## 2.4 | Factor Communality and Variance

After an appropriate orthogonal rotation has been performed, the final step is to validate how successful the resulting factors are at capturing the variance and covariance structure contained within the original data matrix. Remember, any attempt to represent data using a smaller number of factors than original dimensions will inevitably result in some loss of data. Factor validation seeks to quantify this data loss and determine if the amount of information lost is acceptable given the added benefit in simplicity and lack of collinearity.

### 2.4.1 | Calculating the Communality of each Dimension

One of the simplest means of quantifying data loss following factor extraction is to measure it separately for each of the original dimensions.

Consider again the optimally orthogonally rotated two-factor representation and the corresponding loading matrix $\boldsymbol{A}$ of our example data matrix.

```
A <- pca(r = R, nfactors = 2, rotate = "varimax")$loadings[]
A
                          RC1         RC2
Hopeless          0.737684419  0.35721446
Overwhelmed       0.751913330 -0.10999159
Exhausted         0.767501553 -0.04935094
Lonely            0.762527268  0.25339680
Sad               0.798470525  0.27335188
Depressed         0.689253418  0.47745704
Anxious           0.742138215  0.23503718
SelfHarming       0.163183240  0.78456520
SuicidalThoughts  0.232827147  0.82265971
SuicidalAttempts  0.003767682  0.80222254
```

Recall that the rows of this and any other loading matrix represent each of the original dimensions, whereas the columns represent each of the rotated factors. The values in this matrix can be referenced using matrix notation as $a_{ij}$ or $a_{i,j}$, and represent the correlation or loading between dimension $i$ and retained factor $j$.

The sum of these squared loadings across each row or dimension is known as the *communality* $h^2$, which can be interpreted as the proportion of the variance in that dimension that is accounted for or retained by the set of extracted factors:

$$h_i^2 = \sum_{j=1} a_{ij}^2 \tag{2.27}$$

In R, we can use the `rowSums()` function to calculate these metrics or simply calculate them manually.

```
rowSums(A^2)
       Hopeless      Overwhelmed        Exhausted           Lonely
      0.6717805        0.5774718        0.5914941        0.6456578
            Sad        Depressed          Anxious      SelfHarming
      0.7122764        0.7030355        0.6060116        0.6421713
SuicidalThoughts SuicidalAttempts
      0.7309775        0.6435752

A[,1]^2 + A[,2]^2
       Hopeless      Overwhelmed        Exhausted           Lonely
      0.6717805        0.5774718        0.5914941        0.6456578
            Sad        Depressed          Anxious      SelfHarming
      0.7122764        0.7030355        0.6060116        0.6421713
SuicidalThoughts SuicidalAttempts
      0.7309775        0.6435752
```

Although there is no set limit for how much variance should be retained and, by extension, how much loss in variance is acceptable, a common threshold is to retain at least 70% of the variance in each dimension. This helps to avoid any data loss in excess of 30%.

For our example data, the extracted factors retained approximately 71% of the variance in feeling `Sad`, which is just within this limit. The amount of variance in feeling `Exhausted` that was retained by this two-factor solution, on the other hand, was only 58%, indicating a loss of just over 42%. Such a loss is much more concerning.

This pattern of factors retaining substantial amounts of the variance of certain dimensions while losing substantial amounts of other dimensions is fairly common in exploratory factor analysis and makes determining if a sufficient of amount of data has been retained rather challenging.

Thankfully, several alternative metrics to measure data retention and loss have been developed that are somewhat more succinct.

### 2.4.2 | Calculating the Variance of Each Factor

One alternative to calculating the proportion of variance retained from each dimension is to instead measure the variance in the entire set of original dimensions that was retained by each factor:

$$Var(f_j) = \sum_{i=1}^{d} \frac{{a_{ij}}^2}{d} \tag{2.28}$$

Dividing the sum of squared loadings for each column by the dimensionality of the original data in effect averages the proportion of variance accounted for in each dimension by a single factor, thereby provide a single metric for that factor.

In R, we can use the `colSums()` function to calculate these metrics or again calculate them manually.

```
Var <- colSums(A^2) / 10
Var
      RC1       RC2
0.4024304 0.2500147
```

For our example data, the first factor accounted for (on average) just over 40% of the variance in the original dimensions, whereas the second factor accounted for an additional 25% of the remaining variance.

