	PN11orgul t2	-0.885	0.068	-13.038	0.000	-0.885	-0.885
##	PN11orgul t3	-0.140	0.059	-2.382	0.017	-0.140	-0.140
##	PN11orgul t4	0.869	0.067	12.869	0.000	0.869	0.869
##	PN12hostil t1	-0.367	0.060	-6.107	0.000	-0.367	-0.367
##	PN12hostil t2	0.275	0.060	4.620	0.000	0.275	0.275
##	PN12hostil t3	1.090	0.073	14.870	0.000	1.090	1.090
##	PN12hostil t4	1.976	0.127	15.588	0.000	1.976	1.976
##	PN13alerta t1	-1.399	0.085	-16.425	0.000	-1.399	-1.399
##	PN13alerta t2	-0.943	0.069	-13.619	0.000	-0.943	-0.943
##	PN13alerta t3	-0.235	0.059	-3.968	0.000	-0.235	-0.235
##	PN13alerta t4	0.987	0.070	14.024	0.000	0.987	0.987
##	PN14inquiet t1	-1.329	0.082	-16.204	0.000	-1.329	-1.329
##	PN14inquiet t2	-0.635	0.063	-10.053	0.000	-0.635	-0.635
##	PN14inquiet t3	-0.052	0.059	-0.888	0.375	-0.052	-0.052

```

##   PN14inquieto|t4    0.960    0.070   13.782   0.000    0.960    0.960
##   PN15entusia|t1   -1.733    0.105  -16.481   0.000   -1.733   -1.733
##   PN15entusia|t2   -1.061    0.072  -14.646   0.000   -1.061   -1.061
##   PN15entusia|t3   -0.286    0.060   -4.806   0.000   -0.286   -0.286
##   PN15entusia|t4    0.837    0.067   12.527   0.000    0.837    0.837
##   PN16nervoso|t1   -1.042    0.072  -14.494   0.000   -1.042   -1.042
##   PN16nervoso|t2   -0.367    0.060   -6.107   0.000   -0.367   -0.367
##   PN16nervoso|t3    0.213    0.059   3.595   0.000    0.213    0.213
##   PN16nervoso|t4    1.195    0.077   15.567   0.000    1.195    1.195
##   PN17forte|t1    -1.509    0.091  -16.625   0.000   -1.509   -1.509
##   PN17forte|t2    -0.951    0.069  -13.701   0.000   -0.951   -0.951
##   PN17forte|t3    -0.151    0.059   -2.569   0.010   -0.151   -0.151
##   PN17forte|t4    0.918    0.069   13.372   0.000    0.918    0.918
##   PN18apavoso|t1   -0.207    0.059   -3.502   0.000   -0.207   -0.207
##   PN18apavoso|t2    0.421    0.061   6.940   0.000    0.421    0.421
##   PN18apavoso|t3    1.120    0.074   15.088   0.000    1.120    1.120
##   PN18apavoso|t4    1.758    0.107   16.422   0.000    1.758    1.758
##   PN19inspirado|t1  -1.758    0.107  -16.422   0.000   -1.758   -1.758
##   PN19inspirado|t2  -1.061    0.072  -14.646   0.000   -1.061   -1.061
##   PN19inspirado|t3  -0.190    0.059   -3.222   0.001   -0.190   -0.190
##   PN19inspirado|t4    0.747    0.065   11.482   0.000    0.747    0.747
##   PN20chateado|t1   -0.877    0.068  -12.953   0.000   -0.877   -0.877
##   PN20chateado|t2   -0.247    0.059   -4.155   0.000   -0.247   -0.247
##   PN20chateado|t3    0.275    0.060   4.620   0.000    0.275    0.275
##   PN20chateado|t4    1.241    0.078   15.818   0.000    1.241    1.241
##
## Variances:
##             Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   .PN1ativo      0.367
##   .PN2envergo     0.725
##   .PN3atento      0.392
##   .PN4aflit       0.385
##   .PN5determ      0.262
##   .PN6culpado     0.537
##   .PN7empol        0.295
##   .PN8irrit        0.467
##   .PN9interes      0.355
##   .PN10medo       0.389
##   .PN11orgul       0.400
##   .PN12hostil      0.663
##   .PN13alerta      0.688
##   .PN14inquieto    0.516
##   .PN15entusia     0.235
##   .PN16nervoso     0.268
##   .PN17forte        0.465
##   .PN18apavoso     0.392
##   .PN19inspirado    0.437
##   .PN20chateado     0.394
##   G_Factor         1.000
##   AP                1.000
##   AN                1.000

