Preventing Rater Biases in 360-Degree Feedback by Forcing Choice

Supplement. Multiple-group CFA

The default assumption of the 360-degree assessment is the strict measurement invariance (i.e. invariance of thresholds, factor loadings and residual variances) across all rater perspectives. Only when the scale of every competency is equated across the rater perspectives, can we meaningfully compare assessments from all raters. On the other hand, distributions of the trait scores may vary across perspectives – thus, the traits may have different mean and covariance structures.

Method

To place all the competency scores on the same scale, we fitted the SS, SS-Method and FC models using a multiple-group confirmatory factor analysis. For each of the three measurement models, four groups (selves, bosses, peers and subordinates) were analyzed simultaneously, using the Unweighted Least Squares estimator with robust standard errors (denoted ULSMV in Mplus). To set the metric for every measured trait, we fixed its scale origin (the mean) to 0 and its scale unit (the standard deviation) to 1 in the referent group (arbitrarily, we chose "self" as the referent perspective). In the other groups, the means, the variances and the covariances of the traits were freely estimated. The item parameters were held equal across all groups. Specifically, the thresholds, the factor loadings on the 16 competency factors and the Method factor, and the unique variances were held equal (evidence for assuming the Method factor an equivalent construct across all rater perspectives was obtained in the single-group analyses).

Setting up multiple-group models in Mplus is easy; the program defaults assume strong measurement invariance (equivalence of factor loadings and thresholds) and free mean and covariance structures for the latent traits (Muthén & Muthén, 1998 –2015). To ensure strict measurement invariance (equivalence of unique variances) as we did in the present

study, one must explicitly constrain the item residual variance equal across all groups in the common part of the MODEL command.

Differential Test Functioning. Before using the multiple-group estimated scores in further analyses, we tested all scales for Differential Test Functioning (DTF). Because no modification indices could be obtained to identify violations to measurement invariance in the full models with 160 items (due to the very large sizes of these models), we used the scale scores estimated by the SS-Method model as proxies of the competency constructs in assessing DTF in all three measurement models (SS, SS-Method and FC). We then examined DTF for every pair of respondent perspectives (i.e. self-boss, self-peer, etc.) using the variance-based method of Penfield and Algina (2006). The method estimates Differential Item Functioning (DIF) for every polytomous item using the Lui-Agresti estimator of the cumulative common odds ratio, and then estimates the variance τ^2 of the DIF values across the items in each scale. The variance τ^2 is then compared to the variance expected by chance; thus it is the unsigned test of DTF.

Results

Goodness of Fit. The SS, SS-Method, and FC multiple-group models converged. The χ^2 statistics, Root Mean Square Error of Approximation (RMSEA) and Standardized Root Mean Square Residual (SRMR) are given in Table S1. The χ^2 statistics were significant – not surprising given the very large samples; however, the ratios of the χ^2 to the corresponding degrees of freedom were below 2 and smaller than in the single-group models (see Table 1 in the main article). The χ^2 contributions from each of the groups were broadly similar to the individual χ^2statistics, partly supporting the appropriateness of measurement invariance assumptions in each group. All the models had good approximate fits according to the RMSEA (below .05), and good exact fit according to SRMR (except FC models for selves

and bosses that were slightly above the threshold .08), also supporting the measurement invariance assumptions made in the measurement models.

Differential Test Functioning. The DTF for the vast majority of scales was small ($\tau^2 < .07$). Medium effects ($.07 \leq \tau^2 \leq .14$) were observed for Oral Communication between self and all other rater perspectives; however, when examined individually, DIF items cancelled out each other effects to yield only negligible signed effect at the scale level. The same was true for Persuasiveness, where a large unsigned DTF ($\tau^2 = .16$) was observed for the pair self-peers; however, only negligible signed effect resulted from positive DIF effects of 3 items, and negative effects of 2 items. The scale-level IMC scores were considered robust for group comparisons.

References

Muthén, L. K., & Muthén, B. O. (1998 –2015). *Mplus user's guide. Seventh Edition.* Los Angeles, CA: Muthén & Muthén.

