Lotus Manual

Reliability and Accuracy with SPSS

Version 1.0 2015

BENJAMIN FRETWURST

1. Introduction

This manual is intended to introduce the reliability coefficient called Lotus which is constructed simply and easily interpretable.[1] The coefficient is based on the agreement with the most commonly coded value (MCCV) and is determined for two or more coders[2] within each separate coding unit. This document will introduce a SPSS custom dialog package that can be used to calculate Lotus in an unstandardized and standardized form at any level of measurement.[3] In addition, it analyzes several common obstacles occurring in content analysis:

- Lotus can be applied to *categorical*, *ordinal or metrical* scales.
- The calculation of Lotus is easier to understand than the calculation of Krippendorff's alpha.[4]
- The quality of the codebook can be differentiated from the quality of the coder because reliability values can be determined for each coder.
- In contrast to reliability coefficients based on pairwise comparison, incorrect values will not be positively factored into reliability.
- Accuracy can be displayed as a comparison with a *gold standard* and is uniform to the intercoder coefficient Lotus.
- For hierarchical variables, it is easy to display the hierarchical level of a given reliability.
- The reliability of rare phenomena can be calculated.
- Data records do not have to be restructured for Lotus. Coders' data records are simply merged with one another.

2. Reliability and Accuracy of Content Analyses

Empirical science bases its assumptions on the validity of its measurements. There is no direct access to that validity but in the learning process, scientists develop methods to systematize their measurements to the point that enables them to inter-subjectively agree when a sufficient degree of certainty is reached (CHALMERS 1999). The validity of scientific inferences must be judged by the methods they are derived from. Content analyses, just like any other empirical method, demand validity and reliability. When multiple experienced researchers agree on the *result* of a measurement, we can then refer to the "accuracy" (KRIPPENDORFF 1980) as the *validity* of the tool chosen for measurement. Thus, accuracy can be assessed in a simple way. The requirement of *reliability* considers the agreement among coders regarding the

[1] LOMBARD et al (2002) find that "although a handful of tools are available to implement the sometimes complex formulae required, information about them is often difficult to find and they are often difficult to use. It therefore seems likely that many studies fail to adequately establish and report this critical component of the content analysis method." (LOMBARD et al 2002: 588)

[2] Even if the actual coding is done by individuals rather than ensembles, a reliability check with a second or more coders is needed (EVANS 1996).

[3] Krippendorff's alpha can easily be displayed, too, because the underlying SPSS macro code automatically restructures the necessary data records as needed.

[4] „This [Krippendorff's alpha] is a highly attractive coefficient but has rarely been used because of the tedium of its calculation." (NEUENDORF 2002: 151)

measurement itself. If both reliability and accuracy can be defined as an agreement between subjects, in principle, their verifiability should also be the same.

2.1 Reliability

Reliability expresses the similarity of measurements (coded values) across repeated measurements of the same material (NEUENDORF 2002*). Intercoder reliability* is defined as the amount of accordance among two or more coders. KRIPPENDORFF emphasized: “agreement is what we measure; reliability is what we wish to infer from it” (KRIPPENDORFF 2004: 5). Although reliability is a methodological standard construct, it is not always entirely clear what it measures or what it should be based on. NEUENDORF 2002 (p. 145) names:

- the quality of coding scheme as the measurement tool,
- the quality of the coder,
- the quality of the coder training,
- characteristics of the test material.

At various points during the research process, a reliability coefficient takes on different functions and describes gradually distinct aspects of the trustworthiness of the measurement. It should:

- control for coder training,
- indicate how strongly the coding is determined by the code book,
- measure reproducibility,
- describe a study's data quality.

The reliability of the coders, the code book, and the coders' training are factored into the measurement error variance. For reasons having to do with practical aspects of content analysis, a file that uses the predefined coding units as test materials is set up. For this test material file, the reliability coefficient provides valid information about the variance of measurement error within the data. Whether or not coders agree as to the values assigned to a variable cannot be ascribed to the quality of the measurement tool with any differentiation. In order to be able to determine the measurement error of the tool, the proportion of the coders' measurement errors must be subtracted. The proportion of the coders' measurement errors corresponds to the intra coder errors and can be accessed by the standard deviation of the coder reliabilities.

reliability of a variable = code book reliability + stdv(coder reliabilities)

Like the errors of individual coders, the influence of coder training can also produce measurement errors. One can try to identify the influence of coder training by experimenting with before/after measurements, but it hardly makes sense to measure reliability before training, given that, without training, coders cannot use the tool or can only use it erroneously. The significantly greater coder error rate prior to training would therefore be incorporated into the difference between before and after. So the extent to which coder training affects reliability cannot be clearly determined even by experimentation. The fact that decisions and coder instructions are made transparent in the codebook is thus a matter of globally inter-subjective verifiability and therefore of transparency.

2.2 Accuracy

One part of inter-subjective validity is accuracy. This can be tested within the pretest in which a “gold standard” is created by the research director. In this particular form, validity is measured as *expert validity*. One good way is forming a team made up of the research director and the best coders after the reliability coding has been completed to jointly decide on the *gold standard* for each coding unit, i.e. what value best represents each case of coding. The material of the reliability test should be coded: by (1) the researcher director, (2) the team of the research director and the experienced coders (after the reliability coding). KRIPPENDORFF (1980) opts for the term “accuracy” as an intermediate form between reliability and validity.

2.3 Entropy

Information theory defines information entropy as a measure of the reduction of uncertainty (SHANNON 1948 and PIERCE 1980). Applied to content-analysis coding, the number of codable categories is uncertainty and the coder decision is reduction of uncertainty. The informational content of a variable depends on the possible categories. The important thing is to consider the amount of information contained in a variable. If a variable has few categories, it will be easier to reliably code it than a variable with many categories would be. This consideration is technically and mathematically expressed in all standardization algorithms.

