



Evaluation of Formative Measurement Models

Contents

- 5.1 Convergent Validity – 92**
- 5.2 Indicator Collinearity – 93**
- 5.3 Statistical Significance
and Relevance of the Indicator
Weights – 93**
- 5.4 Case Study Illustration:
Formative Measurement
Models – 97**
 - 5.4.1 Model Setup and Estimation – 97
 - 5.4.2 Reflective Measurement Model
Evaluation – 102
 - 5.4.3 Formative Measurement Model
Evaluation – 102
- References – 111**

Learning Objectives

After reading this chapter, you should understand:

1. The concept of redundancy analysis and how to apply it to evaluate convergent validity
2. Collinearity, its implications, and how to assess it
3. Significance testing using bootstrapping and bootstrap confidence intervals
4. How to assess formative measurement models using SEMinR

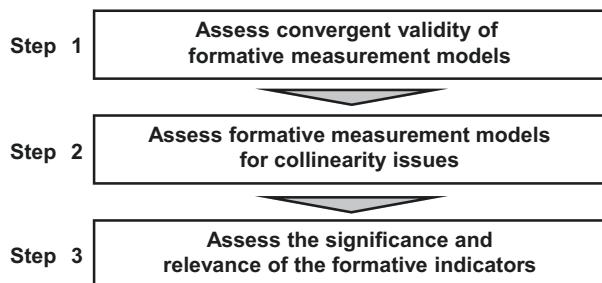
5

PLS-SEM is the preferred approach when formatively specified constructs are included in the PLS path model (Hair, Risher, Sarstedt, & Ringle, 2019). In this chapter, we discuss the key steps for evaluating **formative measurement models** (■ Fig. 5.1). Relevant criteria include the assessment of (1) convergent validity, (2) indicator collinearity, and (3) statistical significance and relevance of the indicator weights. In the following, we introduce key criteria and their thresholds and illustrate their use with an extended version of the corporate reputation model.

5.1 Convergent Validity

In formative measurement model evaluation, **convergent validity** refers to the degree to which the formatively specified construct correlates with an alternative reflectively measured variable(s) of the same concept. Originally proposed by Chin (1998), the procedure is referred to as **redundancy analysis**. To execute this procedure for determining convergent validity, researchers must plan ahead in the research design stage by including an alternative measure of the formatively measured construct in their questionnaire. Cheah, Sarstedt, Ringle, Ramayah, and Ting (2018) show that a global single item, which captures the essence of the construct under consideration, is generally sufficient as an alternative measure – despite limitations with regard to criterion validity (Diamantopoulos, Sarstedt, Fuchs, Wilczynski, & Kaiser, 2012; Sarstedt, Diamantopoulos, Salzberger, & Baumgartner, 2016). When the model is based on secondary data, a variable measuring a similar

■ Fig. 5.1 Formative measurement model assessment procedure. (Source: Hair, Hult, Ringle, & Sarstedt, 2022, Chap. 5; used with permission by Sage)



concept would be used (Houston, 2004). Hair et al. (2022) suggest the correlation of the formatively measured construct with the reflectively measured item(s) should be 0.708 or higher, which implies that the construct explains (more than) 50% of the alternative measure's variance.

5.2 Indicator Collinearity

Collinearity occurs when two or more indicators in a formative measurement model are highly correlated. High correlation increases the **standard error** of the indicator weights, thereby triggering type II errors (i.e., false negatives). More pronounced levels of collinearity can even trigger sign changes in the indicator weights, which leads to interpretational confounding. For example, a collinearity-induced sign change might lead to a negative weight in an indicator measuring an aspect of corporate performance such as “[the company] is a very well-managed company.” Such a sign change would imply the better the respondents' assessment of the company's management, the lower its perceived performance. This type of result is inconsistent with a priori assumptions and is particularly counterintuitive when the correlation between the construct and the indicator is in fact positive. The standard metric for assessing indicator collinearity is the **variance inflation factor (VIF)**. When **VIF** values are higher, the level of collinearity is greater. **VIF** values of 5 or above indicate collinearity problems. In this case, researchers should take adequate measures to reduce the collinearity level, for example, by eliminating or merging indicators or establishing a **higher-order construct** – see Hair et al. (2022, Chap. 5). However, collinearity issues can also occur at lower **VIF** values of 3 (Becker, Ringle, Sarstedt, & Völckner, 2015; Mason & Perreault, 1991). Hence, when the analysis produces unexpected sign changes in the indicator weights, the initial step is to compare the sign of the relationship using bivariate correlation. If the relationship sign differs from the correlation sign, researchers should revise the model setup, also by eliminating or merging indicators or establishing a higher-order construct.

5.3 Statistical Significance and Relevance of the Indicator Weights

The third step in assessing formatively measured constructs is examining the statistical significance and relevance (i.e., size) of the indicator weights. The **indicator weights** result from regressing each formatively measured construct on its associated indicators. As such, they represent each indicator's relative importance for forming the construct. **Significance testing** of the indicator weights relies on the bootstrapping procedure, which facilitates deriving standard errors from the data without relying on any distributional assumptions (Hair, Sarstedt, Hopkins, & Kuppelwieser, 2014).

Excuse

In **bootstrapping**, a large number of samples (i.e., **bootstrap samples**) are drawn from the original sample, with replacement (Davison & Hinkley, 1997). The number of bootstrap samples should be high but must be at least equal to the number of valid observations in the dataset. Reviewing prior research on bootstrapping implementations, Streukens and Leroi-Werelds (2016) recommend that PLS-SEM applications should be based on at least 10,000 bootstrap samples. The bootstrap samples are used

to estimate the PLS path model 10,000 times. The resulting parameter estimates, such as the indicator weights or path coefficients, form a bootstrap distribution that can be viewed as an approximation of the sampling distribution. Based on this distribution, it is possible to calculate the standard error, which is the standard deviation of the estimated coefficients across bootstrap samples. Using the standard error as input, we can evaluate the statistical significance of the model parameters.