Given that the amount of variance accounted for by different orthogonal factors is disjoint, we can use the *cumulative variance* to measure the proportion of variance that is jointly retained by each factor and those that preceded it:

$$Cummulative\ Var(f_j) = \sum_{i=1}^{j} Var(f_i) \tag{2.29}$$

We can quickly implement this equation for any factor in R using simple arithmetic.

```
CummulVar_f2 <- Var[1] + Var[2]
CummulVar_f2
      RC1
0.6524452
```

For our example data, the cumulative proportion of variance explained by both the first and second factor was just over 65%, indicating that our two-factor solution lost just over 35% of the variance in the original dimensions. This loss, unfortunately, is beyond the commonly used threshold of 30%, indicating that our two-factor solution was not sufficient to retain an acceptable amount of information.

Before moving one, it is important to note that one way of addressing this loss in data is to simply increase the intrinsic dimensionality of the solution. Using three instead of two factors, for example, would by definition result in a higher proportion of variance being retained. Remember, identifying the intrinsic dimensionality and, by extension, the number of factors to extract from the data is a heuristic process, which means that one value is not inherently better than any other. One means of identifying an acceptable intrinsic dimensionality involves ensuring that the resulting data lost does not exceed some threshold – in this case, 30%.

Now, although calculating these values using the rowSums() and colSums() function or even doing it manually is always an option, one benefit of the pca() function is that each of these metrics, as well as several others, are actually already included as part of the output.

```
pca(r = R, nfactors = 2, rotate = "varimax")
Principal Components Analysis

Call: principal(r = r, nfactors = nfactors, residuals = residuals,
    rotate = rotate, n.obs = n.obs, covar = covar, scores = scores,
    missing = missing, impute = impute,
    oblique.scores = oblique.scores,
    method = method, use = use, cor = cor, correct = 0.5,
    weight = NULL)

Standardized loadings (pattern matrix) based upon correlation matrix
                   RC1   RC2   h2   u2 com
Hopeless          0.74  0.36 0.67 0.33 1.4
Overwhelmed       0.75 -0.11 0.58 0.42 1.0
Exhausted         0.77 -0.05 0.59 0.41 1.0
Lonely            0.76  0.25 0.65 0.35 1.2
Sad               0.80  0.27 0.71 0.29 1.2
Depressed         0.69  0.48 0.70 0.30 1.8
Anxious           0.74  0.24 0.61 0.39 1.2
SelfHarming       0.16  0.78 0.64 0.36 1.1
SuicidalThoughts  0.23  0.82 0.73 0.27 1.2
SuicidalAttempts  0.00  0.80 0.64 0.36 1.0

                        RC1  RC2
SS loadings            4.02 2.50
Proportion Var         0.40 0.25
Cumulative Var         0.40 0.65
Proportion Explained   0.62 0.38
Cumulative Proportion  0.62 1.00

Mean item complexity =  1.2
Test of the hypothesis that 2 components are sufficient.

The root mean square of the residuals (RMSR) is  0.08

Fit based upon off diagonal values = 0.97
```

The segment of output immediately below the loading matrix (which we have in the past isolated using the $loadings[] operation) includes the sum of squared loadings for each factor, as well as both the proportion of variance and the cumulative proportion of variance explained by each factor. Note that these values are identical to those calculated earlier.

The Proportion Explained component of the output provides a measure of the relative amount of variance explained by the two factors. This is a useful metric for quickly determining the relative importance or benefit of each retained factor.

For our example data, the first factor accounts for 62% of the entire variance accounted for by the two-factor solution, whereas the second factor accounts for the remaining 38%.

A cumulative version of this metric is also presented in the `Cumulative Proportion` component of the output, but these values are rarely used in interpretation.

The loading matrix also contains some additional metrics that can be used in factor validation that were not retained when the loadings themselves were isolated earlier using the `$loadings[]` operation. The column immediately after the loadings or correlations, for example, contains the estimated communality $h^2$ for each dimension, which we have described earlier.