3. The LOTUS Coefficient

The reliability of content-analysis measurement is understood as inter-coder reliability. It is defined as the percentage of agreement of all coders with (one of) the most common coded value in each coding unit.

MERTEN (1983) distinguishes between three types of inter-coder reliability:

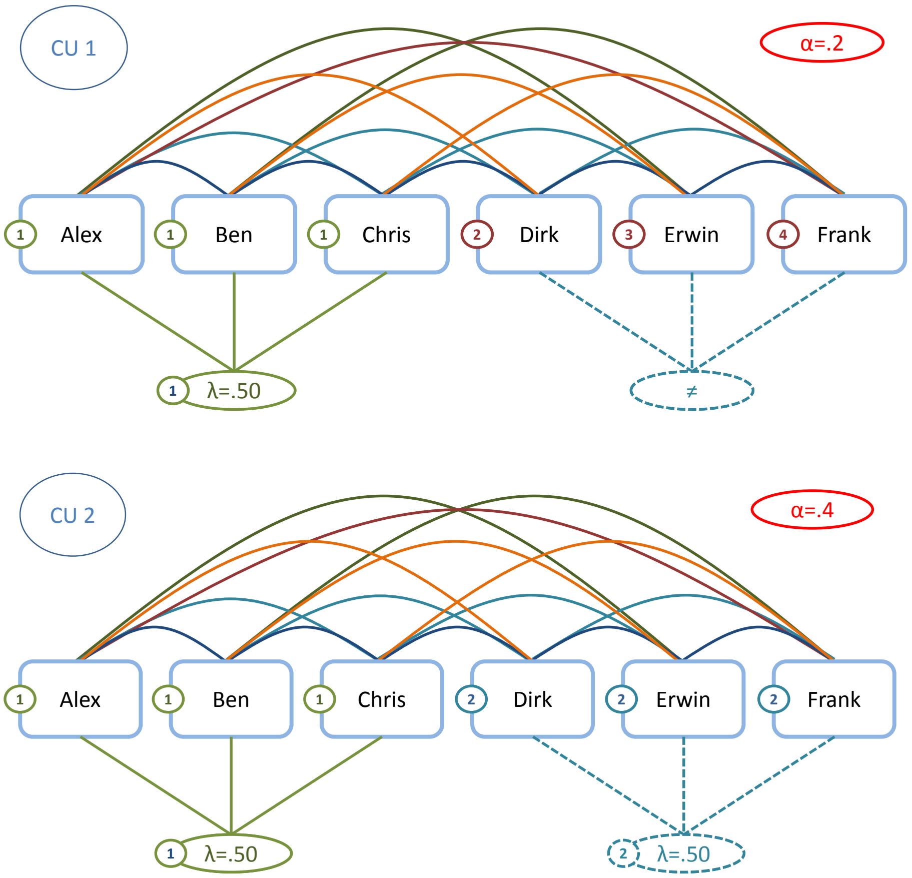
> *Type 1:* The degree of the agreement between every pair of coders among *h* coders is examined (intersection of every two coders out of *h* coders)
>
> *Type 2:* The degree of the overall agreement among *h* coders (intersection of h coders)
>
> *Type 3:* The behavior of the majority of the coders is examined. In the simplest case this is two out of three, thus examining the extent to which two coders out of three (or, more generally: *m* of *h* coders) are in agreement.
>
> (MERTEN 1983: 302p)

The definition and mode of calculation proposed here deviate from the information on pair-wise comparisons (type 2 as according to, for example, HOLSTI (1969) and its derivatives, such as Scott's *pi*, Cohen's *kappa*, or Krippendorff's *alpha*) and absolute or majority agreement (type 3). Lotus constitutes another type: It defines the proportion of agreement among all coders without using each individual comparison as a basis for calculation. Instead, it uses agreement with a reference value – in this case of Lotus the value that is 1. most common coded value (MCCV)[5] per coding unit and 2. the gold standard.

[5] That is in principle the same as the mode per coding unit. To avoid confusion with the mode of a variable, I use MCCV for the most common coded value per coding unit.

The MCCV represents the value implied by the instrument[6] and should be identified as a most common agreement.[7] Coefficients based on pair-wise comparisons inherently include a weakness: The identical coding of incorrect values will be positively factored into the overall reliability. Using the MCCV as a reference value will avoid that problem. Therefore, Lotus only considers this. If coders have jointly coded other values, Lotus will not consider them.

Figure 1 demonstrates the calculation of Lotus in comparison with coefficients that are based on pair-by-pair comparisons. In the example, six coders code every coding unit. The upper connecting curves represent pair-by-pair comparisons. Using the first coding unit (CU1) as an example, three coders consistently coded the same value, 1. The other three each chose other and different values. Lotus (represented as lambda), measures an agreement of .5 (or 50%). In this constellation, Krippendorff's *alpha* yields a value of .2. In the second example (CU2), not only have the first three coders coded the same value, but the other three have coded the same too – yet both groups of coders have chosen a different value each. The Lotus coefficient remains unchanged, indicating that it makes no difference whether value 1 or 2 is regarded as the MCA. By contrast, Krippendorff's *alpha* has doubled because each agreement among coders is factored into the characteristic value.



[6] Measurements that are performed uniformly but incorrectly by all coders will be regarded as invalid, yet reliable. The same applies to all reliability coefficients that are based on pair-wise comparisons because they only consider agreements and hence reliability is still perfect even when all coders uniformly choose the same false category.

[7] It may occur that two categories have been coded with equal frequency. In that case, it does not matter which of those two characteristics is considered the principal agreement.