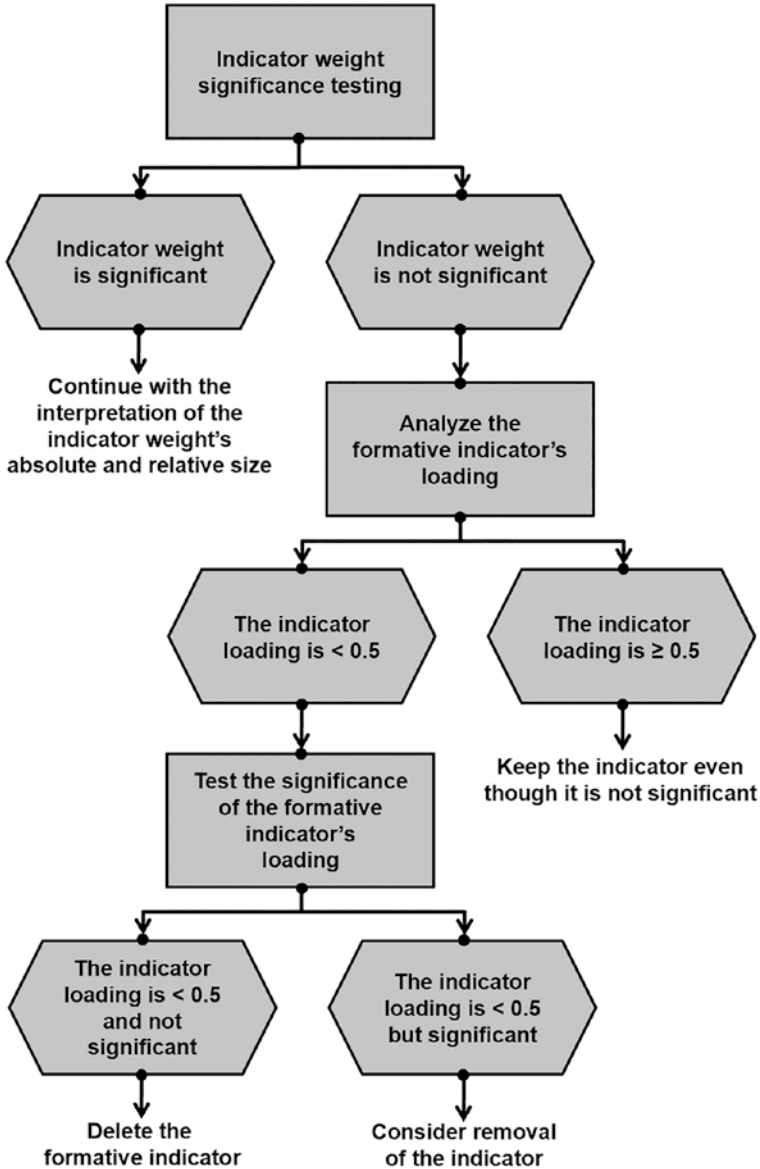
5

The bootstrapping procedure yields *t*-values for the indicator weights (and other model parameters). We need to compare these *t*-values with the **critical values** from the standard normal distribution to decide whether the coefficients are significantly different from zero. Assuming a significance level of 5%, a *t*-value above 1.96 (two-tailed test) suggests that the indicator weight is statistically significant. The critical values for significance levels of 1% ($\alpha = 0.01$) and 10% ($\alpha = 0.10$) probability of error are 2.576 and 1.645 (two tailed), respectively.

Confidence intervals are an alternative way to test for the significance of indicator weights. They represent the range within which the population parameter will fall assuming a certain level of confidence (e.g., 95%). In the PLS-SEM context, we also refer to **bootstrap confidence intervals** because the construction of the confidence interval is inferred from the estimates generated by the bootstrapping process (Henseler, Ringle, & Sinkovics, 2009). Several types of confidence intervals have been proposed in the context of PLS-SEM – see Hair et al. (2022, Chap. 5) for an overview. Results from Aguirre-Urreta and Rönkkö (2018) indicate the **percentile method** is preferred, as it exceeds other methods in terms of coverage and balance, producing comparably narrow confidence intervals. If a confidence interval does not include the value zero, the weight can be considered statistically significant, and the indicator can be retained. On the contrary, if the confidence interval of an indicator weight includes zero, this indicates the weight is not statistically significant (assuming the given significance level, e.g., 5%). In such a situation, the indicator should be considered for removal from the measurement model.

However, if an indicator weight is not significant, it is not necessarily interpreted as evidence of poor measurement model quality. We recommend you also consider the **absolute contribution** of a formative indicator to the construct (Cenfetelli & Bassellier, 2009), which is determined by the formative indicator's loading. At a minimum, a formative indicator's loading should be statistically significant. Indicator loadings of 0.5 and higher suggest the indicator makes a sufficient absolute contribution to forming the construct, even if it lacks a significant **relative contribution**. ■ Figure 5.2 shows the decision-making process for testing formative indicator weights.

5.3 · Statistical Significance and Relevance of the Indicator Weights



■ Fig. 5.2 Decision-making process for keeping or deleting formative indicators. (Source: Hair et al., 2022, Chap. 5; used with permission by Sage)

When deciding whether to delete formative indicators based on statistical outcomes, researchers need to be cautious for the following reasons. First, formative indicator weights are a function of the number of indicators used to measure a construct. The greater the number of indicators, the lower their average weight. Formative measurement models are inherently limited in the number of indicator weights that can be statistically significant (e.g., Cenfetelli & Bassellier, 2009). Second, indicators should seldom be removed from formative measurement models since formative measurement requires the indicators to fully capture the entire domain of a construct, as defined by the researcher in the conceptualization stage. In contrast to reflective measurement models, formative indicators are not interchangeable, and removing even one indicator can therefore reduce the measurement model's content validity (Bollen & Diamantopoulos, 2017).

► Important

Formative indicators with nonsignificant weights should not automatically be removed from the measurement model, since this step may compromise the content validity of the construct.

After the statistical significance of the formative indicator weights has been assessed, the final step is to examine each indicator's relevance. With regard to relevance, indicator weights are standardized to values between -1 and $+1$. Thus, indicator weights closer to $+1$ (or -1) indicate strong positive (or negative) relationships, and weights closer to 0 indicate relatively weak relationships. ■ Table 5.1 summarizes the rules of thumb for formative measurement model assessment.

■ Table 5.1 Rules of thumb for formative measurement model assessment

Criterion	Metrics and thresholds
Convergent validity (redundancy analysis)	≥ 0.708 correlation between the formative construct and a reflective (or single-item) measurement of the same concept
Collinearity	Critical collinearity issues likely occur if VIF ≥ 5 Collinearity issues are usually uncritical if VIF = 3–5 Collinearity is not a problematic issue if VIF < 3
Statistical significance of indicator weights	t -values are greater than 2.576 ($\alpha = 0.01$), 1.960 ($\alpha = 0.05$), or 1.645 ($\alpha = 0.10$), respectively (two tailed) The 95% percentile confidence interval ($\alpha = 0.05$) does not include zero
Relevance of indicators with a significant weight	Larger significant indicator weights indicate a higher relative contribution of the indicator to the construct
Relevance of indicators with nonsignificant weights	Indicators with loadings of ≥ 0.50 that are statistically significant are considered relevant

Source: authors' own table

5.4 Case Study Illustration: Formative Measurement Models

5.4.1 Model Setup and Estimation

The simple corporate reputation model introduced in ► Chap. 3 (► Fig. 3.2) and evaluated in ► Chap. 4 describes the relationships between the two dimensions of corporate reputation (i.e., competence and likeability) as well as the two key target constructs (i.e., customer satisfaction and loyalty). While the simple model is useful to explain how corporate reputation affects customer satisfaction and customer loyalty, it does not indicate how companies can effectively manage (i.e., improve) their corporate reputation. Schwaiger (2004) identified four driver constructs of corporate reputation that companies can manage by means of corporate-level marketing activities. ■ Table 5.2 lists and defines the four driver constructs of corporate reputation.

