step1(data, measurementmodel, id)

Input:

* data: a data frame with an id variable and the indicator variables for the measurementmodel
* measurementmodel: a string describing the measurement model using lavaan syntax.
* id: a character that indicates the id variable in data (the variable that indicates which observations belong to which person)

Output: a list with two elements

* fit: the lavaan fit object
* data: the input data frame (which is needed again for step2())

step2(step1output)

Input:

* step1output: the object that was generated using the step1() function

Output: a list with four elements:

* data: the input data frame to which the factor scores have been appended
* rho: a vector containing the values of *ρ* (i.e., the model-based reliabilities)
* kappa: a vector containing the values of *κ*
* fit\_step1: the lavaan fit object from step 1

step3(step2output, structuralmodel = NULL)

Input:

* step2output: the object that was generated using the step2() function
* structuralmodel: optional, a string describing the structural model using lavaan syntax. Note: The lagged variables are created automatically by the function. They are named by appending “\_lag” to the names of the latent constructs in step 1. For example, if the factors in step1() have been named “f1” and “f2”, then the lagged variables of the factor scores are automatically named “f1\_lag” and “f2\_lag”. These names must be adhered to when specifying the structuralmodel or the function will not work.

Output: a list with two elements:

* fit\_step3: the lavaan fit object
* data: the data set that was used to estimate the model

stepwiseSE(step2output, step3output)

Input:

* step2output: the object that was generated using the step2() function
* step3output: the object that was generated using the step3() function

Output: a list with three elements:

* SE: the vector of adjusted standard errors
* z\_values: the vector of adjusted z-values
* p\_values: the vector of adjusted p-values