Figure 1: Lotus in comparison with Krippendorff's *alpha*

3.1 Lotus for Variables of Different Scales of Measurement

Comparisons between nominal variables are simple because the coded characteristics are either equal or not; there is no such relationship as ‘more’ or ‘less’ between them. Comparisons appear to be more difficult when not only identical but also similar codes are to be regarded as reliable. When applied to ordinal and metrical variables, Krippendorff's *alpha* includes distance measurement, against which reliability is relativized (see KRIPPENDORFF 2004). The Lotus coefficient proposed here makes comparisons against a given *tolerance range*. The example in Figure 2 shows the idea for continuous variables. As a measure of the highest agreement, the mathematical mean for continuous variables is calculated and all codes within the tolerance range above and below the mean are regarded as in agreement. Consequently, the maximum range of tolerable deviation must be predefined. If, for example, the length of a news report is recorded in seconds, a deviation of three seconds may still be considered tolerable while the same deviation for a ”sound bite” within that particular news report may already be too imprecise. Reliability values for metric variables must therefore always be identified within their range of tolerance. Consequently, the coefficient proposed here is also clearly more transparent than the Krippendorff’s squared distance arithmetic for continuous variables.

Not all metrical variables are continuous. Ordinal and metrical variables that have only few possible values are used more often for content analyses than continuous variables. Discrete codes with few values should always be treated as categorical variables even when the phenomena to be measured have a continuous nature.[8]

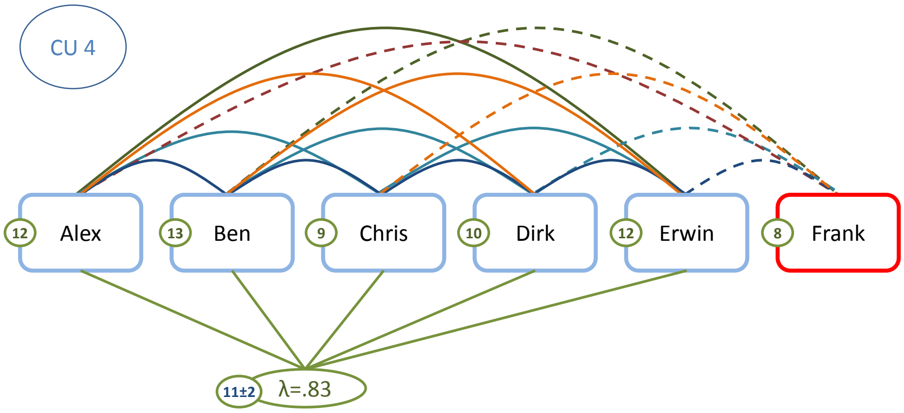


Figure 2: Lotus with tolerance range

[8] The degree of change in Lotus can be tested in the context of the coder training with the help of tolerance 1 if coder decisions are at categorical borders. In publications, however, the reliability coefficients that emerge for the number of values with which the analyses were carried out must always be given. That way, if values that are measured with more nuances are collected for the presentation of results, the improved reliabilities of the reduced characteristics can be documented.

3.2 Standardized Lotus

The simple Lotus coefficient can be interpreted intuitively. However, depending on which procedure of the content analysis is to be examined, it does allow for too many influences. If the quality of the coding instructions is the only element under scrutiny then the number of possible categories of a variable is not a factor. ROGOT and GOLDBERG (1966) emphasize the importance of contrasting observed with expected inter-coder agreements. *S-Lotus* is the Lotus coefficient that is correlated to these random agreements. If the Lotus coefficient is equal to two variables, then the variable with more categories should result in a higher S-Lotus.

The starting point for the calculation is the simple Lotus coefficient, which is reduced by the expected agreements respectively the inverse value of the number of categories (*K*).[9] Given that Lotus has a maximum of 1, if there is perfect agreement, S-Lotus should also be 1. S-Lotus is standardized to 1 (see formula), so it can take on a value of 1 if perfect agreement occurs. S-Lotus therefore gives the ratio of coding that is in agreement with all possible agreements that are not expected to happen by coincidence.

$$\text{S– Lotus} = \frac{Lotus - 1/K}{1 - 1/K}$$

For two coders and dichotomous variables, S-Lotus is similar to Cohens κ (kappa) (COHEN 1960; GWET 2001) and Scott's π (pi) (SCOTT 1955).[10] In the same way, Krippendorff's *alpha* is relativized to the anticipated likelihood of coding that is correct by chance. In this respect, S-Lotus and Krippendorff's *alpha* have the same measurement target.

4. Accuracy with Lotus

The percentage of agreement of all coders with a target that is considered to be sufficiently valid – the gold standard – can be used as an indicator of accuracy. Coding is compared with the gold standard instead of the MCCV. The Lotus for the gold standard (Lotus-GS) is thus the percentage of agreement with the gold standard. As the basis for their calculation is identical, Lotus and Lotus-GS are directly comparable.[11]

The maximum proportion of agreement with the gold standard is equal to inter-coder reliability. If the gold standard always indicates the same value as the majority of the coders, then Lotus and Lotus-GS will be the same. If the gold standard indicates a value different from the majority, then Lotus will always be higher than Lotus-GS.[12] This mathematically reflects the logical relationship between reliability and validity: Reliability is the *necessary* but not

[9] For continuous variables, the maximum *K* is equal to *N* when each case has a different characteristic. Consequently, *K* is also finite where continuous variables are concerned.

[10] POTTER & LEVINE-DONNERSTEIN (1999) discuss the problem of pi, calculating P_e as number of times each value on a coding variable is chosen. P_e in the S-Lotus formula is a priori calculated as inverse of the number of variable attributes.