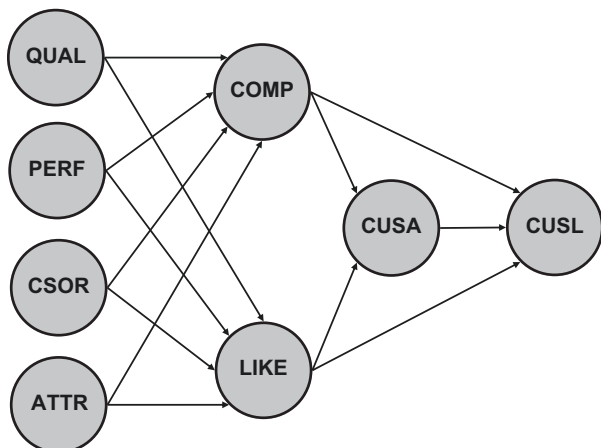
All four driver constructs are (positively) related to the competence and likeability dimensions of corporate reputation in the path model. ■ Figure 5.3 shows

■ **Table 5.2** The driver constructs of corporate reputation

Construct name	Construct definition
QUAL	The quality of a company's products and services as well as its quality of customer orientation
PERF	The company's economic and managerial performance
CSOR	The company's corporate social responsibility
ATTR	The company's attractiveness as an employer

Source: authors' own table

■ **Fig. 5.3** The extended corporate reputation model. (Source: Hair et al., 2022, Chap. 5; used with permission by Sage)



the constructs and their relationships, which represent the extended structural model for our PLS-SEM example in the remaining chapters of the book. To summarize, the extended corporate reputation model has three main conceptual/theoretical components:

1. The target constructs of interest (*CUSA* and *CUSL*)
2. The two corporate reputation dimensions, *COMP* and *LIKE*, that represent key determinants of the target constructs
3. The four exogenous driver constructs (i.e., *ATTR*, *CSOR*, *PERF*, and *QUAL*) of the two corporate reputation dimensions

5

The endogenous constructs on the right-hand side in **Fig. 5.3** include a single-item construct (i.e., *CUSA*) and three reflectively measured constructs (i.e., *COMP*, *CUSL*, and *LIKE*). In contrast, the four new driver constructs (i.e., exogenous latent variables) on the left-hand side of **Fig. 5.3** (i.e., *ATTR*, *CSOR*, *PERF*, and *QUAL*) have formative measurement models in accordance with their role in the reputation model (Schwaiger, 2004). Specifically, the four new constructs are measured by a total of 21 formative indicators (detailed in **Table 5.3**) that have been derived from literature, qualitative studies, and quantitative pretests (for more details, see Schwaiger, 2004). **Table 5.3** also lists the single-item reflective global measures for validating the formative driver constructs when executing the redundancy analysis.

We continue to use the `corp_rep_data` dataset with 344 observations introduced in **Chap. 3** for our PLS-SEM analyses. Unlike in the simple model that was used in the previous chapter, we now also have to consider the formative measurement models when deciding on the minimum sample size required to estimate the model. The maximum number of arrowheads pointing at a particular construct occurs in the measurement model of *QUAL*. All other formatively measured constructs have fewer indicators. Similarly, there are fewer arrows pointing at each of the endogenous constructs in the structural model. Therefore, when building on the 10-time rule of thumb, we would need $8 \cdot 10 = 80$ observations. Alternatively, following Cohen's (1992) recommendations for multiple ordinary least squares regression analysis or running a power analysis using the G*Power program (Faul, Erdfelder, Buchner, & Lang, 2009), we would need only 54 observations to detect R^2 values of around 0.25, assuming a significance level of 5% and a statistical power of 80%. When considering the more conservative approach suggested by Kock and Hadaya (2018), we obtain a higher minimum sample size. Considering prior research on the corporate reputation model, we expect a minimum path coefficient of 0.15 in the structural model. Assuming a significance level of 5% and statistical power of 80%, the inverse square root method yields a minimum sample size of approximately 155 (see **Chap. 1** for a discussion of sample size and power considerations).

The corporate reputation data can be accessed by the object name `corp_rep_data`:

```
# Load the SEMinR library
library(semInR)
# Load the corporate reputation data
corp_rep_data <- corp_rep_data
```


Table 5.3 The indicators of the formatively measured constructs

Quality (<i>QUAL</i>)	
<i>qual_1</i>	The products/services offered by [the company] are of high quality
<i>qual_2</i>	[The company] is an innovator, rather than an imitator with respect to [industry]
<i>qual_3</i>	[The company]'s products/services offer good value for money
<i>qual_4</i>	The products/services offered by [the company] are good
<i>qual_5</i>	Customer concerns are held in high regard at [the company]
<i>qual_6</i>	[The company] is a reliable partner for customers
<i>qual_7</i>	[The company] is a trustworthy company
<i>qual_8</i>	I have a lot of respect for [the company]
Performance (<i>PERF</i>)	
<i>perf_1</i>	[The company] is a very well-managed company
<i>perf_2</i>	[The company] is an economically stable company
<i>perf_3</i>	The business risk for [the company] is modest compared to its competitors
<i>perf_4</i>	[The company] has growth potential
<i>perf_5</i>	[The company] has a clear vision about the future of the company
Corporate social responsibility (<i>CSOR</i>)	
<i>csor_1</i>	[The company] behaves in a socially conscious way
<i>csor_2</i>	[The company] is forthright in giving information to the public
<i>csor_3</i>	[The company] has a fair attitude toward competitors
<i>csor_4</i>	[The company] is concerned about the preservation of the environment
<i>csor_5</i>	[The company] is not only concerned about profits
Attractiveness (<i>ATTR</i>)	
<i>attr_1</i>	[The company] is successful in attracting high-quality employees
<i>attr_2</i>	I could see myself working at [the company]
<i>attr_3</i>	I like the physical appearance of [the company] (company, buildings, shops, etc.)
Single-item measures of <i>QUAL</i>, <i>PERF</i>, <i>CSOR</i>, and <i>ATTR</i> for the redundancy analysis	
<i>qual_global</i>	Please assess the overall quality of [the company's] activities
<i>perf_global</i>	Please assess [the company's] overall performance
<i>csor_global</i>	Please assess the extent to which [the company] acts in socially conscious ways
<i>attr_global</i>	[The company] has a high overall attractiveness

Source: Hair et al., 2022, Chap. 5; used with permission by Sage

Excuse

The corporate reputation data file and project are also bundled with SEMinR. Once the SEMinR library has been loaded, we can access the demonstration code for ► Chap. 5 by using the `demo()` function on the object "`seminr-primer-chap5`".