Penfield, R. D., & Algina, J. (2006). A generalized DIF effect variance estimator for measuring unsigned differential test functioning in mixed format tests. *Journal of Educational Measurement, 43,* 295-312.

Table S1

Goodness of fit for alternative measurement models in multiple-group analyses assuming full measurement invariance

Model	SS	SS-Method	FC
Observed variables	80 (5 categories)	80 (5 categories)	120 (2 categories)
degrees of freedom	12,944	12,618	27,884**
χ^2 total	20,767	19,274	41,418
χ^2 contributions from each group			
Self	5,765	5,881	11,237
Boss	5,536	5,132	12,587
Peers	4,524	4,155	9,686
Subordinates	4,942	4,107	7,908
RMSEA (90% CI)	.023 (.022–.023)	.021 (.021–.022)	.020 (.020–.021)
SRMR for each group			
Self	.061	.052	.081
Boss	.063	.051	.087
Peers	.057	.046	.077
Subordinates	.054	.043	.073

Note. The models were limited to the first half of the questionnaire to enable calculation of χ^2 and RMSEA (see Endnote *v* in the main article); SRMR were calculated on the full questionnaire. 90% CI = 90 percent confidence interval. ** The degrees of freedom in the forced-choice models were adjusted for redundancies among the thresholds and tetrachoric correlations estimated from the binary outcome variables (Maydeu-Olivares, 1999). There are 4 redundancies per block of 4 items, thus the degrees of freedom printed by Mplus were reduced by 4*20 = 80 in each rater group.

Appendix S2. Mplus syntax for multiple-group analysis of a forced-choice questionnaire

Here we provide sample Mplus syntax for conducting multiple-group analysis of a hypothetical short forced-choice questionnaire with 3 blocks of size n = 4 measuring 4 traits. Binary outcomes of pairwise comparisons assumed derived from **full rankings**. If "most-least" format is used, estimation must proceed in 3 steps – first imputation, then estimating average parameters across multiple sets, and then finally applying averaged parameters to create syntax for estimating factior scores on the original data (with missing responses). Refer to Brown & Maydeu-Olivares (2012) for more detail on this process.

FC questionnaire data can be analysed using either a Thurstonian factor model as illustrated in Figure 2 of the main paper, or a Thurstonian IRT model, which is mathematically equivalent to the Thurstonian factor model (has the same parameters and fit, but reparameterized as a first-order factor model to enable estimation of factor scores). **Syntax 1** given below pertains to the Thurstonian factor model. It is very simple to read and interpret, and this is why it is recommended at the model testing stage. However, it cannot be used for scoring because the uniquenesses in this second-order factor model are parameterised to be 0 (see Brown & Maydeu-Olivares, 2012). **Syntax 2** is for Thurstonian IRT model, which can be used for both parameter estimation and person score estimation. Both models will have the same fit and the same parameter estimates.