[11] If the two coefficients for a variable differ considerably from each other, this is an indicator that coding instructions were focused too strongly on reliability to the detriment of validity.

[12] Inter-coder reliability can only fall below agreement with the gold standard in the unlikely event that each coder has coded a variable for a code unit differently but one coder is in agreement with the gold standard.

sufficient precondition for validity. An unreliable measurement cannot be valid. On the other hand, reliably invalid measurements are possible.

5. Monte Carlo Simulation

In this section, the characteristics of the Lotus coefficient are displayed based on a Monte Carlo simulation and compared with Krippendorff's *alpha*. For this purpose, data with the following prescribed characteristics were compiled (simulated):

- its number of categories,
- the likelihood that coders will produce coding that is in agreement,
- the likelihood of its agreement with a prescribed gold standard.

In addition to the prescribed agreements, there are also chance agreements. The study simulated how strongly coders' actions are determined by codebook guidelines and training and to what extent additional chance agreements occur. If, for example, agreement for a variable with two characteristics is supposed to be 50%, there is still a possibility of 25% random agreement, as there is still an even likelihood that the values of the remaining 50% will also be in agreement.

There were 1,000 code units with 20 coders simulated. Above all, the scale of the coding units is larger than in conventional reliability tests. But because Monte Carlo simulations deal with random processes, the character of the coefficients can be more clearly recognized when large random samples are simulated.

5.1 Simulation and Results for Lotus

The results of various specifications were entered into Table 1. Inter-coder reliability (ICR) is the preset likelihood of agreement. The second column shows the number of categories (Cat) of each simulated variable, its inverse value defining the impact of chance in this random experiment.

The last column of the guideline shows anticipated agreement.[13] For each of the variables generated according to these standards, the Lotus coefficient (Lotus) and the standardized Lotus (S-Lotus) were calculated using SPSS macros.

On average, the simple Lotus coefficient equals the agreement that is to be expected according to the exact specifications and chance (agreement column). The average deviation is virtually 0. Lotus therefore assesses the simulated agreement without bias. S-Lotus reflects the proportion of prescribed agreement and should correspond to the ICR column. That way, the simulated influence of chance is subtracted from individual coder decisions. S-Lotus is also unbiased and dispersed equally among the values prescribed by the ICR.

Krippendorff's alpha is very small if required agreement is low. With a specification of ICR = .30, a value of .15 would have to been expected, which is still larger than *alpha*. For ICR = .50, *alpha* is equal to Lotus reduced by the likelihood in this simulation, which is to say it equals .25. The larger the prescribed agreement, the more *alpha* will exceed the simple adjustment of agreement to the anticipated chance hits. For low actual agreement, Krippendorff's alpha is too strict and underestimates substantial reliability. When there are many coders, S-Lotus reflects the substantial proportion of the reliability that is induced by the coding instructions.

[13] The anticipated value arises from the guidelines for inter-coder reliability and the chance process for the non-prescribed portion of the simulated coding. Anticipation was calculated as
Anticipated = ICR + Chance · (1 – ICR).

Lotus and Krippendorff's *alpha* are reliability coefficients that should be neutral with respect to the number of coders. The last three columns of the table compare the differences between the simulation with 20 coders and a simulation with four coders. For lower prescribed agreement and few categories, Lotus and *alpha* for few coders are slightly above their values for 20 coders. If chance agreements are minimal, as in the simulation with 100 categories, then the Lotus value for few coders is lower than for many coders. In contrast, *alpha* systematically shows higher values with few coders. For agreement within the intended range of 70 to 100%, *alpha* shows systematically lower values for few coders than for 20 coders. Within that range, Lotus is neutral with respect to the number of coders.

Table 1: Monte Carlo simulation and calculation of *Lotus*, *S-Lotus*, and Krippendorff's *alpha*

Variable Construction				Coefficients			Differences 20 vs. 4 Coders		
ICR	Cate-gories	Chance	Agreement	Lotus	S-Lotus	Alpha	Lotus	S-Lotus	α
.30	2	.50	.65	.65	.30	.09	-.01	-.02	-.05
.50	2	.50	.75	.75	.50	.25	-.03	-.06	.01
.70	2	.50	.85	.85	.69	.47	-.01	-.01	.05
.90	2	.50	.95	.95	.90	.77	.00	.00	.11
.99	2	.50	1.00	.99	.99	.94	.00	.00	.13
.30	5	.20	.44	.44	.30	.09	-.07	-.09	-.01
.50	5	.20	.60	.60	.50	.25	-.02	-.03	.01
.70	5	.20	.76	.76	.70	.47	-.01	-.01	.03
.90	5	.20	.92	.92	.90	.77	-.01	-.01	.12
.99	5	.20	.99	.99	.99	.94	.00	.00	.16
.30	100	.01	.31	.30	.30	.10	.06	.06	-.06
.50	100	.01	.51	.51	.50	.27	.04	.04	-.06
.70	100	.01	.70	.70	.70	.50	.00	.00	-.06
.90	100	.01	.90	.90	.90	.81	.01	.01	.01
.99	100	.01	.99	.99	.99	.97	.00	.00	.02

5.2 Simulation and Results for the Lotus Gold Standard

Agreement with the gold standard (LGS) and the standardized version S-LGS were simulated according to the principle described above (see Table 2). The maximum possible agreement with the gold standard (LGS) equals to agreement with the MCCV (Lotus). The coding that did not agree with the target (GS) could have been one of the possible categories by chance. The number of preset categories revealed the extent to which randomly correct coding was to be expected. From these settings, the expected agreement with the simulated gold standard can be derived.