The next column contains the uniqueness $u^2$ of each dimension, which is simply the complement of the communality:

$$u_i^2 = 1 - h_i^2 \tag{2.30}$$

In our example data, the uniqueness of feeling `Hopeless` is 0.33, indicating that approximately 33% of the variance in this dimension was lost during factor extraction.

The final column contains *Hoffman's index of complexity* $c$ for each dimension, which denotes the number of factors needed to accurately represent the variability in that dimension:

$$c_i = \frac{\left(\sum_{j=1} {a_{ij}}^2\right)^2}{\sum_{j=1} {a_{ij}}^4} \tag{2.31}$$

For our example data, several dimensions have complexities of exactly 1.0, indicating that a single factor was sufficient to represent them, which is ideal in factor analysis. Most other dimensions had complexities below 1.3, which often corresponds to the presence of a single loading in excess of -0.3 or +0.3. Only two dimensions are above this threshold, which confirms our earlier conclusions that even after orthogonal rotation, two dimensions are still complex.

There are several other pieces of information that are included in the output of the `pca()` function; however, because these metrics are so rarely used in factor validation or subsequent analysis, we will not discuss them here. As before, we can explore these and other aspects of the output using the help documentation available via `?pca`.

### 2.4.3 | Reconstructing the Correlation Matrix

All of the metrics discussed thus far have provided a means of measuring the proportion of variance in each dimension separately or in the set of dimensions as a whole (on average) that is retained by the factor analysis. Although these measurements are critical in factor validation, they do not address how well these factors capture the covariance between dimensions.

One way to measure the proportion of covariance between dimensions that is retained after factor extraction is to use the resulting loading matrix to directly reconstruct the correlations

between the original dimensions. Remember, that the correlation is simply a standardized form of the covariance between dimensions that we are attempting to retain.

The *reconstructed correlation matrix* $\widehat{\boldsymbol{R}}$, also known as the *reproduced correlation matrix*, can be found by multiplying the loading matrix with its own transpose:

$$\widehat{\boldsymbol{R}} = \boldsymbol{A} \times \boldsymbol{A}^{T} \tag{2.32}$$

This equation is often referred to as the fundamental equation for factor analysis because it can be used to derive all other equations commonly used in the analysis.

It is interesting to note that because orthogonal rotation is used only to improve the interpretability of factors, this equation will yield identical results regardless of whether a rotated or unrotated loading matrix is used.

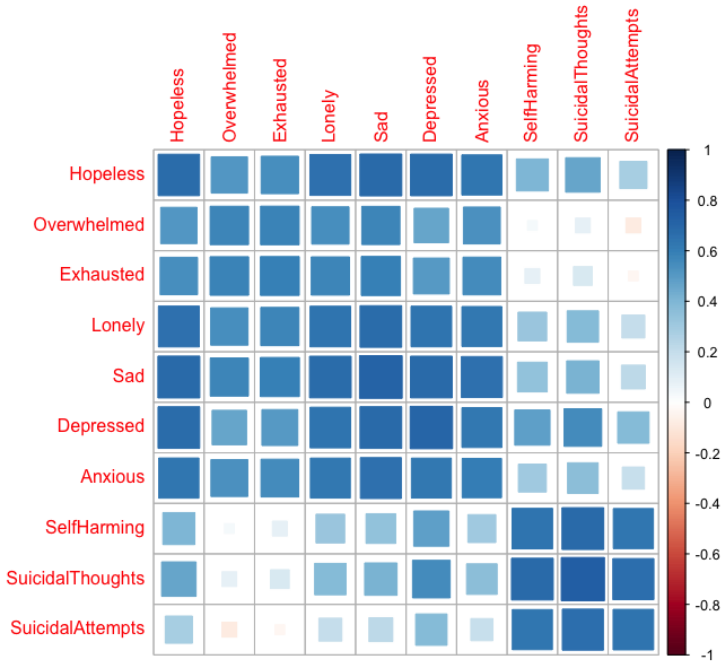
In R, we can use simple matrix algebra to calculate this reconstructed correlation matrix.

```
R_hat <- A %*% t(A)
round(R_hat, 2)
```

```
                  Hopeless Overwhelmed Exhausted  Lonely    Sad
Hopeless              0.67        0.52      0.55    0.65   0.69
Overwhelmed           0.52        0.58      0.58    0.55   0.57
Exhausted             0.55        0.58      0.59    0.57   0.60
Lonely                0.65        0.55      0.57    0.65   0.68
Sad                   0.69        0.57      0.60    0.68   0.71
Depressed             0.68        0.47      0.51    0.65   0.68
Anxious               0.63        0.53      0.56    0.63   0.66
SelfHarming           0.40        0.04      0.09    0.32   0.34
SuicidalThoughts      0.47        0.08      0.14    0.39   0.41
SuicidalAttempts      0.29       -0.09     -0.04    0.21   0.22
                  Depressed Anxious SelfHarming SuicidalThoughts
Hopeless               0.68    0.63        0.40             0.47
Overwhelmed            0.47    0.53        0.04             0.08
Exhausted              0.51    0.56        0.09             0.14
Lonely                 0.65    0.63        0.32             0.39
Sad                    0.68    0.66        0.34             0.41
Depressed              0.70    0.62        0.49             0.55
Anxious                0.62    0.61        0.31             0.37
SelfHarming            0.49    0.31        0.64             0.68
SuicidalThoughts       0.55    0.37        0.68             0.73
SuicidalAttempts       0.39    0.19        0.63             0.66
                  SuicidalAttempts
Hopeless                      0.29
Overwhelmed                  -0.09
Exhausted                    -0.04
Lonely                        0.21
Sad                           0.22
Depressed                     0.39
Anxious                       0.19
SelfHarming                   0.63
SuicidalThoughts              0.66
SuicidalAttempts              0.64
```