```

### 3.3.7 Teste de modelo com segunda ordem

```

S0_mode <- '

G_Factor =~ PN1ativo + PN2envergo + PN3atento + PN4aflit + PN5determ +
PN6culpado + PN7empol + PN8irrit + PN9interes + PN10medo +
PN11orgul + PN12hostil + PN13alerta + PN14inquie +
PN15entusia + PN16nervo + PN17forte + PN18apavo +
PN19inspi + PN20chate

AP =~ PN1ativo + PN3atento + PN5determ + PN7empol + PN9interes + PN11orgul +
PN13alerta + PN15entusia + PN17forte + PN19inspi

AN =~ PN2envergo + PN4aflit + PN6culpado + PN8irrit + PN10medo + PN12hostil + PN14inquie +
+ PN16nervo + PN18apavo + PN20chate + PN13alerta
'

# Tentar rodar este novo modelo
fit_S0 <- cfa(S0_mode, data = panas_data, ordered = T, orthogonal = F,
estimator = "WLSMV", std.lv=TRUE)

## Warning: lavaan->lav_model_vcov():
##   Could not compute standard errors! The information matrix could not be
##   inverted. This may be a symptom that the model is not identified.

## Warning: lavaan->lav_test_satorra_bentler():
##   could not invert information matrix needed for robust test statistic

fitmeasures(fit_S0, c("chisq", "df", "pvalue", "cfi", "rmsea",
"rmsea.ci.lower","rmsea.ci.upper"))

##          chisq        df      pvalue      cfi      rmsea
##      291.546    146.000     0.000    0.994    0.047
##  rmsea.ci.lower rmsea.ci.upper
##      0.039      0.055

semTools:::compRelSEM(fit_S0)

## G_Factor      AP      AN
##  0.171    0.270    0.912

# Verificar o resultado
summary(fit_S0, fitmeasures = T, standardized = TRUE)

## Warning: lavaan->.local():
##   Unknown argument 'fitmeasures' for 'summary'

## lavaan 0.6-20 ended normally after 61 iterations
##
##   Estimator           DWLS

```

```

## Optimization method                               NLMINB
## Number of model parameters                      124
##
## Number of observations                          457
##
## Model Test User Model:
##                                         Standard   Scaled
## Test Statistic                           291.546    NA
## Degrees of freedom                       146        146
## P-value (Chi-square)                     0.000      NA
## Scaling correction factor                NA         NA
## Shift parameter                         NA         NA
##
## Parameter Estimates:
##                                         Delta
## Parameterization
## Standard errors                         Robust.sem
## Information                            Expected
## Information saturated (h1) model       Unstructured
##
## Latent Variables:
##             Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## G_Factor =~
##   PN1ativo      0.787    NA          0.787    0.787
##   PN2envergo   -0.132    NA         -0.132   -0.132
##   PN3atento     0.869    NA          0.869    0.869
##   PN4aflit     -0.223    NA         -0.223   -0.223
##   PN5determ     0.806    NA          0.806    0.806
##   PN6culpado   -0.247    NA         -0.247   -0.247
##   PN7empol      0.659    NA          0.659    0.659
##   PN8irrit     -0.005    NA         -0.005   -0.005
##   PN9interes    0.671    NA          0.671    0.671
##   PN10medo     -0.183    NA         -0.183   -0.183
##   PN11orgul     0.652    NA          0.652    0.652
##   PN12hostil    -0.083   NA         -0.083   -0.083
##   PN13alerta    0.412    NA          0.412    0.412
##   PN14inquie   -0.034    NA         -0.034   -0.034
##   PN15entusia   0.634    NA          0.634    0.634
##   PN16nervo     -0.047   NA         -0.047   -0.047
##   PN17forte     0.583    NA          0.583    0.583
##   PN18apavo     -0.380   NA         -0.380   -0.380
##   PN19inspi     0.539    NA          0.539    0.539
##   PN20chate    -0.226    NA         -0.226   -0.226
## AP =~
##   PN1ativo      0.157    NA          0.157    0.157
##   PN3atento    -0.016    NA         -0.016   -0.016
##   PN5determ     0.241    NA          0.241    0.241
##   PN7empol      0.593    NA          0.593    0.593
##   PN9interes    0.502    NA          0.502    0.502
##   PN11orgul     0.501    NA          0.501    0.501
##   PN13alerta    0.163    NA          0.163    0.163
##   PN15entusia   0.666    NA          0.666    0.666
##   PN17forte     0.510    NA          0.510    0.510