5

The extended corporate reputation model's structural and measurement models will have to be specified using the SEMinR syntax. Remember that the four drivers are formative constructs, estimated with `mode_B`, while *COMP*, *CUSL* and *LIKE* are reflective constructs, estimated with `mode_A`. The `weights` parameter of the `composite()` function is set by default to `mode_A`. Thus, when no weights are specified, the construct is estimated as being reflective. Alternatively, we can explicitly specify the `mode_A` setting for reflectively measured constructs or the `mode_B` setting for formatively measured constructs. Once the model is set up, we use the `estimate_pls()` function to estimate the model, this time specifying the `measurement_model` and `structural_model` parameters to the extended corporate reputation model objects (`corp_rep_mm_ext`, `corp_rep_sm_ext`). Finally, we apply the `summary()` function to the estimated SEMinR model object `corp_rep_pls_model_ext` and store the output in the `summary_corp_rep_ext` object:

```
# Create measurement model
corp_rep_mm_ext <- constructs(
  composite("QUAL", multi_items("qual_", 1:8), weights =
mode_B),
  composite("PERF", multi_items("perf_", 1:5), weights =
mode_B),
  composite("CSOR", multi_items("csor_", 1:5), weights =
mode_B),
  composite("ATTR", multi_items("attr_", 1:3), weights =
mode_B),
  composite("COMP", multi_items("comp_", 1:3)),
  composite("LIKE", multi_items("like_", 1:3)),
  composite("CUSA", single_item("cusa")),
  composite("CUSL", multi_items("cusl_", 1:3))
)
# Create structural model
corp_rep_sm_ext <- relationships(
  paths(from = c("QUAL", "PERF", "CSOR", "ATTR"), to =
c("COMP", "LIKE")),
  paths(from = c("COMP", "LIKE"), to = c("CUSA", "CUSL")),
  paths(from = c("CUSA"), to = c("CUSL"))
)
# Estimate the model
corp_rep_pls_model_ext <- estimate_pls(
  data = corp_rep_data,
  measurement_model = corp_rep_mm_ext,
  structural_model = corp_rep_sm_ext,
```

```

missing = mean_replacement,
missing_value = "-99")
# Summarize the model results
summary_corp_rep_ext <- summary(corp_rep_pls_model_ext)

```

Just like the indicator data that we used in previous chapters, the `corp_rep_data` dataset has very few missing values. The number of missing observations is reported in the descriptive statistic object nested within the summary return object. This report can be accessed by inspecting the `summary_corp_rep_ext$descriptives$statistics` object. Only the indicators `cusl_1` (three missing values, 0.87% of all responses on this indicator), `cusl_2` (four missing values, 1.16% of all responses on this indicator), `cusl_3` (three missing values, 0.87% of all responses on this indicator), and `cusa` (one missing value, 0.29% of all responses on this indicator) include missing values. Since the number of missing values is relatively small (i.e., less than 5% missing values per indicator; Hair et al., 2022, Chap. 2), we use mean value replacement to deal with missing data when running the PLS-SEM algorithm (see also Grimm & Wagner, 2020).

When the PLS-SEM algorithm stops running, check whether the algorithm converged (Hair et al., 2022, Chap. 3). For this example, the PLS-SEM algorithm will stop when the maximum number of 300 iterations or the stop criterion of 1.0E-7 (i.e., 0.0000001) is reached. To do so, it is necessary to inspect the `corp_rep_pls_model` object by using the `$` operator:

```

# Iterations to converge
summary_corp_rep_ext$iterations

```

The results show that the model estimation converged after eight iterations. Next, the model must be bootstrapped to assess the indicator weights' significance. For now, we run a simple bootstrap as conducted in ► Chap. 4. But in this chapter, we discuss the bootstrap function in further detail when assessing the formative indicator weights' significance. To run the bootstrapping procedure in SEMinR, we use the `bootstrap_model()` function and assign the output to a variable; we call our variable `boot_corp_rep_ext`. Then, we run the `summary()` function on the `boot_corp_rep` object and assign it to another variable, such as `sum_boot_corp_rep_ext`.

```

# Bootstrap the model
boot_corp_rep_ext <- bootstrap_model(
  seminr_model = corp_rep_pls_model_ext, nboot = 1000)
# Store the summary of the bootstrapped model
sum_boot_corp_rep_ext <- summary(boot_corp_rep_ext,
  alpha = 0.10)

```

5.4.2 Reflective Measurement Model Evaluation

An important characteristic of PLS-SEM is that the model estimates will change when any of the model relationships or variables are changed. We thus need to reassess the reflective measurement models to ensure that this portion of the model remains valid and reliable before continuing to evaluate the four new exogenous formative constructs. We then follow the reflective measurement model assessment procedure in ► Fig. 4.1 (for a refresher on this topic, return to ► Chap. 4). The reflectively measured constructs meet all criteria as discussed in ► Chap. 4 – for a detailed discussion of the assessment of reflectively measured constructs for this model, see Appendix B.

5

5.4.3 Formative Measurement Model Evaluation

To evaluate the formatively measured constructs of the extended corporate reputation model, we follow the formative measurement model assessment procedure (► Fig. 5.1). First, we need to examine whether the formatively measured constructs exhibit convergent validity. To do so, we need to carry out a separate redundancy analysis for each construct. The original survey contained global single-item measures with generic assessments of the four concepts – attractiveness, corporate social responsibility, performance, and quality – that we can use as measures of the dependent construct in the redundancy analyses (*attr_global*, *csor_global*, *perf_global*, and *qual_global*) (► Table 5.3). Note that when designing a research study that includes formatively measured constructs, you need to include this type of global measure in the survey. ► Figure 5.4 shows the model set-ups for the redundancy analyses of the four formatively measured constructs in the extended corporate reputation model.

```
# Redundancy analysis
# ATTR
# Create measurement model
ATTR_redundancy_mm <- constructs(
  composite("ATTR_F", multi_items("attr_", 1:3), weights =
mode_B),
  composite("ATTR_G", single_item("attr_global"))
)
# Create structural model
ATTR_redundancy_sm <- relationships(
  paths(from = c("ATTR_F"), to = c("ATTR_G"))
)
# Estimate the model
ATTR_redundancy_pls_model <- estimate_pls(
  data = corp_rep_data,
  measurement_model = ATTR_redundancy_mm,
  structural_model = ATTR_redundancy_sm,
  missing = mean_replacement,
  missing_value = "-99")
# Summarize the model
sum_ATTR_red_model <- summary(ATTR_redundancy_pls_model)
# CSOR
```