The anticipated agreement with the gold standard corresponds to the specific value of the LGS. The deviations are somewhat greater than with simple Lotus, but on average they amount to 0. The LGS therefore estimates the proportion of agreement with a predetermined gold standard without bias. The standardized variant of the LGS (S-LGS) subtracts chance from the agreements. The S-LGS likewise estimates the gold standard (GS) preset in the Monte Carlo simulation without bias.

Table 2: Monte Carlo simulation for LGS and S-LGS

Targets per Variable				Monte Carlo	
GS	Cat	Chance	Anticipation	LGS	S-LGS
.25	2	.50	.63	.62	.25
.35	2	.50	.68	.68	.37
.45	2	.50	.73	.72	.43
.49	2	.50	.75	.74	.49
.63	2	.50	.82	.81	.61
.81	2	.50	.91	.91	.82
.25	4	.25	.44	.42	.23
.35	4	.25	.51	.53	.38
.45	4	.25	.59	.58	.44
.49	4	.25	.62	.62	.49
.63	4	.25	.72	.72	.63
.81	4	.25	.86	.84	.79
.25	6	.17	.38	.36	.23
.35	6	.17	.46	.46	.36
.45	6	.17	.54	.55	.46
.49	6	.17	.58	.58	.50
.63	6	.17	.69	.70	.64
.81	6	.17	.84	.83	.80

Cat: Number of categories; GS: Agreement with a gold standard; Chance: Inverse value of the number of categories (Cat); Anticipation: Proportion of agreements to be expected

6. Implementation in SPSS and Usage of the Lotus dialog

The Lotus dialog is implemented in the SPSS custom dialog and is presented in section 6.2. Lotus simplifies data handling in at least two ways: (1) No additional installation of programs like R or Python and (2) no restructuring of the dataset.

6.1 Demands on the Data Structure

Construction procedure of the reliability dataset: During the test coding, each coder and the research director (gold standard) enter the data the same way into the same SPSS data editor. The individual files are simply added and the original variable names can be maintained. Each coding unit (CU) and coder is embodied in a variable.[14]

[14] If missing CUs are considered, then blank lines containing only the missing CU and the coder ID must be added.

6.2 The custom dialog of Lotus

Lotus can be controlled via SPSS syntax or by the menu. The menu allows access to all functions via the dialog. After installation, Lotus can be found via the Analyze menu, sub-item Descriptive Statistics.

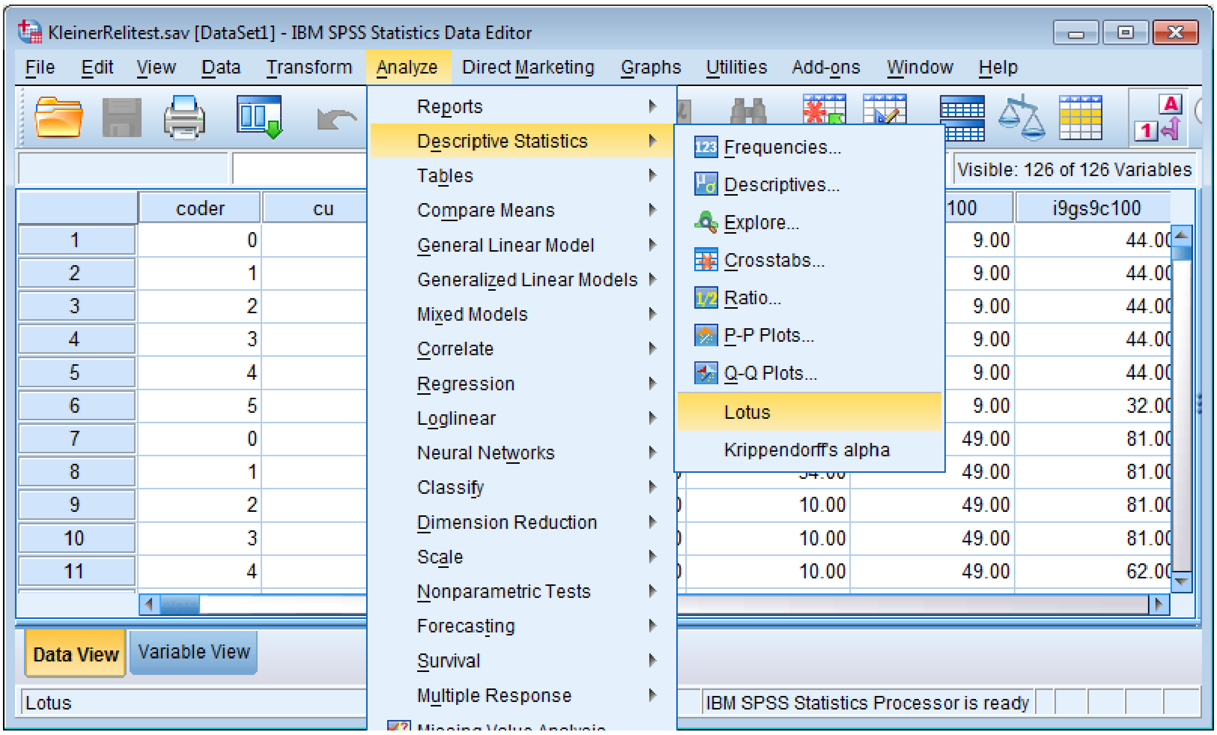
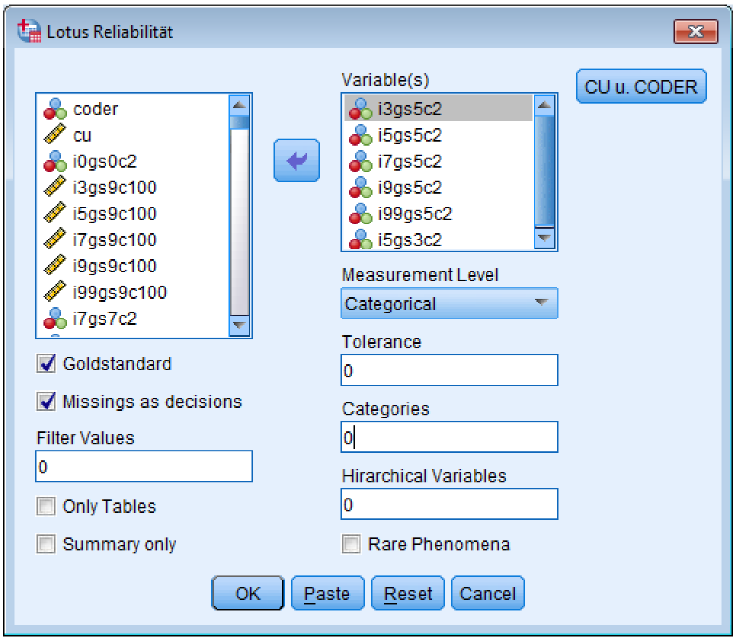