As with many correlation matrices, identifying any patterns in this matrix, even with rounding, is challenging given the sheer number of numerical values. As such, we often rely on a correlation plot or other visualization to investigate any such patterns.

```
corrplot(R_hat)
```




For our example data, we see clear clustering among the first seven dimensions and another cluster among the last three dimensions, which is similar to what we observed in the original correlation matrix (see Section 2.1).

These qualitative comparisons, however, are challenging to make if the number of dimensions in the original data matrix is high. Further, they do not provide a robust means of determining if a sufficient amount of the underlying covariance structure was retained following reconstruction.

The *residual correlation matrix* $\boldsymbol{R_{RES}}$ can be used to quantify the difference between the observed and the reconstructed correlation matrix:

$$\boldsymbol{R_{RES}} = \boldsymbol{R} - \widehat{\boldsymbol{R}} \tag{2.33}$$

Given that the values along the positive diagonal of the observed correlation matrix are, by definition, equal to one, the residuals along this diagonal don't provide much information in terms of data lost or retained. As such, it is common practice to replace these correlations with each dimension's communality. This ensures that the residuals along the positive diagonal, which correspond to the correlation of a dimensions with itself, are equal to zero.

In R, we can use the diag() function to isolate and change the values in the primary diagonal of our original correlation matrix prior to using it to calculate the residual correlation matrix.

```
R_adj <- R
diag(R_adj) <- rowSums(A^2)
R_res <- R_adj - R_hat
round(R_res, 2)
```

```
                  Hopeless Overwhelmed Exhausted  Lonely   Sad
Hopeless              0.00       -0.11     -0.12   -0.02 -0.02
Overwhelmed          -0.11        0.00      0.10   -0.13 -0.13
Exhausted            -0.12        0.10      0.00   -0.11 -0.11
Lonely               -0.02       -0.13     -0.11    0.00  0.02
Sad                  -0.02       -0.13     -0.11    0.02  0.00
Depressed             0.03       -0.13     -0.12   -0.03  0.00
Anxious              -0.06       -0.08     -0.10   -0.08 -0.05
SelfHarming          -0.05        0.10      0.08   -0.04 -0.04
SuicidalThoughts     -0.04        0.07      0.05   -0.04 -0.04
SuicidalAttempts     -0.07        0.13      0.13   -0.02 -0.02
                  Depressed Anxious SelfHarming SuicidalThoughts
Hopeless               0.03   -0.06       -0.05             -0.04
Overwhelmed           -0.13   -0.08        0.10              0.07
Exhausted             -0.12   -0.10        0.08              0.05
Lonely                -0.03   -0.08       -0.04             -0.04
Sad                    0.00   -0.05       -0.04             -0.04
Depressed              0.00    0.00       -0.08             -0.03
Anxious                0.00    0.00       -0.03             -0.04
SelfHarming           -0.08   -0.03        0.00             -0.09
SuicidalThoughts      -0.03   -0.04       -0.09              0.00
SuicidalAttempts      -0.10    0.00       -0.13             -0.10
                  SuicidalAttempts
Hopeless                     -0.07
Overwhelmed                   0.13
Exhausted                     0.13
Lonely                       -0.02
Sad                          -0.02
Depressed                    -0.10
Anxious                       0.00
SelfHarming                  -0.13
SuicidalThoughts             -0.10
SuicidalAttempts              0.00
```