5.4 • Case Study Illustration: Formative Measurement Models

```

# Create measurement model
CSOR_redundancy_mm <- constructs(
  composite("CSOR_F", multi_items("csor_", 1:5), weights =
mode_B),
  composite("CSOR_G", single_item("csor_global"))
)
# Create structural model
CSOR_redundancy_sm <- relationships(
  paths(from = c("CSOR_F"), to = c("CSOR_G"))
)
# Estimate the model
CSOR_redundancy_pls_model <- estimate_pls(
  data = corp_rep_data,
  measurement_model = CSOR_redundancy_mm,
  structural_model = CSOR_redundancy_sm,
  missing = mean_replacement,
  missing_value = "-99")
# Summarize the model
sum_CSOR_red_model <- summary(CSOR_redundancy_pls_model)
# PERF
# Create measurement model
PERF_redundancy_mm <- constructs(
  composite("PERF_F", multi_items("perf_", 1:5), weights =
mode_B),
  composite("PERF_G", single_item("perf_global"))
)
# Create structural model
PERF_redundancy_sm <- relationships(
  paths(from = c("PERF_F"), to = c("PERF_G"))
)
# Estimate the model
PERF_redundancy_pls_model <- estimate_pls(
  data = corp_rep_data,
  measurement_model = PERF_redundancy_mm,
  structural_model = PERF_redundancy_sm,
  missing = mean_replacement,
  missing_value = "-99")
# Summarize the model
sum_PERF_red_model <- summary(PERF_redundancy_pls_model)

# QUAL
# Create measurement model
QUAL_redundancy_mm <- constructs(
  composite("QUAL_F", multi_items("qual_", 1:8), weights =
mode_B),
  composite("QUAL_G", single_item("qual_global"))
)
# Create structural model
QUAL_redundancy_sm <- relationships(
  paths(from = c("QUAL_F"), to = c("QUAL_G"))
)
# Estimate the model
QUAL_redundancy_pls_model <- estimate_pls(

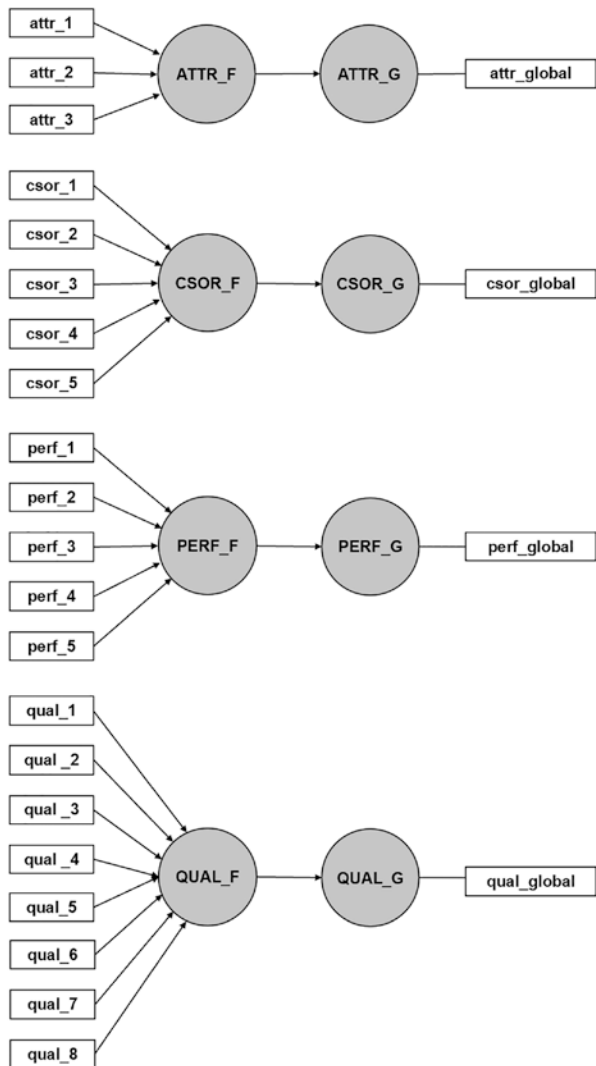
```

```

data = corp_rep_data,
measurement_model = QUAL_redundancy_mm,
structural_model = QUAL_redundancy_sm,
missing = mean_replacement,
missing_value = "-99")
# Summarize the model
sum_QUAL_red_model <- summary(QUAL_redundancy_pls_model)
# Check the path coefficients for convergent validity
sum_ATTR_red_model$paths
sum_CSOR_red_model$paths
sum_PERF_red_model$paths
sum_QUAL_red_model$paths
    
```

5

■ Fig. 5.4 Redundancy analysis of formatively measured constructs. (Source: authors' own figure)



In order to run the redundancy analysis for a formatively measured construct, it must be linked with an alternative measure of the same concept. When considering the formatively measured construct *ATTR*, the measurement model for the redundancy analysis consists of two constructs: (1) *ATTR_F*, which is measured by three formative indicators *attr_1*, *attr_2*, and *attr_3*, and (2) *ATTR_G*, which is measured by the single item *attr_global*. The structural model consists of a single path from *ATTR_F* to *ATTR_G*. We then estimate this model using the `corp_rep_data` dataset and assign the output to the `ATTR_redundancy_pls_model` object. Finally, to identify the path between the two constructs, we need to inspect the `sum_ATTR_red_model$paths`.

- ❗ Each redundancy analysis model is included in the SEMinR demo file accessible at `demo ("seminr-primer-chap5")`, so that the code can easily be replicated. Alternatively, we can create these four models for the convergent validity assessment manually using the code outlined above. Following the steps described in previous chapters, a new structural and measurement model must be created using the SEMinR syntax for each redundancy analysis, and the subsequently estimated model object needs to be inspected for the path coefficients.
- Figure 5.5 shows the results for the redundancy analysis of the four formatively measured constructs. For the *ATTR* construct, this analysis yields a path coefficient of 0.874, which is above the recommended threshold of 0.708 (■ Table 5.1),

```

Console Terminal x Jobs x
~/seminr/
> # Check the path coefficients for convergent validity
> sum_ATTR_red_model$paths
      ATTR_G
R^2      0.764
AdjR^2   0.763
ATTR_F   0.874
> sum_CSOR_red_model$paths
      CSOR_G
R^2      0.735
AdjR^2   0.734
CSOR_F   0.857
> sum_PERF_red_model$paths
      PERF_G
R^2      0.657
AdjR^2   0.656
PERF_F   0.811
> sum_QUAL_red_model$paths
      QUAL_G
R^2      0.648
AdjR^2   0.647
QUAL_F   0.805
> |

```

■ Fig. 5.5 Output of the redundancy analysis for formative measurement models. (Source: authors' screenshot from R)

thus providing support for the formatively measured construct's convergent validity. The redundancy analyses of *CSOR*, *PERF*, and *QUAL* yield estimates of 0.857, 0.811, and 0.805, respectively. Thus, all formatively measured constructs exhibit convergent validity.

In the second step of the assessment procedure (■ Fig. 5.1), we check the formative measurement models for collinearity by looking at the formative indicators' **VIF** values. The `summary_corp_rep_ext` object can be inspected for the indicator **VIF** values by considering the `validity` element for `vif_items`; `summary_corp_rep_ext$validity$vif_items`.

5

```
# Collinearity analysis
summary_corp_rep_ext$validity$vif_items
```

Note that **SEMinR** also provides **VIF** values for reflective indicators. However, since we expect high correlations among reflective indicators, we do not interpret these results but focus on the formative indicators' **VIF** values.