Figure 1: Lotus in the menu

You will see only numeric variables in the list of variables from which to choose. Therefore, string variables need to be recoded to numerical variables through automatic recoding. It is important to note that variables with different characteristics need to be treated separately. All variables from different runs are collected in the data file and can subsequently be represented in a single table (with “Only Tables”).



coder
cu
i0gs0c2
i3gs9c100
i5gs9c100
i7gs9c100
i9gs9c100
i99gs9c100
i7gs7c2

Variable(s)
i3gs5c2
i5gs5c2
i7gs5c2
i9gs5c2
i99gs5c2
i5gs3c2

☑ Goldstandard

☑ Missings as decisions

Filter Values: 0

☐ Only Tables

☐ Summary only

Measurement Level: Categorical

Tolerance: 0

Categories: 0

Hirarchical Variables: 0

☐ Rare Phenomena

Figure 2: Lotus via the custom dialog

6.3 Special Cases and Options

6.3.1 Missings as Values

Generally, missing values are supposed to be treated the same way as other categories, because a decision to omit a missing value is still a coding decision and its reliability must be examined, too. If only the reliability of coding with valid values is considered, the check mark next to "Missings as Values" can be removed. All missing values will be excluded from the analysis and only agreements among valid values will be included.

6.3.2 Ignore Filter Value

Filtered values represent a special case of missing values. Hierarchically organized variables and higher level coding may generate missing values in the resulting variables. Those missing values should not be incorporated into the definition of reliability because otherwise the reliability of the upstream variables would be considered doubled in the coefficients. For example, subject areas such as "politics", "economics", "society", etc. are coded into a higher-level variable (SUBJ). In a second step, a political fields variable (POL) is coded with attributes such as "politics: federal", "politics: state", "politics: international". Now, the latter variable POL will always include missing values if SUBJ does not have the attribute "politics". To avoid such phenomenon, the "Ignore Filter Value" option is provided. Cases treated with the prescribed filter value are not included in the comparisons. This option is independent of the use or suppression of missing values.

6.3.3 Category Comparisons or Average-Based

All characteristics are regarded as categorical variables in the default settings. The average-based setting has to be chosen, if the reliability of continuous variables should be calculated. Principal agreement will no longer be used as a reference value but the mathematical average coding for each code unit will be used instead. If the average does not equal an existing value by chance, there are no agreements for an average-based comparison. Using this option therefore requires a tolerance range to be indicated.

6.3.4 Tolerance

Tolerance ranges can be used for categorical comparisons or average-based reliability tests. For categorical variables, the amount of coding in the tolerance range is equal to the principal agreement. This usage assumes at least an ordinal scale of measurement and can be used to identify problems in deciding on category borders.

For average-based tests, the range above or below the average by the amount of the tolerance value is considered valid. When determining tolerance values, the units the variable has been measured must be considered. Whole-number coding is usually used. Yet, the tolerance range can be smaller than 1. A tolerance value of 0.5, for example, would mean that, for an average that is halfway between two characteristics (x.5), both adjacent characteristics would be considered valid. If the average for a code unit in the example is not halfway between two characteristics, only the characteristic that is closer would be counted as valid.

6.3.5 Number of Categories

The number of possible categories is critical for calculating the "Standardized Lotus". The Lotus macro includes a process that determines how many categories a variable has in the dataset. It may occur that not every possible category will be included in the test material. In that case, the standardized Lotus coefficients would be underestimated. It is therefore possible to indicate how many categories each variable (or variable set with an equal number of categories) theoretically includes. If contents are openly coded, then the number of possible categories is indeterminately high. In that case, a high number of categories may be entered or it is possible to simply look at the simple Lotus. The S-Lotus will exceed the simple Lotus in the event of larger numbers of categories.

$$\text{S– Lotus} = \frac{Lotus - 1/\infty}{1 - 1/\infty} = \frac{Lotus}{1}$$

6.3.6 Hierarchical Variables

The Lotus macro behind the custom dialog utilizes the possibility of testing hierarchical variables. Hierarchical variables consist of characteristics that are organized into higher and lower levels. The hierarchical characteristics must be coded such that the highest level is in the first position of the code, the second level in the following position, and so on. The number of positions that should be cut out of the code can be entered into the "Hierarchical Variables" field. If, for example, the reliability of a three-level variable is to be calculated for the first two levels, then the number of the position of the last level must be entered. The positions entered for the given variable are then cut off (truncated).