```

```

##      PN19inspi      0.587      NA                  0.587      0.587
## AN =~
##      PN2envergo     0.518      NA                  0.518      0.518
##      PN4aflit       0.777      NA                  0.777      0.777
##      PN6culpado     0.667      NA                  0.667      0.667
##      PN8irrit        0.742      NA                  0.742      0.742
##      PN10medo       0.775      NA                  0.775      0.775
##      PN12hostil      0.582      NA                  0.582      0.582
##      PN14inquie     0.705      NA                  0.705      0.705
##      PN16nervo       0.868      NA                  0.868      0.868
##      PN18apavo       0.686      NA                  0.686      0.686
##      PN20chate       0.773      NA                  0.773      0.773
##      PN13alerta      0.357      NA                  0.357      0.357
##
## Covariances:
##          Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## G_Factor ~~
##   AP        -0.104    NA                 -0.104    -0.104
##   AN         0.088    NA                 0.088     0.088
## AP ~~
##   AN        -0.077    NA                 -0.077    -0.077
##
## Thresholds:
##          Estimate Std.Err z-value P(>|z|) Std.lv Std.all
## PN1ativo|t1     -1.872    NA                 -1.872    -1.872
## PN1ativo|t2     -1.399    NA                 -1.399    -1.399
## PN1ativo|t3     -0.885    NA                 -0.885    -0.885
## PN1ativo|t4      0.488    NA                  0.488     0.488
## PN2envergo|t1    -0.635    NA                 -0.635    -0.635
## PN2envergo|t2     0.074    NA                  0.074     0.074
## PN2envergo|t3     0.814    NA                  0.814     0.814
## PN2envergo|t4     2.016    NA                  2.016     2.016
## PN3atento|t1    -1.904    NA                 -1.904    -1.904
## PN3atento|t2    -1.399    NA                 -1.399    -1.399
## PN3atento|t3    -0.829    NA                 -0.829    -0.829
## PN3atento|t4     0.732    NA                  0.732     0.732
## PN4aflit|t1     -1.090    NA                 -1.090    -1.090
## PN4aflit|t2     -0.488    NA                 -0.488    -0.488
## PN4aflit|t3      0.030    NA                  0.030     0.030
## PN4aflit|t4      1.141    NA                  1.141     1.141
## PN5determ|t1    -1.939    NA                 -1.939    -1.939
## PN5determ|t2    -1.303    NA                 -1.303    -1.303
## PN5determ|t3    -0.622    NA                 -0.622    -0.622
## PN5determ|t4      0.662    NA                  0.662     0.662
## PN6culpado|t1    -0.602    NA                 -0.602    -0.602
## PN6culpado|t2     0.008    NA                  0.008     0.008
## PN6culpado|t3     0.690    NA                  0.690     0.690
## PN6culpado|t4     1.621    NA                  1.621     1.621
## PN7empol|t1     -1.939    NA                 -1.939    -1.939
## PN7empol|t2     -1.266    NA                 -1.266    -1.266
## PN7empol|t3     -0.350    NA                 -0.350    -0.350
## PN7empol|t4      0.740    NA                  0.740     0.740
## PN8irrit|t1     -1.303    NA                 -1.303    -1.303
## PN8irrit|t2     -0.451    NA                 -0.451    -0.451

