

SUPPLEMENTARY MATERIAL

Complete lavaan syntax for all models

(lavaan 0.6-21; semTools 0.5-8; R 4.5.2; N = 506)

Manuscript: Multivariate Modeling of Organizational Dynamics and
Faculty Well-Being in Higher Education

=====

S1. MEASUREMENT MODEL (CFA)

=====

```
modelo_medida <- '  
  LT =~ lt_1 + lt_2 + lt_3 + lt_4  
  LS =~ ls_1 + ls_2 + ls_3 + ls_4 + ls_5 + ls_6 + ls_7 +  
    ls_8 + ls_9 + ls_10 + ls_11 + ls_12 + ls_13 + ls_14  
  AE =~ ae_1 + ae_2 + ae_3 + ae_4 + ae_5 + ae_6 +  
    ae_7 + ae_8 + ae_9 + ae_10  
  AC =~ ac_1 + ac_2 + ac_3 + ac_4 + ac_5 + ac_6 + ac_7 +  
    ac_8 + ac_9 + ac_10 + ac_11 + ac_12 + ac_13 + ac_14 +  
    ac_15 + ac_16 + ac_17 + ac_18 + ac_19 + ac_20  
  SV =~ sv_1 + sv_2 + sv_3 + sv_4 + sv_5  
,  
  
fit_cfa <- cfa(modelo_medida, data = datos,  
  estimator = "WLSMV", std.lv = TRUE, ordered = item_vars)
```

=====

S2. STRUCTURAL MODEL (single group)

Direct residual paths LT->SV and LS->SV are estimated, so the
type of mediation (partial vs indirect-only) is tested rather
than imposed by specification (Zhao, Lynch & Chen, 2010).

```

=====

modelo_sem <- '
[measurement block identical to S1]

LT ~~ LS
AC ~ a1*LT + a2*LS
AE ~ b1*LT + b2*LS + b3*AC
SV ~ c1*AE + c2*LT + c3*LS      # direct residual paths

# Simple indirect effects
ind_LT_AE_SV := b1 * c1
ind_LS_AE_SV := b2 * c1
# Serial indirect effects (via harassment)
ind_LT_AC_AE_SV := a1 * b3 * c1
ind_LS_AC_AE_SV := a2 * b3 * c1
# Direct residual effects
dir_LT_SV := c2
dir_LS_SV := c3
# Total effects = direct + all indirect
total_LT_SV := c2 + b1*c1 + a1*b3*c1
total_LS_SV := c3 + b2*c1 + a2*b3*c1
'

fit_sem <- sem(modelo_sem, data = datos,
               estimator = "WLSMV", std.lv = TRUE, ordered = item_vars)

# Standardized coefficients with delta-method 95% CIs (reported in Table 4):
standardizedSolution(fit_sem, ci = TRUE)

=====

```

S3. BOOTSTRAP SENSITIVITY ANALYSIS (BCa, B = 5000)

=====

```
fit_boot <- sem(modelo_sem, data = datos,  
               estimator = "ML", std.lv = TRUE,  
               se = "bootstrap", bootstrap = 5000)
```

```
parameterEstimates(fit_boot, boot.ci.type = "bca.simple", ci = TRUE)
```

=====

S4. COVARIATE-ADJUSTED SENSITIVITY MODEL

(age standardised; university type: private = 1;
region dummies: highlands, jungle; reference = coast)

=====

```
modelo_sem_ctrl <- paste0(modelo_sem, '  
AC ~ z_edad + univ_priv + reg_sierra + reg_selva  
AE ~ z_edad + univ_priv + reg_sierra + reg_selva  
SV ~ z_edad + univ_priv + reg_sierra + reg_selva  
' )
```

```
fit_sem_ctrl <- sem(modelo_sem_ctrl, data = datos_ctrl,  
                   estimator = "WLSMV", std.lv = TRUE, ordered = item_vars)
```

=====

S5. MEASUREMENT INVARIANCE ACROSS GENDER (ordinal indicators)

Because the indicators are ordered-categorical, the scalar
step constrains item THRESHOLDS (Wu & Estabrook, 2016);
intercepts are fixed at zero under this parameterisation.
Sparse response categories were collapsed across groups
prior to fitting to prevent empty bivariate cells.

```
=====

# M0 - Configural
fit_conf <- cfa(modelo_medida, data = datos_mg, group = "sd_sexo",
               estimator = "WLSMV", std.lv = TRUE, ordered = item_vars)

# M1 - Metric (equal loadings)
fit_metric <- cfa(modelo_medida, data = datos_mg, group = "sd_sexo",
                 estimator = "WLSMV", std.lv = TRUE, ordered = item_vars,
                 group.equal = "loadings")

# M2 - Scalar (equal loadings + thresholds)
fit_scalar <- cfa(modelo_medida, data = datos_mg, group = "sd_sexo",
                 estimator = "WLSMV", std.lv = TRUE, ordered = item_vars,
                 group.equal = c("loadings", "thresholds"))

# Nested comparisons (Satorra-2000 scaled difference tests)
lavTestLRT(fit_conf, fit_metric) # Dchi2(48) = 239.05, p < .001; DCFI = .002
lavTestLRT(fit_metric, fit_scalar) # scaled Dchi2 < 0 (n.s.); DCFI = -.003

# Invariance decisions combine Dchi2 with approximate fit criteria
# (DCFI <= .010; Cheung & Rensvold, 2002; Chen, 2007). A negative
# scaled Dchi2 is admissible under the mean-and-variance correction
# (Satorra & Bentler, 2010) and indicates no deterioration in fit.
```