6.3.7 Rare Phenomena

The “Rare Phenomena” option deals with it the problem of calculating rare phenomena. The problem is that, under normal circumstances, the absence of applicable characteristics must be coded (GUPTA et al 1996). Consequently, the reliability of such variables is overestimated. With the “Rare Phenomena” option all coding units where all coders entered a 0 will be ignored. Thus the Lotus and S-Lotus will be calculated only for coding units with minimum 1 coding greater than 0.

On the other hand, it is also possible that only very rarely defined content is missing, but it is precisely that omission that is interesting for the analysis. For example, in an analysis of comments, whether or not coders reliably detected the absence of authorial information would be particularly interesting. In that case, the characteristic of the absence of information must be set to a value of 1 and reliability must be calculated with the Rare Phenomena option.

6.3.1 Tables Only

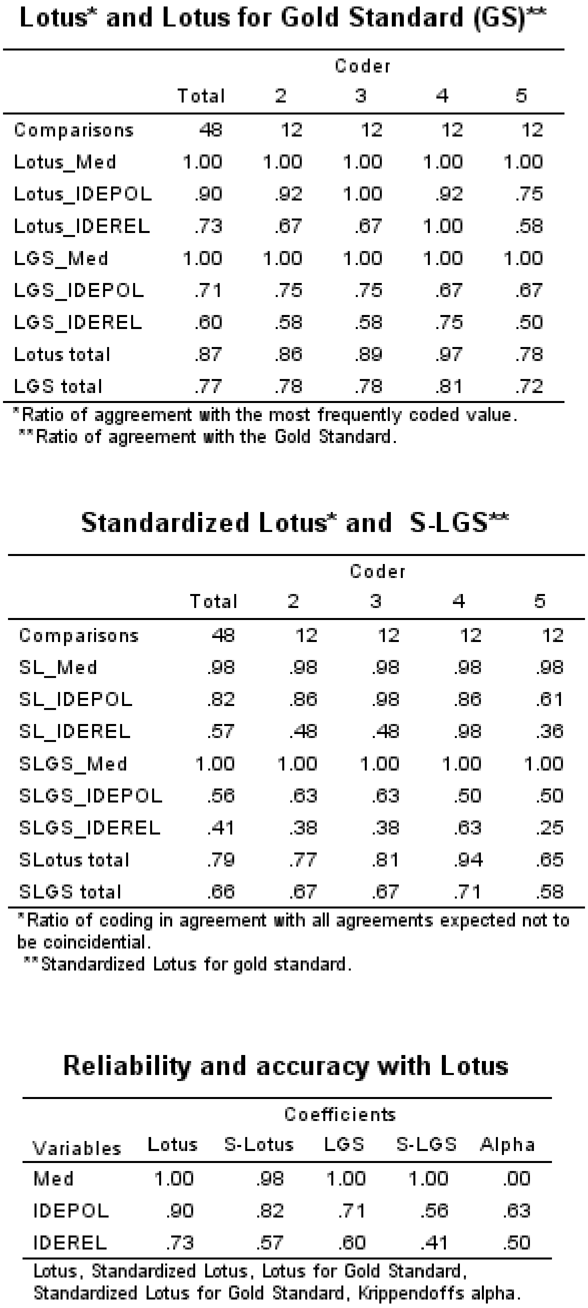
Depending on the type of variable, various analysis options can be selected to handle missing values and other requirements of variables. If multiple variables have the same characteristics, then they can be calculated together. With each run, variables for the reliability values are archived in the data file. When these reliability variables are created, then all such variables can be displayed in a single table. To do this, the “Tables Only” option must be activated, which prevents the reliability variables from being recalculated. The coefficients for different variable conditions can be displayed in one Table only that way.

6.3.2 Summary Only

By default, the custom dialog will issue separate tables for Lotus, S-Lotus, Lotus-GS, and S-Lotus-GS. The agreements of individual coders are displayed in each table with the MCCV or the gold standard. This information helps with coder training because the head trainer can see which coders deviate more significantly from the others or from the gold standard. The average of the Lotus coefficients per coder and the overall average are found in the last lines. The aggregate values beyond all of the variables are only used for comparisons with average coder reliability and must *not* be interpreted as overall coding characteristics. However, if a coder has a low share of agreement with the other coders and with the gold standard after extensive training, then that coder should be retrained or excluded from field time (for example if there is a discernible lack of motivation). The Summary Only option makes it possible to present only the summary of the reliability test.

6.4 The Lotus Dialog's Presentation of Results

The output of the Lotus coefficients begins with tables for: (1) simple Lotus and Lotus for the gold standard (LGS), and (2) standardized Lotus (SL), and standardized Lotus with the gold standard (SLGS) (example in Figure 3). The first line of each of these shows how many comparisons the calculations are based on. Then the reliability of each variable is given in the first column. The average agreement among the individual coders along with the principal agreement of the variables are entered in the subsequent columns. This information serves to identify the problems of individual coders with individual variables. The last line provides information regarding which coders achieved good or bad agreement values overall. After the four tables is a summary that includes the variables' reliability values without the coder values.



Lotus* and Lotus for Gold Standard (GS)**

		Coder			
	Total	2	3	4	5
Comparisons	48	12	12	12	12
Lotus_Med	1.00	1.00	1.00	1.00	1.00
Lotus_IDEPOL	.90	.92	1.00	.92	.75
Lotus_IDEREL	.73	.67	.67	1.00	.58
LGS_Med	1.00	1.00	1.00	1.00	1.00
LGS_IDEPOL	.71	.75	.75	.67	.67
LGS_IDEREL	.60	.58	.58	.75	.50
Lotus total	.87	.86	.89	.97	.78
LGS total	.77	.78	.78	.81	.72

*Ratio of aggreement with the most frequently coded value.
**Ratio of agreement with the Gold Standard.