```

```

##   PN8irrit|t3      0.163     NA          0.163    0.163
##   PN8irrit|t4      1.356     NA          1.356    1.356
##   PN9interes|t1    -1.872     NA         -1.872   -1.872
##   PN9interes|t2    -1.444     NA         -1.444   -1.444
##   PN9interes|t3    -0.732     NA         -0.732   -0.732
##   PN9interes|t4     0.622     NA          0.622    0.622
##   PN10medo|t1     -0.877     NA         -0.877   -0.877
##   PN10medo|t2     -0.163     NA         -0.163   -0.163
##   PN10medo|t3      0.344     NA          0.344    0.344
##   PN10medo|t4      1.184     NA          1.184    1.184
##   PN11orgul|t1    -1.544     NA         -1.544   -1.544
##   PN11orgul|t2    -0.885     NA         -0.885   -0.885
##   PN11orgul|t3    -0.140     NA         -0.140   -0.140
##   PN11orgul|t4     0.869     NA          0.869    0.869
##   PN12hostil|t1   -0.367     NA         -0.367   -0.367
##   PN12hostil|t2     0.275     NA          0.275    0.275
##   PN12hostil|t3     1.090     NA          1.090    1.090
##   PN12hostil|t4     1.976     NA          1.976    1.976
##   PN13alerta|t1   -1.399     NA         -1.399   -1.399
##   PN13alerta|t2   -0.943     NA         -0.943   -0.943
##   PN13alerta|t3   -0.235     NA         -0.235   -0.235
##   PN13alerta|t4     0.987     NA          0.987    0.987
##   PN14inquiet|t1   -1.329     NA         -1.329   -1.329
##   PN14inquiet|t2   -0.635     NA         -0.635   -0.635
##   PN14inquiet|t3   -0.052     NA         -0.052   -0.052
##   PN14inquiet|t4     0.960     NA          0.960    0.960
##   PN15entusia|t1   -1.733     NA         -1.733   -1.733
##   PN15entusia|t2   -1.061     NA         -1.061   -1.061
##   PN15entusia|t3   -0.286     NA         -0.286   -0.286
##   PN15entusia|t4     0.837     NA          0.837    0.837
##   PN16nervo|t1    -1.042     NA         -1.042   -1.042
##   PN16nervo|t2    -0.367     NA         -0.367   -0.367
##   PN16nervo|t3     0.213     NA          0.213    0.213
##   PN16nervo|t4     1.195     NA          1.195    1.195
##   PN17forte|t1    -1.509     NA         -1.509   -1.509
##   PN17forte|t2    -0.951     NA         -0.951   -0.951
##   PN17forte|t3    -0.151     NA         -0.151   -0.151
##   PN17forte|t4     0.918     NA          0.918    0.918
##   PN18apavo|t1    -0.207     NA         -0.207   -0.207
##   PN18apavo|t2     0.421     NA          0.421    0.421
##   PN18apavo|t3     1.120     NA          1.120    1.120
##   PN18apavo|t4     1.758     NA          1.758    1.758
##   PN19inspi|t1    -1.758     NA         -1.758   -1.758
##   PN19inspi|t2    -1.061     NA         -1.061   -1.061
##   PN19inspi|t3    -0.190     NA         -0.190   -0.190
##   PN19inspi|t4     0.747     NA          0.747    0.747
##   PN20chate|t1    -0.877     NA         -0.877   -0.877
##   PN20chate|t2    -0.247     NA         -0.247   -0.247
##   PN20chate|t3     0.275     NA          0.275    0.275
##   PN20chate|t4     1.241     NA          1.241    1.241
##
##   ## Variances:
##   ##           Estimate Std.Err z-value P(>|z|) Std.lv Std.all
##   ## .PN1ativo       0.381