According to the results in ■ Fig. 5.6, *qual_3* has the highest **VIF** value (2.269). Hence, all **VIF** values are uniformly below the conservative threshold value of 3 (■ Table 5.1). We therefore conclude that collinearity does not reach critical levels in any of the formative measurement models and is not an issue for the estimation of the extended corporate reputation model.

Next, we need to analyze the indicator weights for their significance and relevance (■ Fig. 5.1). We first consider the significance of the indicator weights by means of bootstrapping. To run the bootstrapping procedure, we use the `bootstrap_model()` function. The first parameter (i.e., `seminr_model`) allows specifying the model on which we apply bootstrapping. The second parameter `nboot` allows us to select the number of bootstrap samples to use. Per default, we should use 10,000 bootstrap samples (Streukens & Leroi-Werelds, 2016). Since using such a great number of samples requires much computational time, we may choose a smaller number of samples (e.g., 1,000) for the initial model estimation. For the final result reporting, however, we should use the recommended number of 10,000 bootstrap samples.

The `cores` parameter enables us to use multiple cores of your computer's central processing unit (CPU). We recommend using this option since it makes bootstrapping much faster. As you might not know the number of cores in your device, we recommend using the `parallel::detectCores()` function to automatically detect the number of cores and use the maximum cores available. By default, `cores` will be set to the maximum value and as such, if you do not specify this parameter, your bootstrap will default to using the maximum computing power of your CPU. Finally, `seed` allows reproducing the results of a specific bootstrap run while maintaining the random nature of the process. Assign the output of the `bootstrap_model()` function to the `boot_corp_rep_ext` object. Finally, we need to run the `summary()` function on the `boot_corp_rep_ext` object and set the `alpha` parameter. The `alpha` parameter allows selecting the significance level (the default


```

~/seminr/
> # Collinearity analysis ----
> summary_corp_rep_ext$validity$vif_items
QUAL :
qual_1 qual_2 qual_3 qual_4 qual_5 qual_6 qual_7 qual_8
1.806 1.632 2.269 1.957 2.201 2.008 1.623 1.362

PERF :
perf_1 perf_2 perf_3 perf_4 perf_5
1.560 1.506 1.229 1.316 1.331

CSOR :
csor_1 csor_2 csor_3 csor_4 csor_5
1.560 1.487 1.735 1.556 1.712

ATTR :
attr_1 attr_2 attr_3
1.275 1.129 1.264

COMP :
comp_1 comp_2 comp_3
1.397 1.787 1.888

LIKE :
like_1 like_2 like_3
1.945 2.000 1.811

CUSA :
cusa
1

CUSL :
cusl_1 cusl_2 cusl_3
1.802 2.564 1.933

> |

```

■ Fig. 5.6 VIF values. (Source: authors' screenshot from R)

is 0.05) for two-tailed testing. When testing indicator weights, we follow general convention and apply two-tailed testing at a significance level of 5%.

```

# Bootstrap the model
# seminr_model is the SEMinR model to be bootstrapped
# nboot is the number of bootstrap iterations to run
# cores is the number of cpu cores to use
# in multicore bootstrapping
# parallel::detectCores() allows for using
# the maximum cores on your device
# seed is the seed to be used for making bootstrap replicable


```




```

boot_corp_rep_ext <- bootstrap_model(
  semnr_model = corp_rep_pls_model_ext,
  nboot = 1000,
  cores = parallel::detectCores(),
  seed = 123)
# Summarize the results of the bootstrap
# alpha sets the specified level for significance, i.e. 0.05
sum_boot_corp_rep_ext <- summary(boot_corp_rep_ext, alpha =
0.05)
# Inspect the bootstrapping results for indicator weights
sum_boot_corp_rep_ext$bootstrapped_weights


```

5

At this point in the analysis, we are only interested in the significance of the indicator weights and therefore consider only the measurement model. We thus inspect the `sum_boot_corp_rep_ext$bootstrapped_weights` object to obtain the results in  Fig. 5.7.

 Figure 5.7 shows t -values for the measurement model relationships produced by the bootstrapping procedure. Note that bootstrapped values are generated for all measurement model weights, but we only consider the indicators of the formative constructs. The original estimate of an indicator weight (shown in the second column, **Original Est.**;  Fig. 5.7) divided by the bootstrap standard error, which equals the bootstrap standard deviation (column: **Bootstrap SD**), for that indicator weight results in its empirical t -value as displayed in the third-to-last column in  Fig. 5.7 (column: **T Stat.**). Recall that the critical values for significance levels of 1% ($\alpha = 0.01$), 5% ($\alpha = 0.05$), and 10% ($\alpha = 0.10$) probability of error are 2.576, 1.960, and 1.645 (two tailed), respectively.

Attention

The bootstrapping results shown in  Fig. 5.7 will differ from your results. A **seed** is used in random computational processes to make the random process reproducible. However, note that for the same seed, different hardware and software combinations will generate different results. The important feature of the seed is that it ensures that the results are replicable on your computer or on computers with a similar hardware and software setup. Recall that bootstrapping builds on randomly drawn samples, so each time you run the bootstrapping routine with a different seed, different samples will be drawn. The differences become very small, however, if the number of bootstrapping samples is sufficiently large (e.g., 10,000).

The bootstrapping result report also provides bootstrap confidence intervals using the percentile method (Hair et al., 2022; Chap. 5). The lower boundary of the 95% confidence interval (**2.5% CI**) is displayed in the second-to-last column, whereas the upper boundary of the confidence interval (**97.5% CI**) is shown in the last column. We can readily use these confidence intervals for significance testing. Specifically, a null hypothesis H_0 that a certain parameter, such as an indicator weight w_1 , equals zero (i.e., $H_0: w_1 = 0$) in the population is rejected at a given level

```

> # Inspect the bootstrapping results for outer weights
> sum Boot Corp Rep Ext$bootstrapped_weights