Syntax 1. Thurstonian factor model (suitable for estimation only)

```
DATA:  FILE = "Feedback360_all_raters.dat";
VARIABLE:
        NAMES = AssessmentID FocusID RaterType
                !below are binary outcomes for 3 FC blocks
                i1i2 i1i3 i1i4 i2i3 i2i4 i3i4
                i5i6 i5i7 i5i8 i6i7 i6i8 i7i8
                i9i10 i9i11 i9i12 i10i11 i10i12 i11i12;
        USEVARIABLES = i1i2-i11i12;
        CATEGORICAL = ALL;
        GROUPING = RaterType (1=Self 2=Peer 3=Boss 4=Subordinate);
```

```
    CLUSTER = FocusID;

ANALYSIS:
    TYPE=COMPLEX;  !this analysis will account for the nested structure
    ESTIMATOR = ulsmv;
    PARAMETERIZATION=theta;

MODEL:   !this part is common to all groups
    !utilities are indicated by binary outcomes; they are first-order factors
    item1 BY i1i2@1 i1i3@1 i1i4@1;
    item2 BY i1i2@-1 i2i3@1 i2i4@1;
    item3 BY i1i3@-1 i2i3@-1 i3i4@1;
    item4 BY i1i4@-1 i2i4@-1 i3i4@-1;

    item5 BY i5i6@1 i5i7@1 i5i8@1;
    item6 BY i5i6@-1 i6i7@1 i6i8@1;
    item7 BY i5i7@-1 i6i7@-1 i7i8@1;
    item8 BY i5i8@-1 i6i8@-1 i7i8@-1;

    item9 BY i9i10@1 i9i11@1 i9i12@1;
    item10 BY i9i10@-1 i10i11@1 i10i12@1;
    item11 BY i9i11@-1 i10i11@-1 i11i12@1;
    item12 BY i9i12@-1 i10i12@-1 i11i12@-1;

    !binary outcomes are determined by the differences of utilities;
    !therefore, their residual variances are 0
    i1i2-i11i12@0;

    ! traits are indicated by utilities; they are second-order factors
    ! factor loadings are equal across groups
    Trait1  BY item1*1 item5 item9 (1-3);
    Trait2  BY item2*1 item6 item10 (4-6);
    Trait3  BY item3*1 item7 item11 (7-9);
    Trait4  BY item4*1 item8 item12 (10-12);

    !utility intercepts are equal across groups
    [item1-item12@0];

    ! fix one uniqueness per block for identification
    ! constrain other uniquenesses equal across groups
    item1@1; item2-item4 (13-15);
    item5@1; item6-item8 (16-18);
    item9@1; item10-item12 (19-21);

MODEL Self:   !this part is specific to the referent group (here, self)
    !metrics for traits are set in the referent group, and are free in other groups
    Trait1-Trait4@1;  [Trait1-Trait4@0];

    ! the errors command is repeated again,
    !otherwise Mplus sets them to 1 in the referent group by default
```

```
        i1i2-i11i12@0;
```