```

##	.PN2envergo	0.726	0.726	0.726
##	.PN3atento	0.242	0.242	0.242
##	.PN4aflit	0.377	0.377	0.377
##	.PN5determ	0.333	0.333	0.333
##	.PN6culpado	0.523	0.523	0.523
##	.PN7empol	0.295	0.295	0.295
##	.PN8irrit	0.449	0.449	0.449
##	.PN9interes	0.368	0.368	0.368
##	.PN10medo	0.391	0.391	0.391
##	.PN11orgul	0.391	0.391	0.391
##	.PN12hostil	0.662	0.662	0.662
##	.PN13alerta	0.674	0.674	0.674
##	.PN14inquie	0.506	0.506	0.506
##	.PN15entusia	0.242	0.242	0.242
##	.PN16nervo	0.251	0.251	0.251
##	.PN17forte	0.462	0.462	0.462
##	.PN18apavo	0.431	0.431	0.431
##	.PN19inspi	0.431	0.431	0.431
##	.PN20chate	0.382	0.382	0.382
##	G_Factor	1.000	1.000	1.000
##	AP	1.000	1.000	1.000
##	AN	1.000	1.000	1.000

### 3.3.8 Com certeza as labels estão erradas mas depois vemos isso

```

# --- PASSO 1: Preparar os rótulos para o gráfico ---
# A ordem é crucial: 20 itens, depois os 3 fatores (G_Factor, AP, AN)
rotulos_panas_bifactor <- c(
  # 20 itens: PN1 a PN20, representados por números
  "1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
  "11", "12", "13", "14", "15", "16", "17", "18", "19", "20",
  # 3 fatores latentes, com nomes mais descriptivos
  "Geral", "AP", "AN"
)

# --- PASSO 2: Plotar o modelo, salvando em um arquivo PNG largo ---
# O arquivo será salvo no seu diretório de trabalho
png("plot_panas_bifatorial_correto.png", height = 8, width = 12, units = 'in', res = 300)

semPaths(
  fit_bifactor,          # O objeto do seu modelo PANAS
  what = "std",
  whatLabels = "est",
  edge.color="black",

  # --- O argumento chave que define o fator geral do SEU modelo ---
  bifactor = "G_Factor",   # Informa ao semPlot qual é o fator geral

  # --- Layout e Estética ---
  layout = "tree2",        # Mantém o layout hierárquico
  residuals = FALSE,
  intercepts = FALSE,
)