Standardized Lotus* and S-LGS**

		Coder			
	Total	2	3	4	5
Comparisons	48	12	12	12	12
SL_Med	.98	.98	.98	.98	.98
SL_IDEPOL	.82	.86	.98	.86	.61
SL_IDEREL	.57	.48	.48	.98	.36
SLGS_Med	1.00	1.00	1.00	1.00	1.00
SLGS_IDEPOL	.56	.63	.63	.50	.50
SLGS_IDEREL	.41	.38	.38	.63	.25
SLotus total	.79	.77	.81	.94	.65
SLGS total	.66	.67	.67	.71	.58

*Ratio of coding in agreement with all agreements expected not to be coincidential.
**Standardized Lotus for gold standard.

Reliability and accuracy with Lotus

	Coefficients				
Variables	Lotus	S-Lotus	LGS	S-LGS	Alpha
Med	1.00	.98	1.00	1.00	.00
IDEPOL	.90	.82	.71	.56	.63
IDEREL	.73	.57	.60	.41	.50

Lotus, Standardized Lotus, Lotus for Gold Standard, Standardized Lotus for Gold Standard, Krippendoffs alpha.

Figure 3: Lotus output

7. Summary

The Lotus coefficient expands the array of various reliability coefficients with an intuitively understandable, unbiased coefficient that can easily be used in SPSS. The simple Lotus coefficient represents an (actual) agreement among coders. The standardized-Lotus calculation can be regarded as a coefficient that is adjusted for chance and therefore comparable with other coefficients that contain chance adjustments. Commonly, the quality of a study is represented by the validity of its measurements more than its reliability. This article has argued in favor of a proxy for validity: accuracy. A gold standard, which essentially represents a form of expert validity, is used. The Lotus calculation for the gold standard is identical to the calculation for the principal agreement. Therefore, the coefficients for reliability and validity are directly comparable. The hope remains that publication of reliability will increase with the simple use and interpretation of these coefficients.

The various qualities can be measured are presented in Table 3. The table shows the test targets in the Quality section and provides proposed applications for the Lotus coefficient.

Table 3: Lotus for various evaluation targets

Quality	Test	Lotus Application
Random sample	Separate selection test for sample reliability	Lotus (for rare phenomena)
Data	Reliability test (+ sample reliability)	Lotus
Coders	Average deviations relative to content	Standard deviation of standardized Lotus for coders
Reproducibility	Data reliability + coder reliability	Lotus
Instrument	Reproducibility relative to content	Standardized Lotus
Coder training	Intra-coder reliability relative to content	Standardized Lotus
Codebook instructions	Instrument + coder training	Standardized Lotus
Individual coders	Agreement of individual coders with the others	Standardized Lotus per coder

8. Bibliography

Bühner, Markus (2011). Einführung in die Test- und Fragebogenkonstruktion. Pearson.

Chalmers, Alan f. (^{3}1999): What is This Thing Called Science? University of Queensland Press, St. Lucia, Queensland.

Cohen, Jacob (1960). A Coefficient of Agreement for Nominal Scales. in: Educational and Psychological Measurement, 1/1960. pp. 37–46.

Evans, William (1996). Computer-supported content analysis: Trends, tools, and techniques. Social Science Computer Review, 14(3), 269–279.

Fleiss, Joseph L. (1981). The measurement of interrater agreement. In: ibid., Statistical methods for rates and proportions. John Wiley & Sons. Pp. 212–236.

[2 items removed for the review process]

Gwet, Kilem (2001): Handbook of Inter-Rater Reliability. How to Estimate the Level of Agreement Between Two or Multiple Raters. Gaithersburg, MD: Stataxis Publishing Company.

Gupta, P. L./ Gupta, R. C./ Tripathi, R. C. (1996). Analysis of zero-adjusted count data. *Computational Statistics and Data Analysis* 23, pp. 207–218.

Holsti, O. R. (1969). Content analysis for the social sciences and humanities. Reading, MA: Addison-Wesley.

Kolb, Steffen (2004): Verlässlichkeit von Inhaltsanalysedaten. Reliabilitätstest, Errechnen und Interpretation von Reliabilitätskoeffizienten für mehr als zwei Codierer. in: Medien und Kommunikationswissenschaft, 52, 2004/3. pp. 335–354.

Krippendorff, K. (^{3}2013, first 1980). Content analysis. An introduction to its methodology. thousand oaks: sage.

Krippendorff, Klaus (2004). Reliability in content analysis: Some common misconceptions and recommendations. in: Human Communication Research. 3. pp. 411–433.

Lombard, Matthew/ Snyder-Duch, Jennifer/ Bracken, Cheryl. C. (2002). Content analysis in mass communication research: An assessment and reporting of intercoder reliability. Human Communication Research, 28, 587–604.

Merten, Klaus (1983): Inhaltsanalyse: Einführung in Theorie, Methode und Praxis. Opladen: Westdeutscher Verlag.

Neuendorf, Kimberly A. (2002). The content analysis guidebook. Thousand Oaks, CA: Sage.

Potter, James W./ Levene-Donnerstein, Deborah (1999). Rethinking Validity and Reliability in Content Analysis. Journal of Applied Communication Research. 27. 258–284.

Rogot, E./ Goldberg, I. D. (1966). A proposed index for measuring agreement in test-retest studies. J. Chronic Dis., 19, 991–1006.

Scott, William A. (1955): Reliability of content analysis: the case of nominal scale coding. *Public Opinion Quarterly 19.* pp. 321–325.

Wirth, Werner (2001): Der Codierprozess als gelenkte Rezeption. Bausteine für eine Theorie des Codierens. in: Lauf, Edmund & Wirth, Werner (eds.): Inhaltsanalysen. Perspektiven, Probleme, Potentiale. Köln: Halem Verlag: pp. 157–182.