Syntax 2. Thurstonian IRT model (suitable for estimation and scoring)

```
DATA:  FILE = "Feedback360_all_raters.dat";
VARIABLE:
        NAMES = AssessmentID FocusID RaterType
                !below are binary outcomes for 3 FC blocks
                i1i2 i1i3 i1i4 i2i3 i2i4 i3i4
                i5i6 i5i7 i5i8 i6i7 i6i8 i7i8
                i9i10 i9i11 i9i12 i10i11 i10i12 i11i12;
        USEVARIABLES = i1i2-i11i12;
        CATEGORICAL = ALL;
        GROUPING = RaterType (1=Self 2=Peer 3=Boss 4=Subordinate);
        CLUSTER = FocusID;

ANALYSIS:
        TYPE=COMPLEX;  !this analysis will account for the nested structure
        ESTIMATOR = ulsmv;
        PARAMETERIZATION=theta;

MODEL:   !this part is common to all groups
        ! traits are indicated by the binary outcomes directly; they are first-order factors
        Trait1 BY
                i1i2*1 i1i3 i1i4 (L1)
                i5i6 i5i7 i5i8 (L5)
                i9i10 i9i11 i9i12 (L9);
        Trait2 BY
                i1i2*-1 (L2_n)
                i2i3 i2i4 (L2)
                i5i6 (L6_n)
                i6i7 i6i8 (L6)
                i9i10 (L10_n)
                i10i11 i10i12 (L10);
        Trait3 BY
                i1i3*-1 i2i3 (L3_n)
                i3i4 (L3)
                i5i7 i6i7 (L7_n)
                i7i8 (L7)
                i9i11 i10i11 (L11_n)
                i11i12 (L11);
        Trait4 BY
                i1i4*-1 i2i4 i3i4 (L4_n)
                i5i8 i6i8 i7i8 (L8_n)
                i9i12 i10i12 i11i12 (L12_n);

        ! declare uniquenesses and set their starting values
        i1i2*2 (e1e2);
        i1i3*2 (e1e3);
        i1i4*2 (e1e4);
```

```
i2i3*2 (e2e3);
i2i4*2 (e2e4);
i3i4*2 (e3e4);
i5i6*2 (e5e6);
i5i7*2 (e5e7);
i5i8*2 (e5e8);
i6i7*2 (e6e7);
i6i8*2 (e6e8);
i7i8*2 (e7e8);
i9i10*2 (e9e10);
i9i11*2 (e9e11);
i9i12*2 (e9e12);
i10i11*2 (e10e11);
i10i12*2 (e10e12);
i11i12*2 (e11e12);

! declare correlated uniqunesses and set their starting values
i1i2 WITH i1i3*1 i1i4*1 (e1);
i1i2 WITH i2i3*-1 i2i4*-1 (e2_n);
i1i3 WITH i1i4*1 (e1);
i1i3 WITH i2i3*1 (e3);
i1i3 WITH i3i4*-1 (e3_n);
i1i4 WITH i2i4*1 i3i4*1 (e4);
i2i3 WITH i2i4*1 (e2);
i2i3 WITH i3i4*-1 (e3_n);
i2i4 WITH i3i4*1 (e4);

i5i6 WITH i5i7*1 i5i8*1 (e5);
i5i6 WITH i6i7*-1 i6i8*-1 (e6_n);
i5i7 WITH i5i8*1 (e5);
i5i7 WITH i6i7*1 (e7);
i5i7 WITH i7i8*-1 (e7_n);
i5i8 WITH i6i8*1 i7i8*1 (e8);
i6i7 WITH i6i8*1 (e6);
i6i7 WITH i7i8*-1 (e7_n);
i6i8 WITH i7i8*1 (e8);

i9i10 WITH i9i11*1 i9i12*1 (e9);
i9i10 WITH i10i11*-1 i10i12*-1 (e10_n);
i9i11 WITH i9i12*1 (e9);
i9i11 WITH i10i11*1 (e11);
i9i11 WITH i11i12*-1 (e11_n);
i9i12 WITH i10i12*1 i11i12*1 (e12);
i10i11 WITH i10i12*1 (e10);
i10i11 WITH i11i12*-1 (e11_n);
i10i12 WITH i11i12*1 (e12);

MODEL CONSTRAINT:
! factor loadings relating to the same item are equal in absolute value
L2_n = -L2;
```

```
L3_n = -L3;
L6_n = -L6;
L7_n = -L7;
L10_n = -L10;
L11_n = -L11;

! uniquenesses relating to the same item are equal in absolute value
e2_n = -e2;
e3_n = -e3;
e6_n = -e6;
e7_n = -e7;
e10_n = -e10;
e11_n = -e11;

! pair's uniqueness is equal to sum of 2 utility uniqunesses
e1e2 = e1 + e2;       e1e3 = e1 + e3;       e1e4 = e1 + e4;
e2e3 = e2 + e3;       e2e4 = e2 + e4;       e3e4 = e3 + e4;
e5e6 = e5 + e6;       e5e7 = e5 + e7;       e5e8 = e5 + e8;
e6e7 = e6 + e7;       e6e8 = e6 + e8;       e7e8 = e7 + e8;
e9e10 = e9 + e10;     e9e11 = e9 + e11;     e9e12 = e9 + e12;
e10e11 = e10 + e11;   e10e12 = e10 + e12;   e11e12 = e11 + e12;

! fix one uniqueness per block for identification
e1 = 1;          e5 = 1;          e9 = 1;

MODEL Self:   !this part is specific to the referent group (here, self)
    !metrics for traits are set in the referent group, and are free in other groups
    Trait1-Trait4@1;   [Trait1-Trait4@0];

    ! the errors commands are repeated again,
    !otherwise Mplus sets them to 1 in the referent group by default
    i1i2*2 (e1e2);
    i1i3*2 (e1e3);
    i1i4*2 (e1e4);
    i2i3*2 (e2e3);
    i2i4*2 (e2e4);
    i3i4*2 (e3e4);
    i5i6*2 (e5e6);
    i5i7*2 (e5e7);
    i5i8*2 (e5e8);
    i6i7*2 (e6e7);
    i6i8*2 (e6e8);
    i7i8*2 (e7e8);
    i9i10*2 (e9e10);
    i9i11*2 (e9e11);
    i9i12*2 (e9e12);
    i10i11*2 (e10e11);
    i10i12*2 (e10e12);
    i11i12*2 (e11e12);
```

```
SAVE:  !estimate and save factor scores for each respondent
       FILE = Feedback360scores.dat;
       SAVE = FSCORES;
```