```

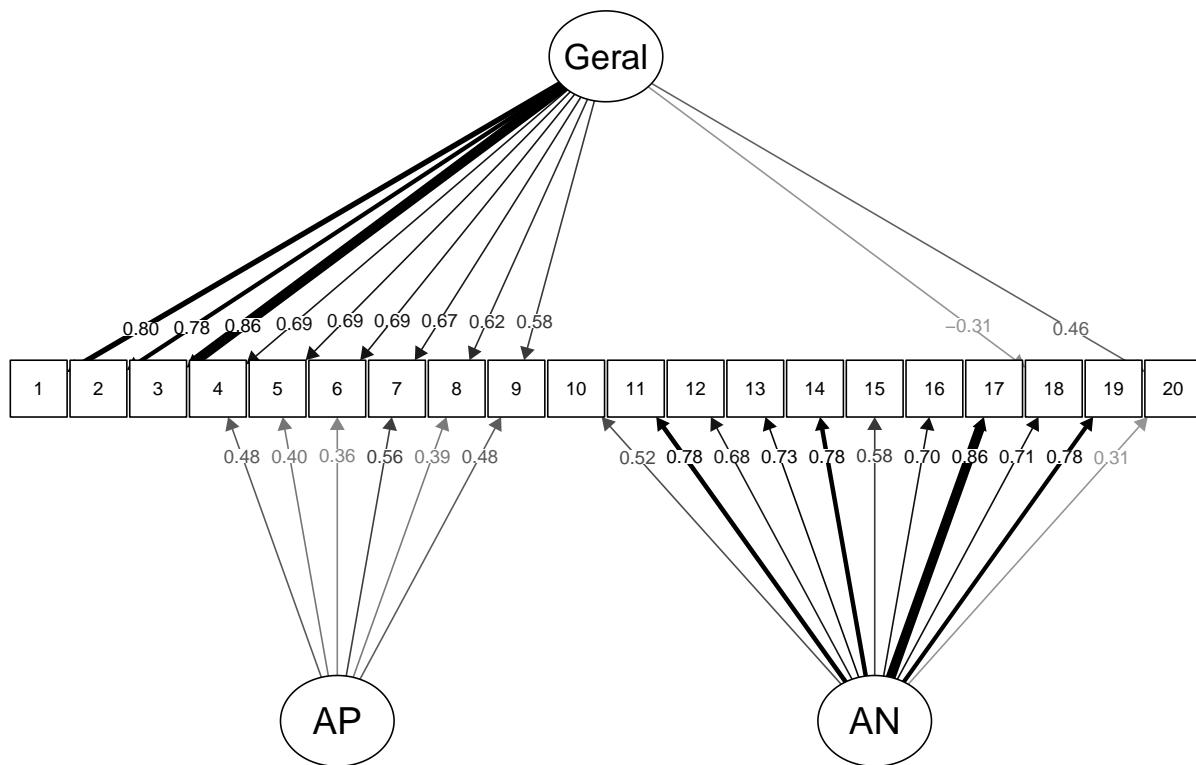
```

thresholds = FALSE,

# --- Rótulos e Tamanhos ---
nodeLabels = rotulos_panas_bifactor, # Usa os rótulos que criamos
edge.label.cex = 0.7,
sizeMan = 5,
sizeLat = 10,
sizeLat2=8,
edge.label.position=0.85,

# --- Outros ajustes visuais ---
style = "lisrel",
nCharNodes = 0,
mar = c(2, 1, 4, 1)      # Margens (bottom, left, top, right)
)

```



```

# Fecha o dispositivo gráfico e salva o arquivo
#dev.off()

```

```

sessionInfo()

```

```

## R version 4.5.0 (2025-04-11 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)