```

		Original Est.	Bootstrap Mean	Bootstrap SD	T Stat.	2.5% CI	97.5% CI
qual_1	-> QUAL	0.202	0.207	0.061	3.338	0.093	0.321
qual_2	-> QUAL	0.041	0.040	0.051	0.808	-0.063	0.135
qual_3	-> QUAL	0.106	0.104	0.065	1.629	-0.027	0.221
qual_4	-> QUAL	-0.005	-0.004	0.054	-0.085	-0.103	0.103
qual_5	-> QUAL	0.160	0.160	0.059	2.714	0.044	0.268
qual_6	-> QUAL	0.398	0.394	0.064	6.224	0.259	0.512
qual_7	-> QUAL	0.229	0.224	0.057	4.006	0.109	0.333
qual_8	-> QUAL	0.190	0.190	0.061	3.099	0.070	0.312
perf_1	-> PERF	0.468	0.465	0.068	6.887	0.326	0.594
perf_2	-> PERF	0.177	0.180	0.068	2.584	0.050	0.314
perf_3	-> PERF	0.194	0.189	0.054	3.603	0.089	0.304
perf_4	-> PERF	0.340	0.339	0.072	4.746	0.201	0.485
perf_5	-> PERF	0.199	0.197	0.062	3.184	0.075	0.323
csor_1	-> CSOR	0.306	0.300	0.083	3.671	0.139	0.467
csor_2	-> CSOR	0.037	0.035	0.069	0.536	-0.097	0.173
csor_3	-> CSOR	0.406	0.406	0.083	4.863	0.241	0.563
csor_4	-> CSOR	0.080	0.079	0.076	1.058	-0.076	0.220
csor_5	-> CSOR	0.416	0.416	0.089	4.662	0.245	0.607
attr_1	-> ATTR	0.414	0.415	0.071	5.848	0.273	0.544
attr_2	-> ATTR	0.201	0.196	0.063	3.165	0.074	0.322
attr_3	-> ATTR	0.658	0.655	0.062	10.549	0.537	0.770
comp_1	-> COMP	0.469	0.468	0.021	22.413	0.429	0.512
comp_2	-> COMP	0.365	0.366	0.017	21.421	0.335	0.400
comp_3	-> COMP	0.372	0.373	0.014	26.068	0.346	0.401
like_1	-> LIKE	0.419	0.420	0.014	29.343	0.393	0.448
like_2	-> LIKE	0.374	0.374	0.013	28.576	0.351	0.401
like_3	-> LIKE	0.363	0.363	0.014	26.477	0.337	0.390
cusa	-> CUSA	1.000	1.000	0.000	.	1.000	1.000
cusl_1	-> CUSL	0.369	0.369	0.016	23.494	0.338	0.401
cusl_2	-> CUSL	0.420	0.421	0.015	28.972	0.395	0.452
cusl_3	-> CUSL	0.365	0.365	0.015	24.427	0.335	0.393

■ Fig. 5.7 Bootstrapped indicator weights. (Source: authors' screenshot from R)

α , if the corresponding $(1 - \alpha)\%$ bootstrap confidence interval does *not* include zero. In other words, if a confidence interval for an estimated coefficient, such as an indicator weight w_i , does not include zero, the hypothesis that w_i equals zero is rejected, and we assume a significant effect.

Looking at the significance levels, we find that all formative indicators are significant at a 5% level, except *csor_2*, *csor_4*, *qual_2*, *qual_3*, and *qual_4*. For these indicators, the 95% confidence intervals include the value zero. For example, for *csor_2*, our analysis produced a lower boundary of **-0.097** and an upper boundary of **0.173**. Similarly, these indicators' *t*-values are clearly lower than 1.960, providing support for their lack of statistical significance.

To assess these indicators' absolute importance, we examine the indicator loadings by running `sum Boot Corp Rep Ext$bootstrapped_loadings`. The output in ■ Fig. 5.8 (column: **Original Est.**) shows that the lowest indicator loading of these five formative indicators occurs for *qual_2* (**0.570**). Furthermore,

5

```

~/seminr/
> # Inspect the bootstrapping results for the outer loadings
> sum_boot_corp_rep_ext$bootstrapped_loadings
      Original Est. Bootstrap Mean Bootstrap SD T Stat. 2.5% CI 97.5% CI
qual_1 -> QUAL      0.741      0.738      0.045 16.619  0.640  0.819
qual_2 -> QUAL      0.570      0.568      0.054 10.636  0.454  0.668
qual_3 -> QUAL      0.749      0.744      0.039 19.281  0.663  0.815
qual_4 -> QUAL      0.664      0.658      0.045 14.606  0.567  0.738
qual_5 -> QUAL      0.787      0.780      0.034 23.222  0.711  0.842
qual_6 -> QUAL      0.856      0.848      0.031 27.547  0.781  0.901
qual_7 -> QUAL      0.722      0.713      0.042 17.090  0.626  0.790
qual_8 -> QUAL      0.627      0.622      0.049 12.706  0.520  0.711
perf_1 -> PERF      0.846      0.839      0.035 24.055  0.766  0.902
perf_2 -> PERF      0.690      0.686      0.047 14.665  0.587  0.775
perf_3 -> PERF      0.573      0.568      0.051 11.156  0.466  0.664
perf_4 -> PERF      0.717      0.715      0.050 14.249  0.614  0.815
perf_5 -> PERF      0.638      0.634      0.059 10.760  0.507  0.739
csor_1 -> CSOR      0.771      0.761      0.050 15.397  0.652  0.852
csor_2 -> CSOR      0.571      0.562      0.060  9.432  0.437  0.671
csor_3 -> CSOR      0.838      0.830      0.043 19.467  0.737  0.904
csor_4 -> CSOR      0.617      0.610      0.057 10.745  0.492  0.716
csor_5 -> CSOR      0.848      0.841      0.043 19.726  0.749  0.917
attr_1 -> ATTR      0.754      0.753      0.051 14.919  0.645  0.841
attr_2 -> ATTR      0.506      0.501      0.066  7.609  0.362  0.626
attr_3 -> ATTR      0.891      0.887      0.033 26.678  0.816  0.946
comp_1 -> COMP      0.824      0.822      0.021 39.717  0.778  0.858
comp_2 -> COMP      0.821      0.821      0.020 40.457  0.781  0.856
comp_3 -> COMP      0.844      0.843      0.019 43.658  0.804  0.878
like_1 -> LIKE      0.880      0.880      0.016 55.828  0.847  0.907
like_2 -> LIKE      0.869      0.867      0.017 50.469  0.832  0.899
like_3 -> LIKE      0.844      0.844      0.019 44.828  0.803  0.879
cusa -> CUSA      1.000      1.000      0.000  .      1.000  1.000
cusl_1 -> CUSL      0.833      0.832      0.024 35.285  0.780  0.874
cusl_2 -> CUSL      0.917      0.917      0.010 89.030  0.894  0.935
cusl_3 -> CUSL      0.843      0.842      0.023 37.177  0.793  0.881
>

```

■ Fig. 5.8 Bootstrapped indicator loadings. (Source: authors' screenshot from R)

results from bootstrapping show that the t -values of the five indicator loadings (i.e., *csor_2*, *csor_4*, *qual_2*, *qual_3*, and *qual_4*) are clearly above 2.576, suggesting that all indicator loadings are significant at a level of 1% (■ Fig. 5.8). Moreover, prior research and theory also provide support for the relevance of these indicators for capturing the corporate social responsibility and quality dimensions of corporate reputation (Eberl, 2010; Sarstedt, Wilczynski, & Melewar, 2013; Schwaiger, 2004; Schwaiger, Sarstedt, & Taylor, 2010). Thus, we retain all indicators in the formatively measured constructs, even though not every indicator weight is significant.

The analysis of indicator weights concludes the evaluation of the formative measurement models. Considering the results from ► Chaps. 4 and 5 jointly, all reflective and formative constructs exhibit satisfactory levels of measurement quality. Thus, we can now proceed with the evaluation of the structural model (► Chap. 6).