For our example data, the difference between the observed and reconstructed correlation between feeling Overwhelmed and feeling Lonely is -0.13, indicating that the loading matrix overestimated this value. The correlation between feeling Overwhelmed and Exhausted, on the other hand, was underestimated by the reconstruction, with a residual of +0.10.

We note that every value in this residual correlation matrix is between -0.13 and +0.13, indicating a relatively close fit between observed and reconstructed values. Although there is no clear threshold regarding how much similarity is sufficient to avoid substantial data loss, residuals with absolute values of 0.10 or lower are generally preferred. When this threshold is not met in a high number of residual values, it may be worthwhile to consider increasing the number of extracted factors so as to capture more of the covariance structure in the data.

## Exercises for Section 2.4

**2.13 Cavity Trees and Fishers.** A cavity tree is a living or dead tree that has enough decay that a cavity or hole has developed in the tree. Most cavities are formed because of decay-causing fungi that gain access to the interior of the tree through cracks from fire and frost, branches breaking away from the tree bole, and woodpecker excavation. An article published in 2020 in the *Canadian Journal of Forest Research* investigated what factors lead to these cavity trees being used by fishers, a medium-sized carnivore native to the forests of North America, as den sites for the birthing and rearing of their young.

| | DBH | Type | Slope | Aspect |
|---|---|---|---|---|
| **DBH** | 1.00 | -0.73 | -0.09 | -0.38 |
| **Type** | -0.73 | 1.00 | 0.23 | 0.49 |
| **Slope** | -0.09 | 0.23 | 1.00 | -0.46 |
| **Aspect** | -0.38 | 0.49 | -0.46 | 1.00 |

The correlation matrix above was calculated using the `DBH` (diameter at breast height in cm) and `Type` (0 for coniferous and 1 for deciduous) of the tree itself, as well as the `Slope` (in degrees from horizontal) and `Aspect` (in degrees from due South) of the ground the tree is on for eight fisher dens throughout the Camp Ripley military facility near Little Falls, Minnesota.

a. Construct this correlation matrix in R with the proper row and column names
b. Find the intrinsic dimensionality using Kaiser's criterion
c. Find the optimally rotated loading matrix using the `"varimax"` algorithm
d. Explain if there appears to be any complexity in this rotated loading matrix
e. Use matrix algebra to find the communality for each of the original four dimensions
f. Explain how much of the variability in each of the original dimensions was lost due to dimensionality reduction and if there are any concerns regarding these losses

**2.14 Online Banking Performance (continued).** Exercise 2.07 introduced data on the cost efficiency focused operational orientations of 32 banks in Taiwan that were published in 2009 in *Computer and Operations Research*, which are recreated below:

| | PC1 | PC2 |
|---|---|---|
| **A** | 0.744 | -0.563 |
| **B** | 0.588 | -0.221 |
| **C** | 0.756 | -0.129 |
| **D** | 0.708 | -0.374 |
| **AB** | 0.739 | -0.574 |
| **BC** | 0.755 | -0.193 |
| **AC** | 0.802 | -0.409 |
| **AD** | 0.825 | -0.548 |
| **BD** | 0.842 | -0.422 |
| **CD** | 0.856 | -0.400 |
| **ABC** | 0.783 | -0.456 |
| **BCD** | 0.855 | -0.405 |

| | | |
|---|---|---|
| **ABD** | 0.830 | -0.525 |
| **ACD** | 0.850 | -0.460 |
| **ABCD** | 0.839 | -0.487 |

The loading matrix above includes the correlations between two extracted factors or principal components (PC) and dimensions related to the estimated efficiency of deposits (A), operational costs (B), number of employees (C), and equipment (D) at each bank in 2005.