=====

S6. MULTIGROUP STRUCTURAL MODELS (gender)

IMPORTANT SPECIFICATION NOTE. In lavaan, a single parameter label (e.g., a1*LT) used in a multigroup model is applied to BOTH groups and silently imposes a cross-group equality constraint. The

structural syntax for the multigroup models therefore uses either NO labels (S6a) or group-specific label VECTORS c(labelM,labelW) (S6c). This corrects the specification error present in the previous submission, which produced identical coefficients across groups.

=====

```
# ---- S6a. Free structural model (M_free): loadings equal, all
#      eight regression paths freely estimated per group.
modelo_sem_mg <- paste0(modelo_medida, '
  LT ~~ LS
  AC ~ LT + LS
  AE ~ LT + LS + AC
  SV ~ AE + LT + LS
')

fit_mg_libre <- sem(modelo_sem_mg, data = datos_mg, group = "sd_sexo",
  estimator = "WLSMV", std.lv = TRUE, ordered = item_vars,
  group.equal = "loadings")

# ---- S6b. Constrained model (M_equal): loadings + all regressions
#      constrained to equality across groups.
fit_mg_const <- sem(modelo_sem_mg, data = datos_mg, group = "sd_sexo",
  estimator = "WLSMV", std.lv = TRUE, ordered = item_vars,
  group.equal = c("loadings", "regressions"))

# ---- Formal test of structural equivalence (reported in Section 4.10)
lavTestLRT(fit_mg_libre, fit_mg_const)

# Result: Dchi2(8) = 17.86, p = .022 -> full structural equivalence rejected.

# Group-specific estimates for Table 8 are taken from fit_mg_libre:
```

```
parameterEstimates(fit_mg_libre, standardized = TRUE, ci = TRUE)
```

```
# ---- S6c. Wald tests of individual path differences (Table 9)
```

```
#      Group-specific label vectors: first element = group 1 (Men),
```

```
#      second element = group 2 (Women).
```

```
modelo_sem_mg_lbl <- paste0(modelo_medida, '
```

```
  LT ~~ LS
```

```
  AC ~ c(a1m,a1w)*LT + c(a2m,a2w)*LS
```

```
  AE ~ c(b1m,b1w)*LT + c(b2m,b2w)*LS + c(b3m,b3w)*AC
```

```
  SV ~ c(c1m,c1w)*AE + c(c2m,c2w)*LT + c(c3m,c3w)*LS
```

```
# Men - Women differences (Wald tests, Table 9)
```

```
d_LT_AC := a1m - a1w
```

```
d_LS_AC := a2m - a2w
```

```
d_LT_AE := b1m - b1w
```

```
d_LS_AE := b2m - b2w
```

```
d_AC_AE := b3m - b3w
```

```
d_AE_SV := c1m - c1w
```

```
d_LT_SV := c2m - c2w
```

```
d_LS_SV := c3m - c3w
```

```
')
```

```
fit_mg_lbl <- sem(modelo_sem_mg_lbl, data = datos_mg, group = "sd_sexo",
```

```
  estimator = "WLSMV", std.lv = TRUE, ordered = item_vars,
```

```
  group.equal = "loadings")
```

```
parameterEstimates(fit_mg_lbl, ci = TRUE) # rows with op == "=="
```

```
# Significant differences: d_LT_AC = 0.210 (p < .05, CI [0.040, 0.379]);
```

```
#      d_LT_AE = 0.233 (p < .05, CI [0.009, 0.457]).
```

```
# All remaining six path differences are non-significant.
```

=====

REFERENCES FOR THE PROCEDURES ABOVE

=====

Chen, F. F. (2007). Structural Equation Modeling, 14(3), 464-504.

Cheung, G. W., & Rensvold, R. B. (2002). Structural Equation
Modeling, 9(2), 233-255.

Satorra, A., & Bentler, P. M. (2010). Psychometrika, 75(2), 243-248.

Wu, H., & Estabrook, R. (2016). Psychometrika, 81(4), 1014-1045.

Zhao, X., Lynch, J. G., & Chen, Q. (2010). Journal of Consumer
Research, 37(2), 197-206.