Summary

The evaluation of formative measurement models starts with convergent validity to ensure that the entire domain of the construct and all of its relevant facets have been covered by the indicators. In the next step, researchers assess whether pronounced levels of collinearity among indicators exist, which would inflate standard errors and potentially lead to sign changes in the indicator weights. The final step involves examining each indicator's relative contribution to forming the construct. Hence, the significance and relevance of the indicator weights must be assessed. It is valuable to also report the bootstrap confidence interval that provides additional information on the stability of the coefficient estimates. Nonsignificant indicator weights should not automatically be interpreted as indicating poor measurement model quality. Rather, researchers should also consider a formative indicator's absolute contribution to its construct (i.e., its loading). Only if both indicator weights and loadings are low or even nonsignificant should researchers consider deleting a formative indicator.

? Exercise

We continue with the analysis of the influencer model as introduced in ► Chaps. 3 and 4. The dataset is called **influencer_data** and consists of 222 observations of 28 variables. ► Figure 3.10 illustrates the PLS path model; ► Tables 3.9 and 3.10 describe the indicators. Note that the indicator *sic_global* serves as global single item in the redundancy analysis of the *SIC* construct.

Load the influencer data, reproduce the influencer model in the SEMinR syntax, and estimate the PLS path model. As we have already assessed the reliability and validity of the reflective measures, we focus on the analysis of the *SIC* construct as follows:

1. Does the *SIC* construct display convergent validity?
2. Do the construct indicators suffer from collinearity issues?
3. Are all indicator weights statistically significant and relevant?
4. If not, based on the indicator loadings and their significance, would you consider deleting one or more of the indicators?

References

- Aguirre-Urreta, M. I., & Rönkkö, M. (2018). Statistical inference with PLS using bootstrap confidence intervals. *MIS Quarterly*, 42(3), 1001–1020.
- Becker, J.-M., Ringle, C. M., Sarstedt, M., & Völckner, F. (2015). How collinearity affects mixture regression results. *Marketing Letters*, 26(4), 643–659.
- Bollen, K. A., & Diamantopoulos, A. (2017). In defense of causal-formative indicators: A minority report. *Psychological Methods*, 22(3), 581–596.
- Cenfetelli, R. T., & Bassellier, G. (2009). Interpretation of formative measurement in information systems research. *MIS Quarterly*, 33(4), 689–708.
- Cheah, J. H., Sarstedt, M., Ringle, C. M., Ramayah, T., & Ting, H. (2018). Convergent validity assessment of formatively measured constructs in PLS-SEM. *International Journal of Contemporary Hospitality Management*, 30(11), 3192–3210.
- Chin, W. W. (1998). The partial least squares approach to structural equation modeling. *Modern Methods for Business Research*, 295(2), 295–336.

- Cohen, J. (1992). A power primer. *Psychological Bulletin*, 112(1), 155–159.
- Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap Methods and Their Application*. Cambridge, MA: Cambridge University Press.
- Diamantopoulos, A., Sarstedt, M., Fuchs, C., Wilczynski, P., & Kaiser, S. (2012). Guidelines for choosing between multi-item and single-item scales for construct measurement: A predictive validity perspective. *Journal of the Academy of Marketing Science*, 40(3), 434–449.
- Eberl, M. (2010). An application of PLS in multi-group analysis: The need for differentiated corporate-level marketing in the mobile communications industry. In V. Esposito Vinzi, W. W. Chin, J. Henseler, & H. Wang (Eds.), *Handbook of Partial Least Squares: Concepts, Methods and Applications*. (Springer handbooks of computational statistics series, vol. II) (pp. 487–514). Berlin: Springer.
- Faul, F., Erdfelder, E., Buchner, A., & Lang, A.-G. (2009). Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses. *Behavior Research Methods*, 41(4), 1149–1160.
- Grimm, M. S., & Wagner, R. (2020). The impact of missing values on PLS, ML and FIML model fit. *Archives of Data Science, Series A*, 6(1), 04.
- Hair, J. F., Hult, T., Ringle, C. M., & Sarstedt, M. (2022). *A primer on partial least squares structural equation modeling (PLS-SEM)* (3rd ed.). Thousand Oaks, CA: Sage.
- Hair, J. F., Risher, J. J., Sarstedt, M., & Ringle, C. M. (2019). When to use and how to report the results of PLS-SEM. *European Business Review*, 31(1), 2–24.
- Hair, J. F., Sarstedt, M., Hopkins, L., & Kuppelwieser, V. G. (2014). Partial least squares structural equation modeling (PLS-SEM): An emerging tool in business research. *European Business Review*, 26(2), 106–121.
- Henseler, J., Ringle, C. M., & Sinkovics, R. R. (2009). The use of partial least squares path modeling in international marketing. *Advances in International Marketing*, 20, 277–320.
- Houston, M. B. (2004). Assessing the validity of secondary data proxies for marketing constructs. *Journal of Business Research*, 57(2), 154–161.
- Kock, N., & Hadaya, P. (2018). Minimum sample size estimation in PLS-SEM: The inverse square root and gamma-exponential methods. *Information Systems Journal*, 28(1), 227–261.
- Mason, C. H., & Perreault, W. D. (1991). Collinearity, power, and interpretation of multiple regression analysis. *Journal of Marketing Research*, 28(3), 268–280.
- Sarstedt, M., Diamantopoulos, A., Salzberger, T., & Baumgartner, P. (2016). Selecting single items to measure doubly-concrete constructs: A cautionary tale. *Journal of Business Research*, 69(8), 3159–3167.
- Sarstedt, M., Wilczynski, P., & Melewar, T. (2013). Measuring reputation in global markets: A comparison of reputation measures' convergent and criterion validities. *Journal of World Business*, 48(3), 329–339.
- Schwaiger, M. (2004). Components and parameters of corporate reputation: An empirical study. *Schmalenbach Business Review*, 56(1), 46–71.
- Schwaiger, M., Sarstedt, M., & Taylor, C. R. (2010). Art for the sake of the corporation: Audi, BMW Group, DaimlerChrysler, Montblanc, Siemens, and Volkswagen help explore the effect of sponsorship on corporate reputations. *Journal of Advertising Research*, 50(1), 77–90.
- Streukens, S., & Leroi-Werelds, S. (2016). Bootstrapping and PLS-SEM: A step-by-step guide to get more out of your bootstrap results. *European Management Journal*, 34(6), 618–632.

Suggested Reading

- Cenfetelli, R. T., & Bassellier, G. (2009). Interpretation of formative measurement in information systems research. *MIS Quarterly*, 33(4), 689–708.
- Hair, J. F., Hult, T., Ringle, C. M., & Sarstedt, M. (2022). *A primer on partial least squares structural equation modeling (PLS-SEM)* (3rd ed.). Thousand Oaks, CA: Sage.
- Sarstedt, M., Hair, J. F., Ringle, C. M., Thiele, K. O., & Gudergan, S. P. (2016). Estimation issues with PLS and CBSEM: Where the bias lies! *Journal of Business Research*, 69(10), 3998–4010.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