a. Construct this loading matrix in R with the proper row and column names
b. Use matrix algebra to find the proportion of variance in the original data that is retained by each of the two factors
c. Find the cumulative proportion of variance that is retained by this two-factor solution
d. Explain if it appears that the second factor was indeed necessary to summarize the correlation structure between the original 15 dimensions

**2.15 Facework in Intercultural Communication (continued).** Exercise 2.04 introduced data on the mediating effects of situations factors on the use of avoiding and corrective facework during intercultural miscommunication in face-to-face contexts, which are recreated below:

| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
|---|---|---|---|---|---|---|---|---|---|---|
| **1. Self-positive face** | 1.0 | .35 | .36 | .19 | .12 | .18 | .21 | .23 | -.15 | -.13 |
| **2. Self-negative face** | .35 | 1.0 | .28 | .41 | .17 | .03 | .06 | .24 | .01 | -.12 |
| **3. Other-positive face** | .36 | .28 | 1.0 | .40 | .23 | .27 | .25 | .32 | -.21 | -.07 |
| **4. Other-negative face** | .19 | .41 | .40 | 1.0 | .24 | .24 | .27 | .21 | -.09 | .05 |
| **5. Severity** | .12 | .17 | .23 | .24 | 1.0 | .75 | .74 | -.01 | .34 | .29 |
| **6. Threatened positive face** | .18 | .03 | .27 | .24 | .75 | 1.0 | .78 | .12 | .12 | .26 |
| **7. Threatened negative face** | .21 | .06 | .25 | .27 | .74 | .78 | 1.0 | .17 | .06 | .24 |
| **8. Unintentional attribution** | .23 | .24 | .32 | .21 | -.01 | .12 | .17 | 1.0 | -.35 | -.27 |
| **9. Intentional attribution** | -.15 | .01 | -.21 | -.09 | .34 | .12 | .06 | -.35 | 1.0 | .27 |
| **10. Incidental attribution** | -.13 | -.12 | -.07 | .05 | .29 | .26 | .24 | -.27 | .27 | 1.0 |

The correlation matrix above includes the covariability between ten different situational factors that were measured on 103 undergraduate students who were U.S. citizens.

a. Construct this correlation matrix in R with the proper row and column names
b. Use the `pca()` function to find the full results of an optimally orthogonally rotated three-factor representation of these data
c. Explain if there appears to be any remaining complexity among the original dimensions using Hoffman's index of complexity
d. Describe any difference between this answer and the one you would derive from investigating only the values in the loading matrix

**2.16 Stock Portfolios (continued).** Exercise 1.13 introduced summary data on the performance of a diverse stock portfolio that spanned five different crypto currency stocks, which are recreated below:

| | SOL | BNB | ETH | BTC | DOGE |
|---|---|---|---|---|---|
| **SOL** | 3805 | 4980 | 42330 | 468665 | 0.354 |
| **BNB** | 4980 | 8293 | 64715 | 700917 | 1.401 |
| **ETH** | 42330 | 64715 | 559182 | 6294123 | 14.990 |
| **BTC** | 468665 | 700917 | 6294123 | 80834339 | 249.131 |
| **DOGE** | 0.354 | 1.401 | 14.990 | 249.131 | 0.005 |

The covariance matrix above was calculated using the daily adjusted closing values of five different crypto currency stocks over the course of a single year.

a. Construct this covariance matrix in R with the proper row and column names
b. Use matrix algebra to find the corresponding correlation matrix
c. Find the optimally rotated loading matrix using the "varimax" algorithm for an intrinsic dimensionality of two
d. Use matrix algebra to find the reconstructed correlation matrix
e. Find the residual correlation matrix
f. Describe if it appears that this two-factor representation has captured a sufficient amount of the covariance structure among the original dimensions