```

```

## 
## Matrix products: default
##   LAPACK version 3.12.1
##
## locale:
## [1] LC_COLLATE=Portuguese_Brazil.utf8  LC_CTYPE=Portuguese_Brazil.utf8
## [3] LC_MONETARY=Portuguese_Brazil.utf8 LC_NUMERIC=C
## [5] LC_TIME=Portuguese_Brazil.utf8
##
## time zone: America/Sao_Paulo
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] plotly_4.10.4          semPlot_1.1.6        GGally_2.2.1
## [4] ggrepel_0.9.6          patchwork_1.3.0      ggplot2_3.5.2
## [7] EGAnet_2.3.0           cSEM_0.6.1         seminr_2.3.7
## [10] semTools_0.5-7        lavaan_0.6-20       GPArotation_2025.3-1
## [13] EFA.dimensions_0.1.8.4 psych_2.5.3       stringr_1.5.1
## [16] stringi_1.8.7         janitor_2.2.1      tidyverse_1.3.1
## [19] dplyr_1.1.4           readxl_1.4.5       readr_2.1.5
## [22] pacman_0.5.1
##
## loaded via a namespace (and not attached):
## [1] splines_4.5.0          tibble_3.2.1        R.oo_1.27.1
## [4] cellranger_1.1.0       XML_3.99-0.18      rpart_4.1.24
## [7] lifecycle_1.0.4         Rdpack_2.6.4       vroom_1.6.5
## [10] mirt_1.44.0            globals_0.18.0      lattice_0.22-7
## [13] MASS_7.3-65             rockchalk_1.8.157 backports_1.5.0
## [16] magrittr_2.0.3          openxlsx_4.2.8      Hmisc_5.2-3
## [19] rmarkdown_2.29           yaml_2.3.10        qgraph_1.9.8
## [22] zip_2.3.2              sessioninfo_1.2.3  pbapply_1.7-2
## [25] minqa_1.2.8            RColorBrewer_1.1-3 lubridate_1.9.4
## [28] multcomp_1.4-28         abind_1.4-8        audio_0.1-11
## [31] quadprog_1.5-8          purrrr_1.0.4       R.utils_2.13.0
## [34] nnet_7.3-20              TH.data_1.1-3      sandwich_3.1-1
## [37] listenv_0.9.1           testthat_3.2.3      vegan_2.6-10
## [40] arm_1.14-4              parallelly_1.44.0  permute_0.9-7
## [43] codetools_0.2-20         tidyselect_1.2.1    farver_2.1.2
## [46] lme4_1.1-37              matrixStats_1.5.0   stats4_4.5.0
## [49] base64enc_0.1-3          jsonlite_2.0.0      polycor_0.8-1
## [52] progressr_0.15.1         Formula_1.2-5      survival_3.8-3
## [55] emmeans_1.11.1           tools_4.5.0        Rcpp_1.0.14
## [58] glue_1.8.0                mnormt_2.1.1       gridExtra_2.3
## [61] xfun_0.52                 mgcv_1.9-3        admisc_0.38
## [64] withr_3.0.2               beepr_2.0         fastmap_1.2.0
## [67] boot_1.3-31              digest_0.6.37      mi_1.1
## [70] timechange_0.3.0          R6_2.6.1          estimability_1.5.1
## [73] colorspace_2.1-1          gtools_3.9.5       jpeg_0.1-11
## [76] R.methodsS3_1.8.2          utf8_1.2.5        generics_0.1.3
## [79] data.table_1.17.0          corpcor_1.6.10    SimDesign_2.19.2
## [82] httr_1.4.7                htmlwidgets_1.6.4  ggstats_0.9.0

```

```
## [85] pkgconfig_2.0.3      sem_3.1-16          gtable_0.3.6
## [88] brio_1.1.5           htmltools_0.5.8.1 carData_3.0-5
## [91] scales_1.4.0          png_0.1-8           reformulas_0.4.1
## [94] snakecase_0.11.1     knitr_1.50          rstudioapi_0.17.1
## [97] tzdb_0.5.0           reshape2_1.4.4     coda_0.19-4.1
## [100] checkmate_2.3.2    nlme_3.1-168        nloptr_2.2.1
## [103] zoo_1.8-14          parallel_4.5.0    foreign_0.8-90
## [106] pillar_1.10.2       grid_4.5.0          vctrs_0.6.5
## [109] OpenMx_2.21.13     xtable_1.8-4       Deriv_4.1.6
## [112] cluster_2.1.8.1    dcurver_0.9.2      htmlTable_2.4.3
## [115] evaluate_1.0.3      pbivnorm_0.6.0     mvtnorm_1.3-3
## [118] cli_3.6.5           kutils_1.73         compiler_4.5.0
## [121] rlang_1.1.6          crayon_1.5.3        future.apply_1.11.3
## [124] labeling_0.4.3      fdrtool_1.2.18     plyr_1.8.9
## [127] viridisLite_0.4.2   lazyeval_0.2.2      lisrelToR_0.3
## [130] Matrix_1.7-3         hms_1.1.3           glasso_1.11
## [133] bit64_4.6.0-1       future_1.40.0       rbibutils_2.3
## [136] igraph_2.1.4         RcppParallel_5.1.10 bit_4.6.0
```